UNIVERSITY OF CALIFORNIA
SANTA CRUZ

ROBUST DETECTION WITH LOCAL STRUCTURE TENSOR: MAXIMUM MARGIN MATRIX COSINE SIMILARITY AND BEYOND

A dissertation submitted in partial satisfaction of the
requirements for the degree of

DOCTOR OF PHILOSOPHY

in

ELECTRICAL ENGINEERING

by

Sujoy Kumar Biswas

September 2016

The Dissertation of Sujoy Kumar Biswas
is approved:

_____
Professor Peyman Milanfar, Chair

_____
Professor Roberto Manduchi

_____
Professor David Helmbold

_____
Professor Claire Gu

_____
Tyrus Miller
Vice Provost and Dean of Graduate Studies

Table of Contents

List of Figures

List of Tables

Abstract

Robust Detection with Local Structure Tensor:

Maximum Margin Matrix Cosine Similarity and Beyond

by

Sujoy Kumar Biswas

An important lesson of two decades of research in object detection comes from the success of mid-level attributes like filters or templates. Histogram of Oriented Gradients, or HOG, had long been the standard tool for representing such templates. With the introduction of convolutional neural network the focus of the feature computation has recently shifted toward the more powerful *representation learning* techniques. Though powerful and better performing, the performance benefit in representation learning comes at the price of long training phase, complicated hardware requirements, and of course, a large set of data with clean annotations.

In this thesis we propose a fundamentally different representation for image templates in the form of multidimesional tensors that looks beyond the histogram features of HOG by aggressively capturing local image geometry. As a consequence, the proposed tensor representation of templates is robust to noise and signal perturbation, and yield excellent localization performance in unconventional and difficult scenarios (e.g., low resolution, noisy image or video) where traditional HOG features fail to perform well. Moreover, owing to signal processing techniques, tensors are amenable to a rich set of tools that make object detection fast, efficient and scalable. Using an exact acceleration of matrix cosine similarity (our decision rule for detection) we make the search for a query image in a bigger target much faster. Building on these premises we have proposed a maximum

margin formulation following a relatively simple and fast training phase, to detect pedestrians in challenging videos of infrared, thermal images. The proposed kernel method is robust enough to handle missed annotations, i.e., noisy annotations, in the ground truth as exhibited in our experimental findings. The thesis contributes further by proposing a dimensionality reduction technique that not only reduces the number of feature channels during detection but also preserves local image geometry in derived subspaces resulting in better discrimination between objects and background.

In the loving memory of Baba

Acknowledgments

Every Ph.D. thesis is shaped by a story to tell a story. Mine is no exception. The five years of research that has gone into shaping this thesis is supported by the extraordinary professional contributions and generous help of many people.

First and foremost, I would like to thank my advisor Professor Peyman Milanfar for his guidance and insightful advice at various stages of my Ph.D. I express my gratitude for his patience, encouragement and confidence in my thoughts. I am also grateful for his support that was often extended beyond the academic boundary, proving instrumental to my existence as an F1 student in the United States.

I am thankful to my committee members Professor Roberto Manduchi, Professor David Helmbold, Professor Claire Gu, and Professor Benjamin Friedlander for their support and guidance in the preparation of this thesis. The suggestions they made in my candidacy exam were invaluable to plan the future course of action. Special thanks goes to Professor Gu as she agreed to serve on my committee in such a short notice.

I would like to thank my mentor in the industry, Dr. John Jordan III, for his guidance, support and technical knowledge that made my summer internship at KLA-Tencor Corporation a rich experience. I extend my gratitude to Dr. Stella Yu, ICSI, UC Berkeley, for a great internship opportunity that was full of learning experience.

During my Ph.D., I have extensively used the knowledge and skill sets that I acquired from my teachers at my alma mater, Indian Statistical Institute. In particular, I would like to express my gratitude to Professor Dipti Prasad Mukherjee, Professor Debasis Sengupta, Professor Bhabatosh Chanda, and Professor C A Murthy, for giving me a wide range of training in statistical machine learning, computer vision, and image processing.

I thank my present and former colleagues Priyam Chatterjee, Hae Jong Seo, Xiang Zhu, Hossein Talebi, Amin Kheradmand, Chelhwon Kim, Golam Md. Imran Hossain, Robert Sumner, Hossein Daraei, Sitansu Kumar Das and Snehasis Mukherjee for holding

many fruitful discussions, help and support. I am particularly thankful to Avijit Dasgupta of IIT Kharagpur for engaging in long discussions related to computer vision, as well as several collaborative attempts to gain deeper understanding of select vision algorithms. I also thank all the staffs of School of Baskin Engineering and UC Santa Cruz who helped me in one way or the other. Special thanks goes to Carol Mullane and Emily Gregg for their extensive support and friendly gestures, and all the staffs at Student Health Center who attended me when I was sick (sometimes seriously).

I would like to thank my friends Caleb Bryce, Curli Coco, Brett Smith, Mary Zuniga, Neil Foley, Vidyuth Ranjith, Nakul Dhotre, Pramod and Gayatri Sahoo, Dipankar Das, Anup Goenka, and Mrinmay Deb for their help and appreciation when things were going really tough. Special thanks goes to Cynthia Mathews for her generous help and epic support during a critical phase of my Ph.D. I would remain fortunate, and privileged, that I came to know Cynthia at some point of my time in Santa Cruz.

This thesis was impossible without the love and sacrifice of my family. The inspiration for a Ph.D. came from my uncles Narayan Prasad Sengupta and Ranjit Kumar Biswas. Also, I could finish my thesis in the peace of my mind because my uncles Narayan, Udayan and Samiran Prasad Sengupta took care of my family at home. I extend my thanks to my in-laws for their love, affection and support through thick and thin.

Finally, my Ph.D. work would remain incomplete without the mention of the three human beings who kept confidence in me when I lost mine, who suffered but put up a bright smile to give me strength, who sacrificed their personal priorities only to see me succeed: Maa, Kutty, and my loving wife, Pritha!

Santa Cruz, California

August 26th, 2016

Sujoy Kumar Biswas

'Would you tell me, please, which way I ought to go from here?'

'That depends a good deal on where you want to get to,' said the Cat.

'I don't much care where —' said Alice.

'Then it doesn't matter which way you go,' said the Cat.

'— so long as I get somewhere,' Alice added as an explanation.

'Oh, you're sure to do that,' said the Cat, 'if you only walk long enough.'

~ Lewis Carroll (Alice's Adventures in Wonderland)

Chapter 1

Introduction

Abstract – This chapter delineates the winning strategies and the lessons learnt in twenty years of research in object detection. The description then leads to current limitations and future scope of object detection. Such discussion lays the foundation and motivation for this thesis following which we summarize the important contributions made in this work.

1.1 Twenty Years of Object Detection: Winning Ideas

A search in *Google Scholar* with the phrase 'object detection' returns two important strands of ideas that survived in course of time. The first is the boosting principles of celebrated Viola-Jones face detector [12, 13] and second is the elegant kernel methodology and structured prediction framework of *deformable part model*[14, 15]. Indeed, boosting and kernel methods dominated the field of object detection almost for a decade giving birth to many useful detector designs. However, the research in object detection dates even further back mostly under the guise of its bigger parent field *visual recognition*.

Eighties to mid nineties were the time when two competing philosophies domi-

nated the early days of visual recognition. One school of thought saw the visual recognition problem as a search problem. The solution to such problems are sought in the space of *correspondences* by means of sophisticated search techniques with two-dimensional or three-dimensional constraints, hypothesis verification, controlling the search explosion, selecting subspaces of the search space, empirical testing and the combinatorics of the matching process and verification [16]. The second school of thought was inspired by Ulf Grenander's *pattern theory* that advocated the philosophy of 'analysis by synthesis'. The idea is that in order to analyze an image one must be able to synthesize it first. This image synthesis philosophy led to the birth of compositional models as [17], [18], as well as generative models like [19], [20], [21]. An excellent essay of those earlier attempts at visual recognition problems is available in the book authored by Amit Yali [22].

With the emergence of larger datasets, and an accompanying need to scale up algorithms, neither of the above ideas yielded satisfactory results in the long run. Geometry based ideas lacked robustness and failed at generalization, and generative models slowly lost ground to more powerful discriminative methods like boosting and support vector machines. However, before decaying away the geometry based methods left an important footprint in the history of object detection, namely, *pictorial structure* (a probabilistic model of representing parts in 2D articulated objects) [23], that served as a predecessor of today's widely successful deformable part model. It is fascinating to remark that the idea of using parts for improved object detection as it happens in pictorial structure dates back to the early work of Fischler and Elschlager in 1971 [24]. The probabilistic framework introduced in *pictorial structure* [23, 25], to facilitate matching of parts and their geometric verification, comes under a broad class of (generative) models known as constellation models [26]. Pictorial structures work by matching graphs (the parts of objects occupy

nodes and their spatial relationship defines the edges) between a query and target image following expectation-maximization. What set [23, 25] apart from other constellation based approaches, e.g., [27], are two assumptions (acyclic graph and Mahalanobis distance function) that reduced the intractable matching complexity of $\mathcal{O}(N^P)$ (N is discrete number of possible locations on an image grid and P denotes number of parts) to a manageable one $\mathcal{O}(NP)$. A different research initiative in this direction that deviated from constellation based approach studied the fitting of templates and minimizing a global energy function [28, 29]. In the years to follow, a reasonably successful template matching strategy for part based object detection was developed by Leibe et al. [30, 31]. They called it *Implicit Shape Model (ISM)* that learned a codebook of local appearance patches along with their relative positions with respect to center of the object. During test time, the local appearance patches act like *parts*, casting votes to the probable centers of the objects providing a likelihood of object center's location.

Another strand of ideas that leads to very fast and efficient detection in test time started with the seminal work of Viola and Jones in 2001 [12]. The Viola-Jones object detector was seminal from multiple perspectives. Not only it made the first use of large scale machine learning in the context of object detection, but it also laid the foundation to the design of modern object detectors — sliding window analysis and multiscale search, which were done in efficient manners through the clever use of integral image, later became standard elements of a full fledged detection pipeline. In contrast to part based models equipped with deformable representations, the classic Viola and Jones detector however uses a monolithic object representation.

1.1.1 The Rise of Kernels

Pictorial structure, *Implicit Shape Model* and other part based models (notably [2, 32]) duly underscored the importance of parts in generic detector design. In most of the cases the parts are treated as rigid templates. In course of time, the template based approaches in object detection exhibited a need for scaling up with sophisticated machine learning algorithms to perform better on increasing dataset sizes. The first work which tried to combine the monolithic rigid template of Viola Jones with the support vector learning was authored by Mohan et al. [33]. They applied Haar wavelet transform to candidate regions in the images and apply support vector machines to resultant features reporting reasonably good performance on a pedestrian dataset. However, the landmark paper that first showed us the *direct* way to design template features for an efficient but effective linear kernel detector came from Dalal & Triggs [34]. It marked the true beginning of mid level attributes as the feature representation for object detection.

The deformable part model [15] is the natural confluence of these two streams of ideas, HOG based linear SVM and pictorial structures. The vision community witnessed its development in the middle of a series of excellent and relevant work, notably by Deva Ramanan [35, 36], Andrew Zisserman [37], Jitendra Malik [38, 39] and Christoph Lampert [40]. The inference algorithm of Pictorial Structure is intelligently used in the learning stage of the structured prediction (also known as learning by inference [41]) leading to efficient computation of latent terms in the objective function. The whole training process is discriminative and results in superior detection performance in comparison to its earlier generative counterparts. The idea of using mixtures of components in deformable part model to handle wide variation in views and poses further widened its applicability to include pose recognition (e.g., [42] serves as the current state of the art).

Though the kernel methodologies [43] have contributed to some of the best detector designs (e.g., [15], [44]) in the history of object detection so far, one limitation that persisted in this school of thought was the absence of feature learning. Indeed, boosting based methodologies which leveraged the feature learning remained somewhat ignored in the kernel based methods. Only notable exception is the work of Ren and Ramanan [45] that explored sparse code learning from HOG features reporting reasonably good performance only with a root filter devoid of part designs.

1.1.2 Ensemble Learning

The primary motivation for boosting lied in feature learning. The three important lessons of Viola Jones detector have been fast feature computation with integral image, a learnt ensemble of weak classifiers, and a very efficient cascade framework to reject sliding windows early in the process pipeline circumventing further computation. Cascades were explored further in computer vision to improve detection efficiency in runtime [46, 47]. One of the most notable contributions in the boosting based object detection is made by Piotr Dollar in a series of work involving multichannel boosting [48], pose regression [49], edge detection [50], and developing mid level attributes for object detection [51].

Boosting based detectors have been particularly successful in upright people detection, particularly pedestrians, across a wide range of pedestrian benchmark data sets. The benchmark study of pedestrians by Dollar et al. [52, 53] is worth mentioning in this regard.

1.1.3 Deep Convolutional Neural Network

The introduction of deep convolutional neural nets [54] in computer vision caused drastic improvement in performance in almost all visual recognition datasets. Several research initiatives explored ways to combine convolutional networks for PASCAL style object detection. Notable mentions include [55, 56, 57] with approaches ranging from regression to saliency based object detection with region proposal. The recently proposed region based convolutional network (R-CNN) proposed by Girshick et al. [58, 59] approaches the detection problem from similar perspective but proposes a scalable learning algorithm that uses three modules. The first module extracts category independent region proposals followed by feature extraction from them by the second module, and the last one is a set of linear SVMs. In more recent work, the R-CNN architecture is applied to other tasks [60, 61] and also extended for faster implementation [62].

In hindsight, it is important to note that with the arrival of deep architecture the focus of vision community has shifted to learning representations. The learning of features provide substantial benefit in performance. The engineered representation of templates, as in HOG, therefore experiences replacement by learnt features wherever possible.

1.1.4 Emergence of a Standard Detection Pipeline

Design of templates or filters, their representation by means of learning (i.e., deep architecture) or engineering (i.e., HOG templates), linear SVMs that often translate to multichannel correlation along multiple feature channels are some of the important learning lessons of past decades. Boosted classifiers and deformable part models served as learning methodologies to scale up the architecture. But in due course the object detection community discovered many other limitations which required scalable solutions. Poses, scales,

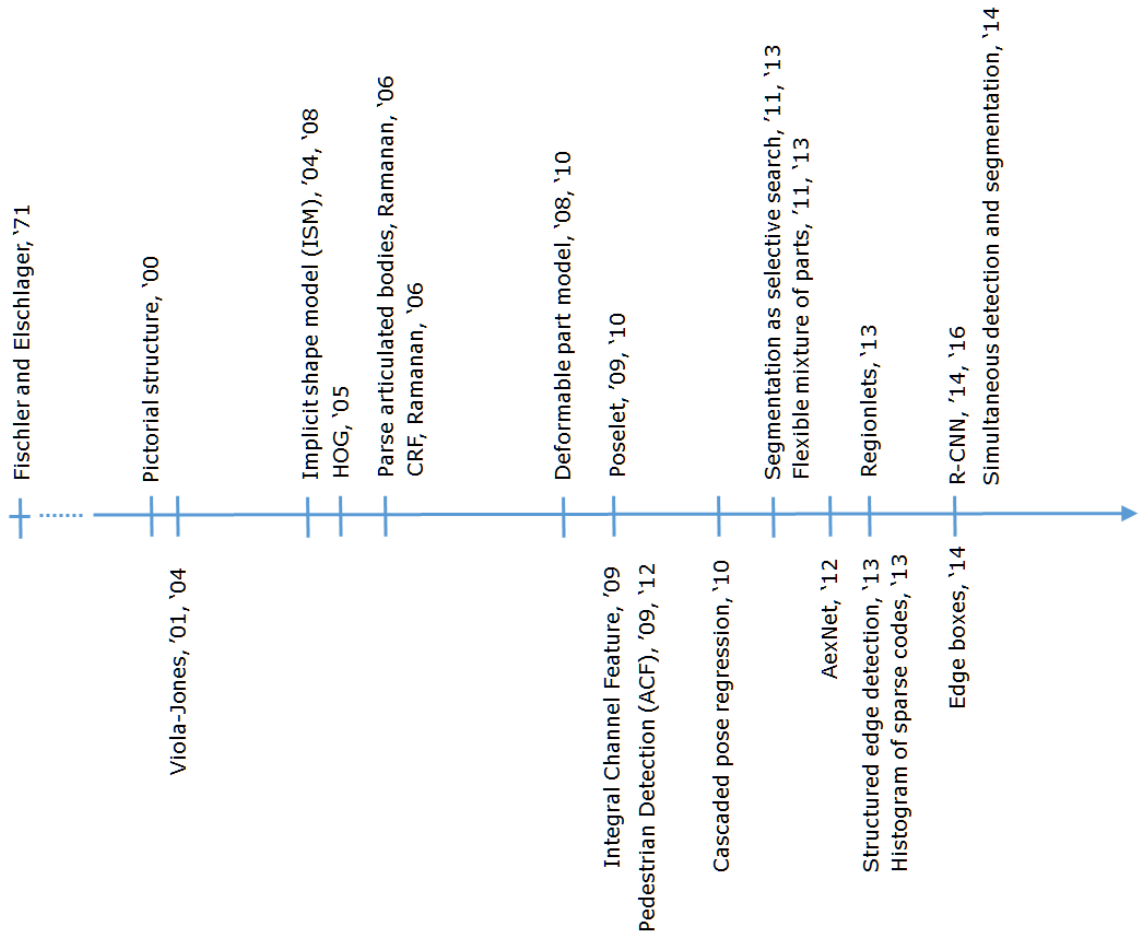


Figure 1.1: **Important strands of research in twenty years of object detection**: Generative models gave way to more powerful discriminative models in early 2000. Geometry based models like pictorial structures, through an efficient means of part representation like HOG, led to part based deformable template in an efficient biconvex optimization framework of structured prediction. A different strands of research witnessed feature learning and fast detection in runtime within the framework of boosting methods. Later breakthroughs in representation learning provided a means to learn features with deep convolutional neural nets (AlexNet, 2012, R-CNN, 2014). A shift from sliding window analysis to pre-segmentation based seeds selection for a coarse-to-fine detection strategy evolved into a new direction of research, namely, object proposal.

orientations and part-complexities they all throw challenges when one goes on to build a scalable object detection architecture.

Sliding window analysis, and multiscale search (scaling either the detector or the test image) for handling varying scales, seemed to fail in terms of efficiency because of the overwhelmingly large number of sliding windows, over a wide range of scales, typically observed in an image (Fig. 1.1). One important limitation of otherwise successful part based model is the runtime required for the detection of root filters and several part filters, across scales, costing efficiency. When combined with a mixture model, it gets harder to contain the overwhelming number of parts and their poses in connection with multiple components.

As a solution to this predicament the object detection community has extended the idea of saliency detection to find probable cadidate regions (called *object proposals* in an image through a coarse search that can later undergo fine inspection. Such coarse-to-fine approach [63, 64, 65] would reduce complexities in a detection-by-classifying-windows paradigm. This has sparked a new line of research in computing object proposals as initial candidates or seeds for final detection scores [66]. In retrospect, we see the quest for more and more sophisticated means for *representing mid level attributes* is on and as a result we see a number of excellent research work, namely Edge Boxes [51], BING [67], R-CNN [59], and Deepbox [68], among many others.

1.2 Motivations and Scope of Thesis

The success of midlevel attributes, their representation, learning such representations with carefully annotated big training sets, correlations as detection principle — these ideas definitely underlined the important lessons learnt in the past research endeavors as

Figure 1.2: Noisy, low resolution thermal images offer unique challenges to reliable estimation of features. A robust strategy to effectively handle noise is a necessity for good feature representation.

outlined in the previous sections (Fig. 1.1). However, one important limitation of the research endeavors so far is the absence of a robust feature computation strategy in noisy,

challenging environment. The images in typical computer vision datasets are high resolution snapshots of objects where the objects, associated contours and textures are relatively distinct. There is no need to enhance the image, nor the need to compute features under uncertainty. This makes the current research endeavors somewhat limited in their application to domains like low resolution, noisy, thermal infrared images. Histogram feature, if computed in such scenario, is often a naive estimate of local structure in the presence of noisy gradients (Fig. 1.2).

Similar to the lack of a principled strategy for feature computation, detection in a noisy and challenging environment has also remained a relatively unexplored area. Besides robust feature computation, the right decision rule for detection, under uncertainty, also requires careful investigation. This also includes the challenging one shot detection setting where search for a given query object (in the form of a smaller image) takes place in a bigger target image (target may contain one or multiple instances of the query object). Being a one shot setting, there lies no scope for an explicit training mechanism. An important question is how to make prediction in such a restricted environment.

Another growing concern in the present trend of object detection is the notoriously complicated and unduly long training phase. Indeed, reduction of training time has initiated a new area of research in object detection [69]. A clean, efficient and short training phase is often the need of the day. In fact, in industrial applications it is often the case that a preliminary detector with decent detection performance is required to get a first hand feel of the dataset, as well as to aid the training process. In both the cases, a reasonably good detector with efficient training is a valuable necessity.

In a recent line of research, *Locally Adaptive Regression Kernel* or *LARK* [6, 70, 71] showed excellent performance in capturing local image geometry. Originally designed for

(a) (b)

Figure 1.3: One-shot object detection requires a robust decision rule and appropriate thresholding for deciding object locations in absence of prior training data. In both (a) and (b), the queries on top left need to be detected in the bigger target images under significant pose, scale and color variation.

denoising and image filtering purpose [72], their ability to capture spatial structure even in the middle of heavy noise has made them the right choice as low level descriptor [73]. At the heart of LARK lies *Local Steering Kernel* (*LSK*) that measures orientation of texture at a given point. Of course, HOG or histogram based descriptor comes with the added advantage of local invariance but LARK, or LSK, sacrifices the local invariance in exchange for robustness. To meet the need for detection in a restricted environment, for example an one-shot detection scenario in Fig. 1.3, a Bayes optimal decision rule has been proposed in [6]. The decision rule results in the formulation of a similarity function named as Matrix Cosine Similarity (MCS). MCS ensures good performance in challenging scenarios like low resolution and noise. Its high sensitivity to fairly low visual similarity makes it a good candidate for a generalization scheme. However, MCS as proposed in [6, 70, 71] is computationally demanding and its long runtime computation makes it a very inefficient metric for deciding object locations. This limits the scalability of the detection architecture proposed in [6, 70, 71].

In addition, the related work till date explores LSK in various detection scenarios but exploration of MCS has so far been limited, lacking a suitable generalization methodology. Seo and Milanfar [6, 70] restricted LSK and MCS to the study of one shot object detection. They did not consider a learning based approach for extending the general applicability of their work. Inspired by their work, later investigation by Zoidi et al. [74] used LSK for human detection in videos. In a further extension, You et al. [75] used LSK features for learning local metric for ensemle based object detection. Though laudable, in all of such research endeavors, the generalization principle of MCS has been missing. MCS, in hindsight, is naturally associated with LSK features: it is a consequence of representing features as tensors. The whole premise behind this thesis is that tensors combined with MCS kernel would invariably lead to a novel training scheme, with faster and robust prediction result. This line of research has so far been unexplored because the perspective till date focuses on using LSK features as standalone element of traditional object detection framework. Our intention lies in bridging this gap.

In this thesis, we differ from earlier approaches in feature representation and advocate a tensor representation of mid level attributes. Our intended approach explores beyond the histogram features that till date serves as the standard regimen in engineered template based object detection. Though this sounds high dimensional, we will show in this thesis that the tensor representation permits efficient scaling up of the one-shot detection methodologies as available in [6, 70, 71]. The proposed tensor form of features not only performs well in noisy environments but also makes it amenable to large scale machine learning treatments when combined with MCS as a decision rule. The following section succinctly enumerates our contribution.

1.3 Thesis Contributions

The contribution of this thesis starts from the fundamental notion of viewing images as tensors. The traditional custom of deriving feature vectors from a sliding window is refuted here. Feature computation, learning and inference take place keeping the tensor representation intact. We will show how we leverage this fact in terms of speed and performance.

1.3.1 Beyond Histogram: Representing Local geometry with Structure Tensor

The histogram computation in HOG essentially provides a summary of image statistics in a local region. Our intention is to compute spatial geometry around each pixel. As a consequence for any $M \times N$ grayscale image the computation of image geometry at each pixel supplies us a $M \times N \times l$ tensor, where l represents the number of feature channels. Each of such l dimensional feature encodes the structural geometry around the relevant pixel. Indeed, the feature vector is essentially a structure tensor representative of local geometry. The first half of Chapter 2 describes the introduction and foundation of the basic conventions. We scale up on this convention in the second half of Chapter 2 as follows.

1.3.2 Scaling up with Faster Matrix Cosine Similarity

As previously indicated, MCS as a measure of similarity is very expensive. A brute force computation of MCS ruins the efficiency of the detection during runtime. We propose a fast computation of MCS to ameliorate such issues. In fact, such acceleration becomes a natural consequence of tensor representation of features. Our contribution in this area also establishes a theoretical connection with signal processing techniques. We show in Chapter 2, how rich tools in signal processing can result in superior detection

performance.

1.3.3 Maximum Margin Matrix Cosine Similarity with Structure Tensor

Building on the accelerated detection stage we propose a maximum margin framework using LSK as the tensor feature and MCS as the kernel. As a result, the support vectors in our architectures are LSK tensors of positive and negative examples which are hard for the classifiers. The proposed *support tensor machine* framework achieves fast detection, rapid hard mining and improved prediction in runtime. Chapter 3 describes the methodology and reports the performance on challenging thermal image dataset.

1.3.4 Effective Dimension Reduction for Improved Detection

Reduction of dimension of a tensor, or channels of a multidimensional feature tensor, is always a welcome idea that improves runtime performance because less computation is involved. In fact, the projection of features onto lower dimensions often makes the intra class separation wider, e.g., linear discriminant analysis [76, 77], leading to better separation of objects from background, improving detection. We present a study in Chapter 4 investigating the role of dimensionality reduction in the context of LSK features, the image datasets that we deal with (thermal infrared images), and the resulting detection performance (in terms of both time and effectiveness).

In summary, the contributions above make a learning based extension of Seo and Milanfar [6, 70]. However, such contribution have also made the ingredients ready for general applicability to a host of other scenarios. The final chapter, Chapter 5, outlines a few such potential uses which are immediate and readily discernible. We also conclude the thesis in that chapter.

Chapter 2

Foundation and Fast Matrix Cosine Similarity

Abstract – We introduce the founding concepts in this chapter with emphasis on the feature representation and decision rule. In the second half of this chapter we show how the inefficiencies in the proposed architecture of Seo and Milanfar in [6, 70] have been circumvented, leading to an improved detection performance. Accelerated detection is the key to building a scalable solution to a large scale machine learning setup, as described in the forthcoming chapters.

2.1 Looking Beyond the Histogram Features: Motivation and Overview

A standard object detection pipeline has long accepted *Histogram of Oriented Gradients*, or *HOG*, the de facto building block to represent a template. Before feature learning became popular [54], HOG templates were widely used with success for building or scaling up a full fledged detection system. In fact, it still serves as the top most choice for

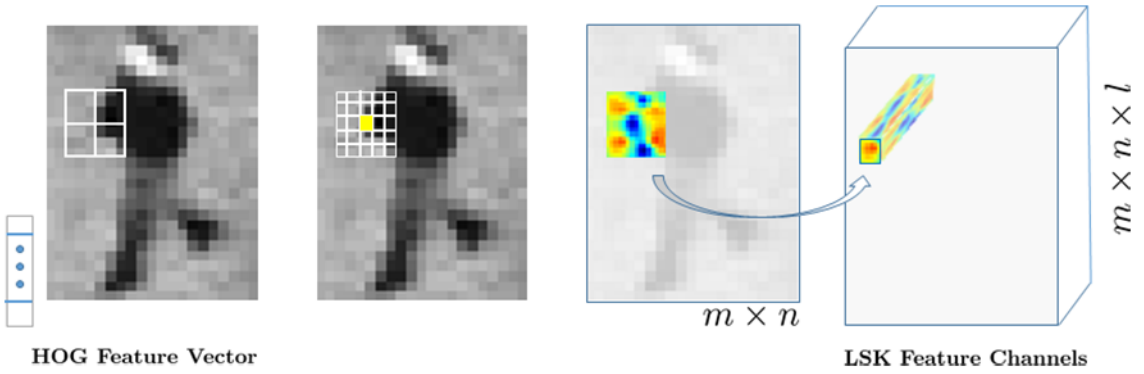


Figure 2.1: Unlike HOG (far left) that derives a feature vector by pooling histograms from rectangular cells, the steering kernel coefficients are computed over a patch, between the central pixel (shown in yellow in second image from left) and its neighbors, followed by concatenation into a vector (two images on far right). The dense computation ensures that we end up with a tensor feature having same width and height as that of the original image.

engineering templates if one does not resort to the long training procedure for feature learning.

As previously indicated HOG works by computing a summary of gradient distribution in a given region. Such statistics are often inadequate in case of noisy, low resolution images, typically found in infrared thermal videos. Instead, Local Steering Kernel (or LSK), works by looking at the gradient distribution over a region, and then estimating the **dominant** orientation in that region. Such estimate is carried out reliably by computing a *smoothed* local structure tensor. The local structure tensor is crudely described as an **average** covariance matrix (of local image gradients) which represents the topology of local signal manifold. What all this boils down to is that the local structure tensor captures dominant orientation in a local patch. We use this orientation information in computing the LSK coefficients between central pixel in the region and all its neighbors (refer to Fig

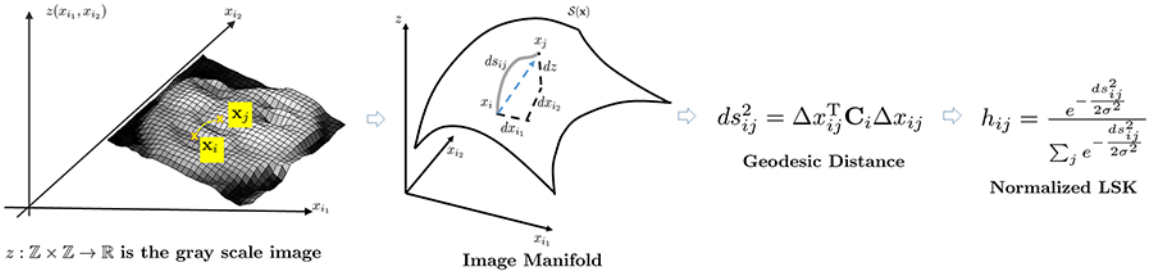


Figure 2.2: Geodesic interpretation of LSK: the geodesic distance (ds_{ij}) between the points $\mathbf{x}_i$ and $\mathbf{x}_j$ on the image manifold $\mathcal{S}(\mathbf{x})$ is used to derive the LSK coefficients

2.1).

The LSK coefficients of the central pixel in the given patch are stored in a vector as shown in the Fig. 2.1. We do this process for all the pixels in the image in a dense fashion. Research has shown that dense computation of low level descriptors significantly improve the recognition accuracy [6]. It follows that these densely computed descriptors are highly informative, because each feature vector stores local structure in the neighborhood of the concerned pixels.

Dense computation of the descriptors make them discrminitaory no doubt, but when taken together, they tend to be redundant. Hence, we derive features by applying dimensionality reduction (namely, PCA) to these tensor descriptors, in order to retain only the salient characteristics of the local steering kernels. The result is also a tensor of same height and width but reduced number of feature channels.

2.2 Tensor Features: Definition and Description

We visualize a $m \times n$ gray-scale image as the parameterized image surface $\mathcal{S}(\mathbf{x}_i) = \{\mathbf{x}_i, z(\mathbf{x}_i)\}$, where $\mathbf{x}_i$ denotes the 2D coordinate vector $\mathbf{x}_i = [x_{i_1}, x_{i_2}]^{\mathrm{T}}$, having intensity $z(\mathbf{x}_i)$.

The local geodesic distance between the two neighboring points $\mathbf{x_i}$ and $\mathbf{x_j}$ on the image manifold $\mathscr{S}(\mathbf{x_i})$ can be approximated by the differential arc length ds_{ij} as described below [71] (see Fig. 2.2). In the following derivation, $\mathbf{x}_j \in \mathscr{N}(\mathbf{x}_i)$ where $\mathscr{N}(.)$ denotes neighborhood operator defined over a local window of size $p \times p$.

$$
\begin{aligned}
ds_{ij}^2 &= dx_{i_1}^2 + dx_{i_2}^2 + dz^2, \text{ where } \mathbf{x}_j \in \mathscr{N}(\mathbf{x}_i) \\
&= dx_{i_1}^2 + dx_{i_2}^2 + (\frac{\partial z}{\partial x_{i_1}} dx_{i_1} + \frac{\partial z}{\partial x_{i_2}} dx_{i_2})^2, \\
&\approx \Delta x_{i_1 j}^2 + \Delta x_{i_2 j}^2 + (\frac{\Delta z}{\Delta x_{i_1 j}} \Delta x_{i_1 j} + \frac{\Delta z}{\Delta x_{i_2 j}} \Delta x_{i_2 j})^2, \\
&= \begin{pmatrix} \Delta x_{i_1 j} & \Delta x_{i_2 j} \end{pmatrix} \begin{pmatrix} \frac{\Delta z}{\Delta x_{i_1 j}}^2 + 1 & \frac{\Delta z}{\Delta x_{i_1 j}} \frac{\Delta z}{\Delta x_{i_2 j}} \\ \frac{\Delta z}{\Delta x_{i_1 j}} \frac{\Delta z}{\Delta x_{i_2 j}} & \frac{\Delta z}{\Delta x_{i_2 j}}^2 + 1 \end{pmatrix} \begin{pmatrix} \Delta x_{i_1 j} \\ \Delta x_{i_2 j} \end{pmatrix}^{\mathrm{T}}, \\
&= \Delta x_{ij}^{\mathrm{T}} \mathbf{C}_i \Delta x_{ij} + \Delta x_{ij}^{\mathrm{T}} \Delta x_{ij}^{\mathrm{T}}, \\
&\approx \Delta x_{ij}^{\mathrm{T}} \mathbf{C}_i \Delta x_{ij}^{\mathrm{T}}, \qquad (2.1)
\end{aligned}
$$

The approximation involves the following discretizations: $dx_{i_1} \approx \Delta x_{i_1 j_1} = x_{j_1} - x_{i_1}$, and $dx_{i_2} \approx \Delta x_{i_2 j_2} = x_{j_2} - x_{i_2}$ (i.e., $\Delta x_{i_1 j_1}$ and $\Delta x_{i_2 j_2}$ representing displacements along the two image-axes in Fig. 2.2). Also, we assume $\Delta x_{ij} = [\Delta x_{i_1 j_1} \;\; \Delta x_{i_2 j_2}]^{\mathrm{T}}$, and the matrix $\mathbf{C}_i$ denotes the local gradient covariance matrix (also called as steering matrix in [72]) computed at $\mathbf{x}_i$. The last step in the above derivation results from ignoring the data independent part $\Delta x_{ij}^{\mathrm{T}} \Delta x_{ij}^{\mathrm{T}}$ which remains constant as it does not include image information.

Straightforward computation of $\mathbf{C}_i$ in (2.1) based on raw image gradient at a single pixel may be too noisy. Therefore, to estimate $\mathbf{C}_i$ in a robust fashion, first we compute the derivatives of the image signal $z(\mathbf{x}_i)$ over a patch Ω_i of pixels centered at pixel $\mathbf{x}_i$ and we denote such *average* steering matrix as $\mathbf{C}_{\Omega_i}$. This accumulation of first derivatives guards against the undesirable effect of noise and perturbations. Secondly, we further smooth the signal manifold, to strictly focus on the dominant pattern of local texture, by computing

$\mathbf{C}_{\Omega_i}$ in a stable way that includes eigen-decomposition. Combining these two steps we write the final expression of the steering matrix $\mathbf{C}_{\Omega_i}$ as follows:

$$\begin{aligned}
\mathbf{C}_{\Omega_i} &= \sum_{m \in \Omega_i} \begin{pmatrix} \frac{\Delta z(m)}{\Delta x_{i_1}}^2 & \frac{\Delta z(m)}{\Delta x_{i_1}} \cdot \frac{\Delta z(m)}{\Delta x_{i_2}} \\ \frac{\Delta z(m)}{\Delta x_{i_1}} \cdot \frac{\Delta z(m)}{\Delta x_{i_2}} & \frac{\Delta z(m)}{\Delta x_{i_2}}^2 \end{pmatrix}, \\
&= \nu_1 \mathbf{u_1} \mathbf{u_1}^{\mathrm{T}} + \nu_2 \mathbf{u_2} \mathbf{u_2}^{\mathrm{T}}, \\
&\approx (\sqrt{\nu_1 \nu_2} + \varepsilon)^{\theta} \cdot \\
&\quad (\frac{\sqrt{\nu_1} + \tau}{\sqrt{\nu_2} + \tau} \mathbf{u_1} \mathbf{u_1}^{\mathrm{T}} + \frac{\sqrt{\nu_1} + \tau}{\sqrt{\nu_2} + \tau} \mathbf{u_2} \mathbf{u_2}^{\mathrm{T}}),
\end{aligned} \tag{2.2}$$

where, ν_1 and ν_2 are eigenvalues of $\mathbf{C}_{\Omega_i}$ corresponding to eigenvectors $\mathbf{u}_1$ and $\mathbf{u}_2$, respectively. Also in the derivation above, $\varepsilon, \tau, \theta$ are regularization parameters to avoid numerical instabilities and kept constant throughout all the experiments in this paper at 10^{-7}, 1 and 0.1 respectively.

2.2.1 From Structure Tensor to Local Steering Kernel (LSK)

The robust estimate of the local steering matrix $\mathbf{C}_{\Omega_i}$ is used in the computation of local geodesic distance ds_{ij} following Eq. 2.1. The geodesic distance between pixel $\mathbf{x}_i$ and $\mathbf{x}_j$ is raised to the negative exponent to define the *Local Steering Kernel* coefficient as follows:

$$\exp(-\frac{ds_{ij}^2}{2\sigma^2}) = \exp(-\frac{\Delta x_{ij}^{\mathrm{T}} \mathbf{C}_{\Omega_i} \Delta x_{ij}^{\mathrm{T}}}{2\sigma^2}). \tag{2.3}$$

A few steps of calculations would reveal that the above form does indeed represent a *kernel* satisfying the properties of the Mercer's theorem [78]. Such kernel coefficients are the building blocks of our feature tensor. We describe the formation of feature tensor in the following passage.

The descriptor at location $\mathbf{x}_i$ is denoted as a l-dimensional vector $\mathbf{h}_i \in \mathbb{R}^l$. The elements of $\mathbf{h}_i$ are the LSK coefficients 2.3 defined between $\mathbf{x}_i$ and its neighbors over a

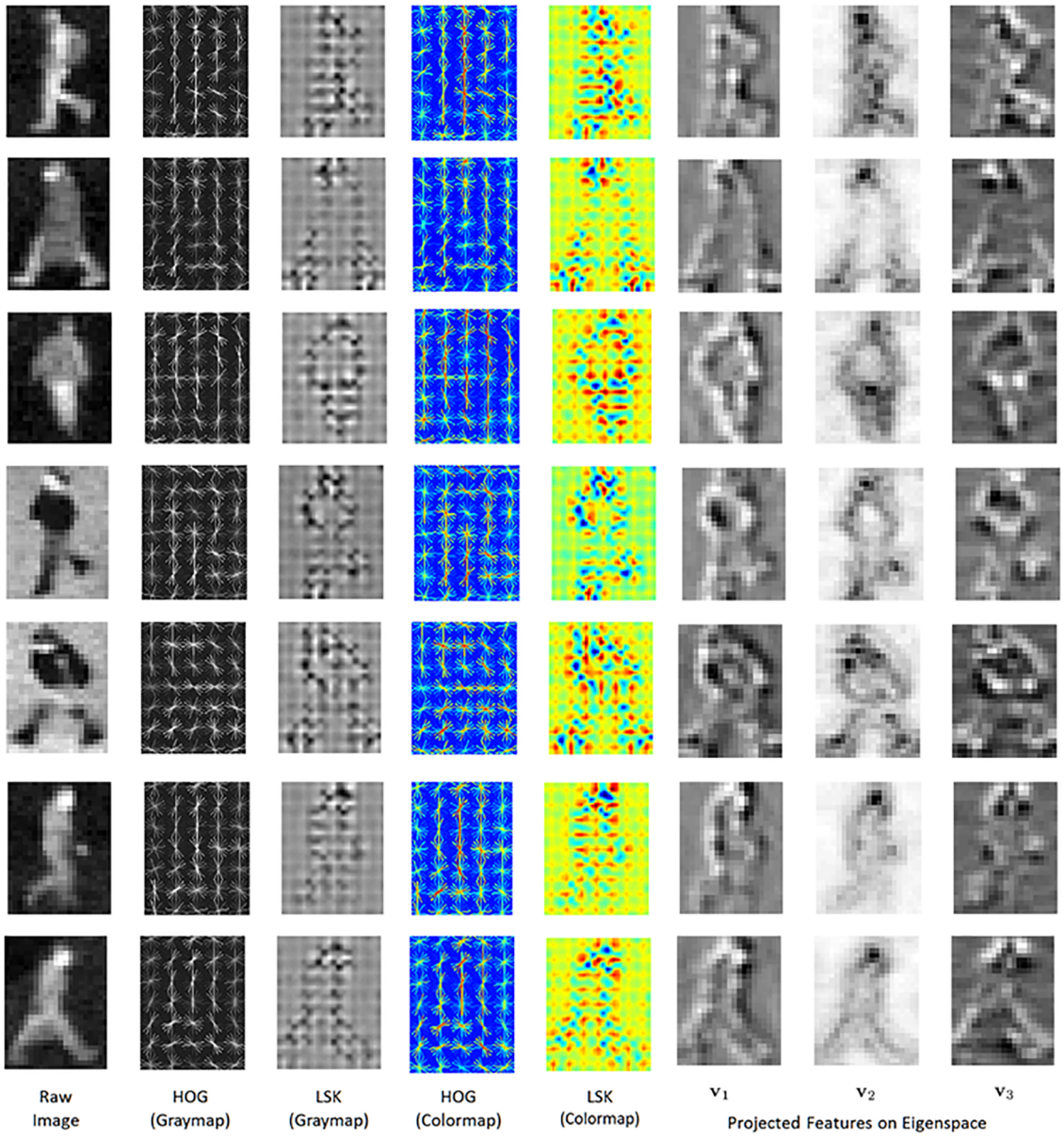


Figure 2.3: **LSK Visualization** First column displays raw infrared images of pedestrians in different poses. HOG and LSK features are displayed in grayscale (second and third column respectively) as well as in in colormap (fourth and fifth column respectively). LSK is displayed thus after computing them in non overlapping blocks. Also, both HOG and LSK are displayed after scaling up 3 times. Columns sixth, seventh and eighth show LSK features after projecting LSK descriptors on three leading principal components.

$p \times p$ local window. Specifically, $\mathbf{h}_i = (h_{i1}, h_{i2}, \ldots, h_{ij}, \ldots, h_{p^2})$, and $\mathbf{h}_i$ corresponds to the pixel location $\mathbf{x}_i$. It is also observed, that LSK descriptors when normalized to a unit vector become robust to illumination changes. The normalization in the denominator is carried out by summing the local geodesic similarities over all the neighbors of $\mathbf{x}_i$ in its $p \times p$ local neighborhood.

Following the discussion above and 2.3 in particular, we define the general term h_{ij} between two descriptor locations $\mathbf{x}_i$ and $\mathbf{x}_j$ as follows:

$$h_{ij} = \frac{e^{-\frac{ds_{ij}^2}{2\sigma^2}}}{\sum_{j=1}^{p^2} e^{-\frac{ds_{ij}^2}{2\sigma^2}}}, \quad j = 1, 2, \ldots, p^2, \tag{2.4}$$

where ds_{ij} is approximated as in (2.1). The local image descriptors are computed densely at every pixel, that makes the number of descriptors from the given image as mn. The descriptor vectors $\mathbf{h}_i$, are stacked together to define the descriptor matrix for the image as $\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_{mn}] \in \mathbb{R}^{m \times n \times l)}$.

2.2.2 Discriminatory Subspace Learning with PCA

To distill the redundancy resulting from dense computation of descriptors we embed $\mathbf{H}$ in a low dimensional but discriminatory subspace $\mathbf{v}$. It is done in a way such that $\mathbf{h}_i$, when projected on $\mathbf{v}$, respect the local geometric pattern. One way to compute such subspace can be principal component analysis or PCA. The desired set of eigenvectors which builds the low dimensional PCA subspace comprises the leading d eigenvectors.

We collect the set of d eigenvectors as columns of $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_d] \in \mathbb{R}^{l \times d}$. Since the descriptors are densely computed they typically lie on a lower dimensional manifold. As a consequence, we can expect d to be quite small in comparison to the dimension l of the descriptors. In practice, d is selected to be a small integer, and it turns out that this

small set of eigenvectors is good enough to discriminate the query from the background clutter. It turns out that 3 or 4 eigenvalues are good enough to store 75% of the energy in the eigenspectrum, which is used as a measure to restrict the number of eigenvectors to a leading few.

The descriptor matrices $\mathbf{H}$ are projected on $\mathbf{V}$ to produce salient features that preserve important spatial details. The locally salient LSK tensor features $\mathbf{F}$ are defined by the following equations:

$$\mathbf{F} = \mathbf{V}^{\mathrm{T}}\mathbf{H} \in \mathbb{R}^{d\times(mn)}; \tag{2.5}$$

The tensor features $\mathbf{F}$ learnt with PCA are shown in Fig. 2.3. The results are contrasted to those derived from HOG. The results in the figure(s) demonstrate that LSK is able to preserve greater amount of details in the projected features than HOG. During detection the detailed spatial geometry captured in LSK result in better localization. The reason why LSK representations inculcate more information compared to HOG features lies in the construction of steering matrix $\mathbf{C}_{\Omega_i}$, and later in the development of steering kernel h_{ij}.

2.3 One Shot Object Detection

One shot, generic object detection involves searching for a single query object in a larger target image [79, 80]. Query objects typically appear in target images with wide variations both geometric as well as optical. Geometric variations can include severe changes in scale and orientation (pose) of the query, whereas optical variations may result from differences in lighting, resolution and noise level. Besides, presence of clutter and not having enough out-of-class examples make the detection task prone to false alarms.

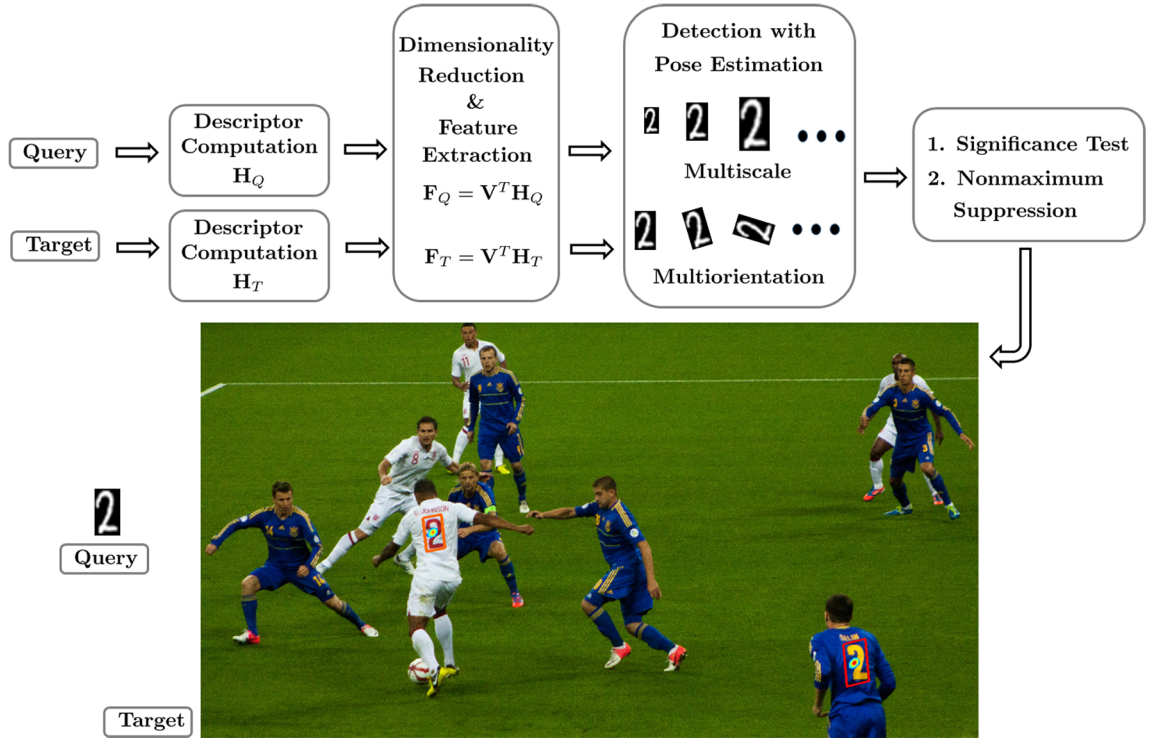


Figure 2.4: Overview of our one shot detection scheme: we aim to detect a given query [81] (e.g., symbol, face, human pose, car, flower) appearing in a visually similar manner in a bigger target image [1]

We denote the query and target images by Q and T respectively, and compute high dimensional descriptors densely over both query and target, storing them in descriptor matrices $\mathbf{H}_Q$ and $\mathbf{H}_T$, respectively. The dense computations make the descriptors highly informative but also redundant. Hence, to facilitate fast, efficient and effective detection we extract compact but salient features $\mathbf{F}_Q$ and $\mathbf{F}_T$ from the high dimensional descriptors. The subspace $\mathbf{V}$ is learnt from the query image alone. Since, T is bigger than query Q, we

[1] The original picture (available online, 7th September, 2014: `https://www.flickr.com/photos/tdd/8504835638/`) used (strictly for academic purpose) is owned by Tomasz Dunn, and licensed under Creative Commons Attribution 2.0 Generic (CC BY 2.0) for free use and modification with attribution.

sweep the query window over target, and comparing features $\mathbf{F}_Q$ and $\mathbf{F}_{T_i}$ (extracted from i-th position of the sliding window over T) we estimate the likelihood of the presence of Q in T (Fig. 2.4).

The traditional sliding window based detection sweeps the query window over T, and at each position $\mathbf{x}_i$ (center of sliding window) in T the MCS decision rule [6] is computed as follows:

$$\rho_i = \rho(\mathbf{F}_Q, \mathbf{F}_{T_i}) = \text{trace}(\frac{\mathbf{F}_Q^{\text{T}}\mathbf{F}_{T_i}}{\|\mathbf{F}_Q\|_F \|\mathbf{F}_{T_i}\|_F}) \in [-1, 1], \tag{2.6}$$

where $\|\cdot\|_F$ denotes the Frobenius norm. To suppress the small correlation values of (2.6) the Lawley-Hotelling Trace statistic ([82, 83]) $f(\rho_i) = \frac{\rho_i^2}{1-\rho_i^2}$ was proposed in [6]. Our findings support that $f(\rho_i)$ (henceforth be called resemblance values) does suppress smaller values (mostly coming from false alarms).

2.3.1 FDR Control with Benjamini-Hochberg Method

Since we do not have prior knowledge of the object's response to our detector (since there is no scope of training in one shot detection), thresholding the $f(\rho)$ values at the right point becomes a challenge. We address the issue of distinguishing signal from the false alarms by employing Benjamini-Hochberg procedure [84, 70] for false discovery rate (FDR) control.

Let our proposed detector impose a threshold τ (to be determined) on the maximum likelihood estimate (over pose and scales) of resemblance values $f(\rho_i), \forall i = 1, 2, \ldots, MN$, giving us R as the total number of detections of which W are incorrect (i.e., false alarms). In what follows, $U = \frac{W}{R}$ is the proportion of error committed by our detector. Since we do

not know W apriori, U denotes the unobservable random quotient —

$$U = \begin{cases} \frac{W}{R}, \text{ if } R > 0, \\ 0, \qquad\qquad \text{otherwise.} \end{cases} \tag{2.7}$$

The FDR, defined by the expectation $\mathbf{E}(U)$, is controlled at a desired level α while maximizing the expectation $\mathbf{E}(R)$. We have $p_1, p_2, \ldots, p_{MN}$ which denote the p-values ($p_i = 1 - P_{\rho_i}$, where P_{ρ_i} is the cumulative distribution function of resemblance values $f(\rho)$) corresponding to $\{f(\rho_1), f(\rho_2), \ldots, f(\rho_{MN})\}$. FDR control is readily implemented as follows:

- Step 1: define maximum allowable desired FDR bound (on an average) $\alpha \in (0, 1)$.
- Step 2: order the p-values in ascending order yielding $\{p_{(1)} \leq p_{(2)} \leq \ldots \leq p_{(MN)}\}$.
- Step 3: let $f(\rho_{(r)})$ be the query window on target corresponding to $p_{(r)}$. Let β be the largest r for which $p_{(r)} \leq \frac{r}{MN}\alpha$.
- Step 4: identify the threshold τ corresponding to $p_{(\beta)}$, and predict that the query windows (centered at $\mathbf{x}_i$) having $f(\rho_i)$ above τ contain instances of the query object Q.

After the significance testing with τ as above we perform non-maxima suppression [85] as the last step to eliminate duplicate detections close to an already identified MCS peak.

Though the algorithm is resilient to minor scale and rotation perturbation of the query, severe changes in its pose require an altogether different strategy. Here, we handle two kinds of in-plane query distortions — scaling and rotation. In contrast to the setup of [6] we do not scale the target image features. Instead, we scale and rotate query features and leave the target features untransformed (for computational reasons). Once we obtain the MCS values, for all scales and orientation, at a particular sliding window location,

we select the right scale and orientation by doing maximum likelihood estimation (just keeping the maximum score) following [6].

2.4 Fast Detection with Efficient Matrix Cosine Similarity

To mitigate prohibitive computational load $\mathcal{O}(M \times N \times m \times n)$ owing to straightforward sliding window search, Seo et al. mostly relied on evaluating MCS on a sparse grid (coarse-to-fine) search [6], or saliency based pruning techniques [86] to aggressively reduce search space. Though valid and partly effective, such approximate search methods run the risk of missing detection peaks — a fact that often manifests itself as missed detection or as imprecisely located/oriented bounding box on the target image.

Besides pruning based approximate approaches (e.g., active learning in [87]) in the past, exact search of decision function maxima with branch and bound search schemes have also been investigated for object detection. Though branch and bound techniques [88, 89] are designed to converge to global maximum of decision function they are specifically designed for (bag-of-word style) histogram features, and it is not directly evident how to extend such frameworks to MCS based decision rule (the signs of feature elements can not be known a-priori to design sign based integral images [88, 89]).

Exact Acceleration of Matrix Cosine Similarity

Our intention remains going beyond the sliding window scheme to get rid of the $m \times n$ factor in the complexity of $\mathcal{O}(M \times N \times m \times n)$, and at the same time sticking to the exact computation of MCS. We proceed by first reshaping query feature $\mathbf{F}_Q \in \mathbb{R}^{d \times (mn)}$ to $m \times n \times d$ feature matrix, and in a similar fashion we reshape $\mathbf{F}_{T_i} \in \mathbb{R}^{d \times (MN)}$ to the size $M \times N \times d$. With slight notational abuse we retain the same nomenclatures for the reshaped query and

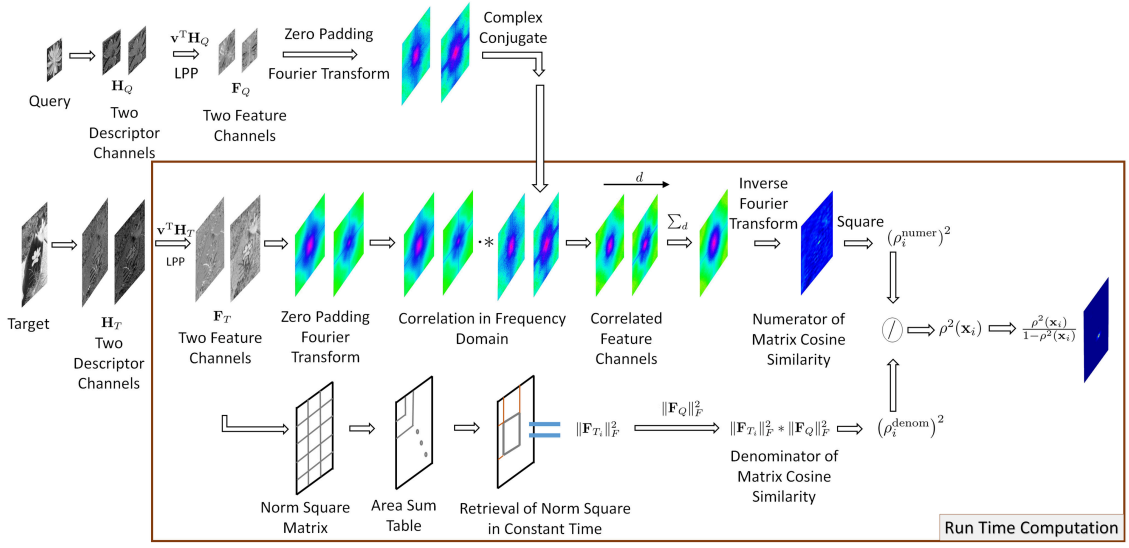


Figure 2.5: The illustration of the fast detection algorithm resulting into exact acceleration of matrix cosine similarity computation

target features. Paying this polite nod, we write the MCS expression (2.6) in the following fashion, noting that the numerator is (feature channel wise) cross-correlation between $\mathbf{F}_Q$ and $\mathbf{F}_{T_i}$ which can be efficiently computed in Fourier domain:

$$\begin{aligned}
&\rho(\mathbf{F}_Q, \mathbf{F}_{T_i}) \\
&= \sum_{c=1}^{d}\sum_{q=1}^{n}\sum_{p=1}^{m} \frac{\mathbf{F}_Q(p,q,c)\mathbf{F}_T(x_{i1}+p, x_{i2}+q, c)}{\|\mathbf{F}_Q\|_F \|\mathbf{F}_T(x_{i1}:x_{i1}+m, x_{i2}:x_{i2}+n, 1:d)\|_F}, && (2.8)\\
&= \frac{\sum_{c=1}^{d}\sum_{q=1}^{n}\sum_{p=1}^{m} \mathbf{F}_Q(p,q,c)\mathbf{F}_T(x_{i1}+p, x_{i2}+q, c)}{\|\mathbf{F}_Q\|_F \|\mathbf{F}_T(x_{i1}:x_{i1}+m, x_{i2}:x_{i2}+n, 1:d)\|_F}, && (2.9)\\
&= \frac{\text{IFT}\{\sum_{c=1}^{d} \overline{\text{FT}}\{\mathbf{F}_Q(:,:,c)\}\text{FT}\{\mathbf{F}_T(:,:,c)\}\}}{\|\mathbf{F}_Q\|_F \|\mathbf{F}_T(x_{i1}:x_{i1}+m, x_{i2}:x_{i2}+n, 1:d)\|_F}, && (2.10)
\end{aligned}$$

where $\text{FT}\{\cdot\}$, $\text{IFT}\{\cdot\}$, and $\overline{\text{FT}}\{\cdot\}$ denote Fourier transform, inverse Fourier transform, and conjugated Fourier transform respectively. Two important facts are worth mentioning at this point. First, correlation can directly be achieved by multiplying one Fourier transform

with another, conjugated. Second, since Fourier transform is a linear operator, one can perform the channel wise correlation right in frequency domain followed by channel wise summation [90].

However, two important distinctions with [90] exist in the proposed acceleration. First, we do not compute spatial correlation by converting it into an equivalent convolution problem in the frequency domain (by 180° rotation of query); correlation between two signals can be directly achieved in frequency domain by first taking Fourier transform of both signals, and then taking complex conjugate of just one of them followed by point by point multiplication (Hadamard product) in frequency domain. Second, MCS has also a normalization factor $\|\mathbf{F}_{T_i}\|$ in the denominator (2.6) requiring a different strategy for faster computation that Dubout *et al.* did not face in [90]. The trick to efficient computation of the normalization factor lies in a precomputed area sum table of target feature vectors' L_2 norm. Due to the presence of $\|\mathbf{F}_{T_i}\|$ in the denominator of MCS (2.6) one can not carry out the entire computation in Fourier domain. The target feature elements in $\|\mathbf{F}_T\|$ are individually squared and summed across all channels followed by an integral image construction. Next, one goes through this integral image and compute $\|\mathbf{F}_{T_i}\|^2$ in constant time with three arithmetic operations. This is followed by squaring and dividing the numerator by denominator to yield $f(\rho_i)$.

Indeed, a similar technique has found application in a somewhat dated but absolutely relevant work of J. P. Lewis [91]. However, Lewis used a different form of correlation [1] to consider, and the idea proposed in his work does not involve multichannel features, nor the multiscale and multioriented pattern detection. So, the methodology described in [91] is roughly comparable to a special case of a much more general framework proposed

[1]The idea of correlation manifests itself in various forms and definitions, and quite rightly there exists at least 13 distinct ways to look at the definition of correlation [92].

here when the number of feature channels reduces to one, and detection happens at single scale and orientation.

Fig. 2.5 shows the details of accelerated computation of MCS (at single scale and orientation). The descriptors are computed a priori, — from both query as well as target — and are treated as parts of the data set following the setup of Lampert [93]. We extract query features, scale and rotate them, do zero padding to bring each feature channel (matrix) to a pre-defined DFT size, apply forward Fourier transform on each feature channel followed by complex conjugation. However, for target T, the transformation on computed features is nil to keep the runtime cost at a bare minimum, only one forward Fourier transform in each feature channel is applied.

Implementation & Computational Time Analysis

To sweep $m \times n \times d$ query window over all locations of $M \times N \times d$ target array one requires to check $(M-m+1)(N-n+1)$ windows for potential objects. A direct evaluation of (2.6) requires, in each of such sliding windows, first, element by element product followed by summation (in each feature channel) for numerator giving us roughly $2dmn$ computations; and second, similar operations for norm in the denominator produces again (roughly) $2dmn$ computations. Combining the major components and considering total a configurations (equalling to the number of scales times number of orientations), we write the following total computation cost for sliding window scheme.

$$C_{\text{SW}} \approx 4d \cdot a \cdot (M-m+1) \cdot (N-n+1) \cdot m \cdot n. \tag{2.11}$$

Note here, operations like division and Lawley-Hotelling transformation $f(\rho_i) = \frac{\rho_i^2}{1-\rho_i^2}$ are of the order of $\mathcal{O}(MN)$, and hence negligible. Before we derive the exact compu-

tational cost for proposed fast detection methodology, we note that correlation performed by means of DFT is circular rather than linear, which we require. The difference lies in the fact that circular correlation is an aliased version of its linear counterpart. As long as the DFT matrix is large enough the resulting circular correlation will equal the linear correlation. This is ensured by padding each query feature channel $(m \times n)$ and corresponding target feature channel $(M \times N)$ with sufficient zeros so that zero padded arrays are at least as large as $(M+m-1) \times (N+n-1)$. We assume the zero padded DFT size is $(M_p \times N_p)$, where $M_p \geq M+m-1$, and $N_p \geq N+n-1$. It is also worthwhile to mention that a good practice is in keeping the DFT size at a power of 2 for leveraging the inherent efficiency of Fourier transform. Of course, with variable target size one can go with mixed-radix DFT. Now, a single forward/backward DFT involves computational cost $C_{\text{DFT}} \approx 2.5 M_p N_p \log_2(M_p N_p)$ as in [90]. We need d forward DFT for target plus one inverse DFT for each configuration of the query (Fig. 2.5). Hence, considering the cost of Hadamard product across all feature channels for all configurations ($C_{\text{prod}} = daM_pN_p$) followed by the cost of channel wise summation ($C_{\text{sum}} = daM_pN_p$), we write the total cost for numerator of (2.10) as,

$$\begin{aligned} C_{\text{numer}} &= dC_{\text{DFT}} + C_{\text{prod}} + C_{\text{sum}} + aC_{\text{DFT}}, \\ &\approx (d+a)2.5M_pN_p\log_2(M_pN_p) + 2adM_pN_p. \end{aligned} \tag{2.12}$$

Producing the norm squared integral image from target feature matrices requires time complexity $2dMN$, because each feature element is squared and summed over all d-channels. Retrieval of $\|\mathbf{F}_{T_i}\|^2$ corresponding to the sliding window location $\mathbf{x}_i$ in target T happens in constant time $\mathcal{O}(1)$ with only three arithmetic operations yielding $3MN$ cost. Next, squaring the numerator followed by the division by the product of $\|\mathbf{F}_{T_i}\|^2$ and the constant term $\|\mathbf{F}_Q\|$ are again three constant time operations per configuration. The construction of $f(\rho)$ involves another two constant operations (subtraction in denominator and division) per

configuration. Taking all this information into account, we end up with the following cost of denominator computation across all d feature channels,

$$\begin{aligned} C_{\text{denom}} &= 2dMN + 3MN + 3aMN + 2aMN, \\ &= (2d + 3 + 5a)MN. \end{aligned} \tag{2.13}$$

Considering the division of numerator by the denominator the total cost C_{proposed} of MCS computation boils down to —

$$\begin{aligned} C_{\text{proposed}} &= C_{\text{numer}} + C_{\text{denom}}, \\ &\approx 2.5(d + a)M_pN_p \log_2(M_pN_p) \\ &+ 2adM_pN_p + (2d + 3 + 5a)MN. \end{aligned} \tag{2.14}$$

The key observation here is that the proposed detection methodology has made the computational cost independent of the query size $m \times n$ for a *fixed* DFT size $M_p \times N_p$. Large computational mileage results from this fact especially when the query size changes as long as the maximum required DFT size is less than the fixed DFT size. There is further advantage when $d + a << ad$, i.e., with increasing number of query templates, and feature channels, one reaps increasing benefit in comparison to sliding window scheme. Indeed, Dubout *et al.* [90] have achieved almost 13 times theoretical speedup by leveraging Fourier transform in their part based detection process. In our setup, if we plugin typical values for the cost parameters in the final cost expression (2.14) we get the following result: for $m = 64, n = 64, M = 128, N = 128, a = 1, d = 5$ we get theoretical speedup 20, and for $M = 256, N = 256$ the speedup is 41. Table 2.1 gives some ideas of achievable speedup with our (unoptimized) implementation.

[6].

Figure 2.6: User defined object detection in the movie *Charade (1963)*: in leftmost column the user defined query object is highlighted, and example detections are displayed on right. Correct detection has been achieved with high resemblance value even in case of partial occlusion.

2.4.1 Accelerated Visual Search: Query Detection in Video

In the past sections we have derived a theoretical estimate of the runtime of our accelerated search technique. It is worthwhile to mention that for a fixed DFT size we achieve a runtime performance that is independent of the query size. Surely, setting the DFT size too high (by zero padding) to accommodate both query and target inside, may led to somewhat inefficient memory usage. There are various techniques to get around this difficulty. One can work with query and target of reduced sizes, or the more technically correct solution is to perform overlap-add and overlap-save methodology following a mixed-radix implementation.

Figure 2.7: Query object detection in movie *Dressed to Kill (1980)*: in top row, leftmost column, the user selects the *bow-tie* as query object, and sample detections are shown in right panels. In second row, leftmost column, the selected *biscuit jar* as query gets detected in subsequent frames in the middle of heavy clutter, scale change, and partial occlusion.

In our experiments we have presented results comparing the proposed fast object detection with sliding window based scheme [6]. The number of feature channels for evaluating MCS has been kept constant at five. Table 2.1 summarizes the runtime in seconds for single scale object detection using two queries of sizes 64×64 and 128×128. The *power-of-2* implementation assumes the smallest (power-of-2) DFT size as (M_p, N_p), large enough to hold query plus target sizes, i.e., $M_p \geq M + m - 1$, and $N_p \geq N + n - 1$. Consequently, a 64×64 query and 128×128 target should have a minimum (power-of-2) DFT size of 256×256, and a target 512×512 (or 768×768) should have a DFT size of 1024×1024. We also present runtime in seconds with mixed-radix implementation as part of of our results.

Figure 2.8: Detection results in movie *Ferris Bueller's Day Off (1986)*: in top row, leftmost column, the user selects the *wall painting* (within camera focus), and subsequent detections include cases with heavy out-of-focus instance and partial occlusions. In second row, the selected *jersey number* is detected against considerable geometric distortion. Lastly, in the third and fourth rows, we see the *red-wing logo* detected in a perfect manner on the T-Shirt despite some challenging distortions like scale, and even aspect ratio.

Clearly, the results show considerable performance gain rendering the real time object detection feasible. For multiscale search in Table 2.2, we have categorically used 10 scales, transforming the query features by 0.5 to 2.0 times the original query size (and compared with [6] who transform the target features). For joint multiscale and multiangle detection in Table 2.2, besides using 10 scales, we have checked 12 orientations (per scale) with equal angular spacing. Table 2.2 reports runtime based on mixed radix implementation of discrete Fourier transform.

All the experiments are done in a standard desktop machine with 8 GB RAM, Intel Core i7-2600 CPU @3.40 GHz using standard MATLAB functions with no GPU support. Of course, our proposed methodology is general enough to avail of the benefit of GPU computation which would result in even shorter computation time.

Table 2.1: Runtime of Proposed Fast Object Detection in Comparison with Sliding Window Scheme

Query Size (pixels)		64 × 64				128 × 128		
Target Size (pixels)		128 × 128	256 × 256	512 × 512	768 × 768	256 × 256	512 × 512	768 × 768
Sliding Window [6] (sec.)		0.3665	2.8815	15.6680	38.9581	5.4294	45.1551	132.1556
Proposed (in sec.)	Power-of-2	0.0304	0.0886	0.3296	0.3313	0.0929	0.3259	0.3341
	Mixed-radix	0.0184	0.0366	0.1360	0.2429	0.0540	0.1650	0.2492

Table 2.2: Runtime of Fast Object Detection with Pose Estimation in Comparison with Sliding Window Scheme

Pose estimation for Query size 64 × 64 pixels		Different Target Sizes (in pixels)			
		128 × 128	256 × 256	512 × 512	768 × 768
Multiscale Search Time	Sliding Window [6] (in sec.)	4.607	34.341	182.475	448.740
	Proposed (in sec.)	0.116	0.380	1.718	2.248
Multiscale, Multiangle Search Time	Sliding Window [6] (in sec.)	53.255	398.788	2140.610	5145.767
	Proposed (in sec.)	1.317	4.068	20.142	24.756

Inspired by the famous project *Video Google* by Sivic and Zisserman [94], and later studied by Lampert [93], we have extended our work to user-defined query detection in movies. Before we go into the experimental details, we point out some novel features of our approach in comparison with previous approaches to *Video Google*. First, our methodology does not require the overhead of bulk codebook creation. The user is free from the tedious task of feature quantization for building a visually descriptive dictionary. Secondly, the proposed detector robustly deals with in-plane variation (i.e., change in scale and orientation), handling extreme clutter, low resolution as well as partial occlusion. We have carried out our experiment on three movie data sets, namely, *Charade (1963)*, *Dressed to Kill (1980)*, and *Ferris Bueller's Day Off (1986)*. The first two movies come with gray scale frames and the last one with color frames. We have processed the following frame sizes for the above movies: 312×240, 320×240, and 416×170. The number of feature channels used is five in number, and we have used FDR $\alpha = 1\%$ to achieve the detection results as shown in Fig. 2.6, 2.7, and 2.8. It is reasonable here to search for the query at 5 scales, 0.8, 0.9, 1.0, 1.1, and 1.2 times the size of the query image selected by the user. We do not consider multioriented detection in this experiment. With the present set of queries we have achieved following detection rates for the three movies: 97.21%, 92.05%, and 88.66%, respectively, as opposed to 93.17%, 84.88%, and 82.25% by [6]. The missed detections result when the query suffers severe off-the-plane distortions resulting in major viewpoint alteration. The proposed method is able to detect queries amidst major in-plane distortions, like significant scale change, partial occlusion, out-of-focus blur, and low resolution. The average time consumed per frame for all the three movies are as follows: 0.131 sec, 0.150 sec, and 0.122 sec, as opposed to 9.581 sec, 11.622 sec, and 10.210 sec by [6].

It is true that codebook-based approaches [94, 89] consume much shorter run-

time to process movie frames but two important distinctions exist here. First, our work is training-free and we don't require the user to build a codebook. Secondly, the codebook based approaches [93] do not process the movie frames in linear fashion. In contrast, we are interested in exact search, and hence, the present methodology processes the frames successively in linear sequence; our task is motivated by the long term goal of real-time object detection with smart phone cameras or mobile devices when the frames may not be available a-priori.

Chapter 3

Support Tensor Machines: Pedestrian Detection in Thermal Infrared Images

Abstract – Pedestrian detection in thermal infrared images poses some unique challenges because of the low resolution and noisy nature of the images. Here we propose a methodology to use Local Steering Kernel (LSK) tensors as low-level descriptors for detecting pedestrians in far-infrared images. LSK [6] is specifically designed to deal with intrinsic image noise and pixel level uncertainty by capturing local manifold *geometry* succinctly instead of collecting local orientation *statistics* (e.g., histograms in HOG). Our second contribution is the introduction of a new image similarity kernel in the popular max-margin framework of support vector machines that results in a relatively short and simple training phase for building a rigid template as pedestrian detector. Our third and last contribution is to replace the sluggish but *de facto* sliding window based detection methodology with multichannel discrete Fourier transform, facilitating very fast and efficient pedestrian localization. Our experimental studies on publicly available thermal infrared images justify

our proposals and model assumptions.

3.1 Introduction

The computer vision community has made good progress in people and pedestrian detection in natural images and videos in last decades. However, such endeavors in locating pedestrians has mostly been restricted to photographs captured with visible range sensors [95, 52]. Infrared and thermal imaging sensors which provide excellent visible cues in unconventional settings, e.g., night time visibility, have historically found their use in military, security and medical applications. However, with increasing image quality and decreasing price and size, some of the sensing devices are finding commercial deployment for home and office monitoring and automotive applications [96, 97]. Research effort has so far been limited in this domain for building reliable and efficient computer vision systems for infrared thermal image sensors. The objective of this paper is to address this concern. In this work we aim at pushing the state of the art by taking into account the limitation of the current approaches as well as particular characteristics of infrared images.

Thermal image sensors typically have a spectral sensitivity ranging from 7 microns to 14 microns band of wavelength. The capacity of these imaging devices to appropriately capture images of objects depends on their emissivity and reflectivity in a non-trivial fashion. The material and surface properties control emissivity whereas amount of background radiation reflected by the objects influence their reflectivity. Such uncertainties often lead to various distortions in thermal images, notably, *halo effect, hotspot areas, radiometric distortions* to name a few. Fig. 3.1 illustrates halo effect in a natural-thermal image pair. Also noticeable is the fact that textures visible on objects often get suppressed in thermal images. This fact has important bearing as far as visual recognition is concerned

because the negative support vectors corresponding to background tend to be far less descriptive. Fig. 3.1 also illustrates the challenge involved in detecting foreground objects because of inherently noisy nature of the infrared images. From the representative images it is fair to conclude that a successful visual recognition system must include a robust noise handling component in the feature computation as well as a strong measure of visual similarity that can reduce missed detections as well as false positives per image.

In this chapter we focus our attention on detecting pedestrians, particularly walking at a distance from the camera. There can be a number of experimental settings in which this problem can be studied. One setup can consist of images containing pedestrians against arbitrary background. This is reminiscent of the popular INRIA pedestrian data set with visible range sensors. One usually approaches this problem by building a pedestrian specific classifier and locating the pedestrians in the test set of images. Another set up could be a sequence of images captured as video while the sampling rate of video frames is non uniform. In such case one can assume a background model, but non uniform sampling rate disallows the tracking possibility of the detected pedestrians. The next obvious setting is of course uniformly sampled video sequence where both background model and tracking of humans can be incorporated in parallel to the detection task of humans. Insofar we have assumed a stationary camera for capturing the images. However, the most relevant scenario for automotive applications would be to capture images from a moving vehicle. However, since there are not too many such data sets available at present, we focus our attention and study on building pedestrian detectors with stationary camera.

The low resolution and noisy nature of thermal infrared images offer novel set of challenges to the detector [97]. Training a detector with noisy label information is a popular

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

Figure 3.1: Infrared images are different: natural color images exhibit textures which are suppressed in infrared images (left pair images). As a consequence, many background texture features like trees and buildings may remain relatively nondescriptive (third from left) which complicates the separation of the background in feature space during the learning process. In addition, the high noise adds to the complexity of detecting foreground objects (far right).

research area in computer vision, but dealing with heavy noise and artifacts in image signal while performing visual recognition tasks has garnered relatively low attention from the community. This is particularly relevant in infrared domain where sensor noise is high, and feature variability is much less compared to natural photographs. Our objective, like that of HOG (that is widely used in state of the art pedestrian detectors [15] as low-level features), is to capture local structure in a stable fashion. Geometric invariance (in HOG [34]) and scale invariance (in SIFT [9]) are relaxed at the expense of robustness of the descriptor to deal with high noise level. For accomplishing that purpose we advocate the use of Local Steering Kernel (LSK) [72, 6, 71] as low level image region descriptor. LSK lies at the heart of Locally Adaptive Regression Kernel or LARK and has been primarily used in image denoising tasks (we shall follow the name LSK since it intuitively makes more sense indicating characteristics of our features). HOG and LSK both capture local orientation information; however, HOG computes orientation *statistics* over local image surface, whereas LSK captures local orientation geometry and is thus more stable in dealing with

image noise. It is worth mentioning that LSK has its origin in image filtering where it is successfully applied to various image restoration tasks like denoising and deblurring.

Past work has highlighted the effective use of matrix cosine similarity as a robust measure for computing image similarity [6, 79]. Motivated by this result we have extended matrix cosine similarity to introduce a max-margin training formulation for learning a decision boundary that can separate pedestrian from the background. The standard technique for object search, i.e., sliding window based object detection, incur prohibitive computational cost. To resolve this issue Lampert *et al.* have proposed a branch and bound technique [88, 89]. In our work, we propose a relatively simple but efficient technique to improve the runtime search for pedestrians performing multiscale object detection in less than a second. We proceed with the system overview in the next section.

3.2 System Overview

Unlike color images that contain multiple color channels, infrared images typically have a single channel that we denote by an $M \times N$ image matrix I. From a $m \times n$ cropped image window (of pedestrian/ background) we densely compute low-level (l-dimensional) descriptors $\mathbf{H} \in \mathbb{R}^{l \times (mn)}$. After dimension reduction of $\mathbf{H}$ by PCA we obtain the decorrelated features $\mathbf{F}$. The features with class labels are used as input to support vector machines (SVM) using a (proposed) linear kernel in order to learn a decision boundary to separate pedestrian from background in the feature space. Fig. 3.2 illustrates how the support vectors are combined to build the rigid detector. The search for pedestrians in a test image proceeds in the frequency domain (using Fourier transform) with integral image based normalization, yielding an elegant framework for extremely efficient and fast pedestrian detection.

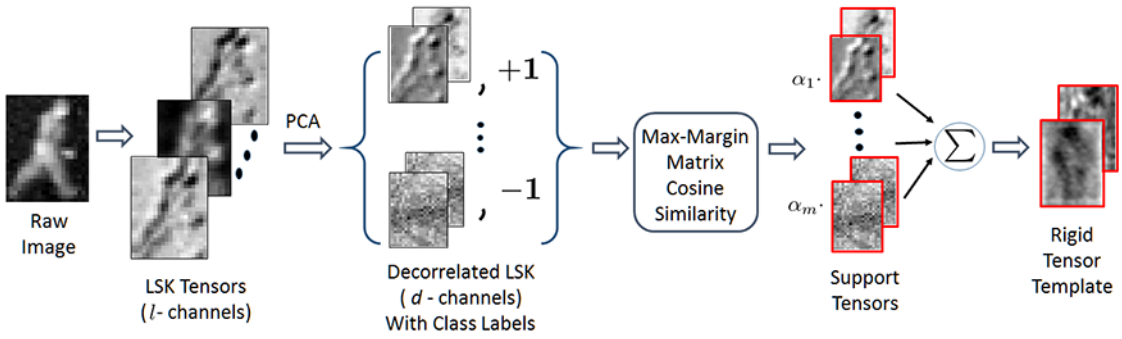


Figure 3.2: LSK tensor descriptors are projected onto leading principal components to yield decorrelated and discriminatory feature tensors which are then used in a max-margin training framework with matrix cosine similarity kernel. Owing to the linearity of the kernel, support vectors are combined into a rigid detector for fast and efficient detection.

3.3 Local Structure Estimation with LSK

Computing all LSK coefficients of pixel $\mathbf{x}_i$ with its neighbors over a $p \times p$ local window leads to the descriptor vector $\mathbf{h}_i$ defined by $\mathbf{h}_i = (h_{i1}, h_{i2}, \ldots, h_{ij}, \ldots, h_{p^2})$. The descriptor vectors $\mathbf{h}_i$ are stacked column wise to define the descriptor matrix as $\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_{mn}] \in \mathbb{R}^{l \times (mn)}$. Dense computation makes the descriptor rich in information but at the same time somewhat redundant (Fig. 3.3). To distill the redundancy we project the high dimensional descriptor $\mathbf{H}$ onto leading principal components. Doing this has two imminent benefits: one, the projected features are clean, sparse and prominent, and two, reducing the dimension of feature vectors helps us achieve faster detection in runtime involving less computations.

We collect the set of d eigenvectors as columns of $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_d] \in \mathbb{R}^{l \times d}$. We choose the number of eigenvectors d in such a way that 60% of the energy is contained in terms of the eigenvalues λ as follows: $d = \arg\min_i \frac{\lambda_i}{\sum_j \lambda_j} > 60\%$. As a result of the projection

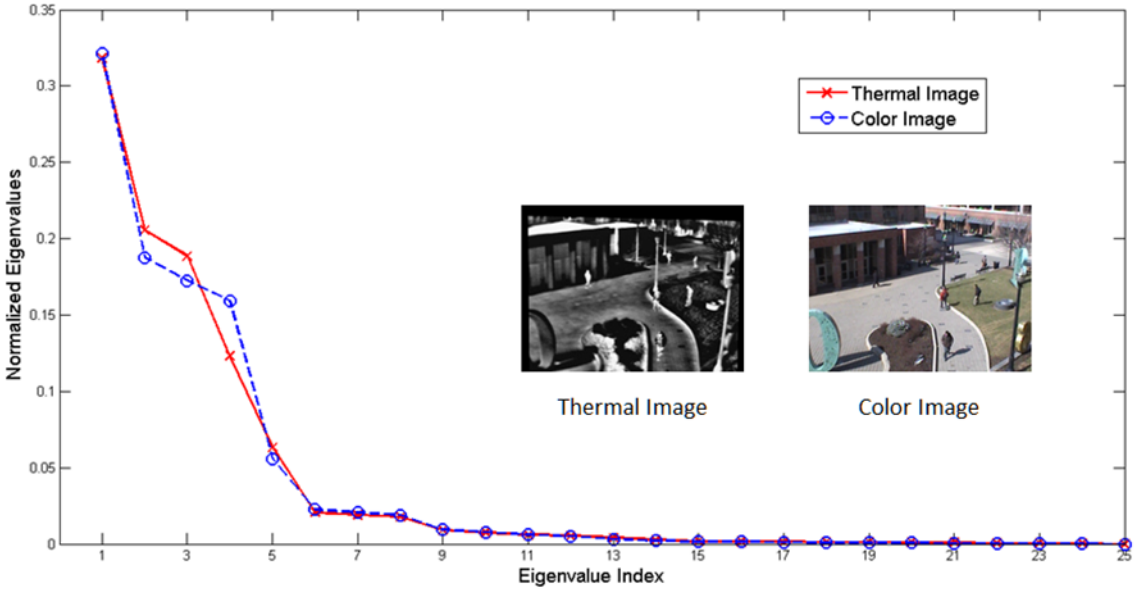


Figure 3.3: LSK descriptors belong to a low dimensional manifold where 70% to 80% of the energy of the eigenvalues is contained in first three or four of them.

we obtain discriminative features $\mathbf{F} \in \mathbb{R}^{d \times (mn)}$, where $d << l$, given by the following:

$$\mathbf{F} = \mathbf{V}^{\mathrm{T}}\mathbf{H}. \tag{3.1}$$

3.4 Design of Linear Detector

In the context of one shot object detection, for computing similarity between two images (of same size), Seo *et al.* [6] and Biswas *et al.* [79] have advocated the use of matrix cosine similarity (MCS). This measure of image similarity is in fact a generalization of cosine similarity from vector features to matrix features. In principle, if $\mathbf{F}_Q$ is a query feature matrix that we try to find in a bigger target image T in a sliding window fashion,

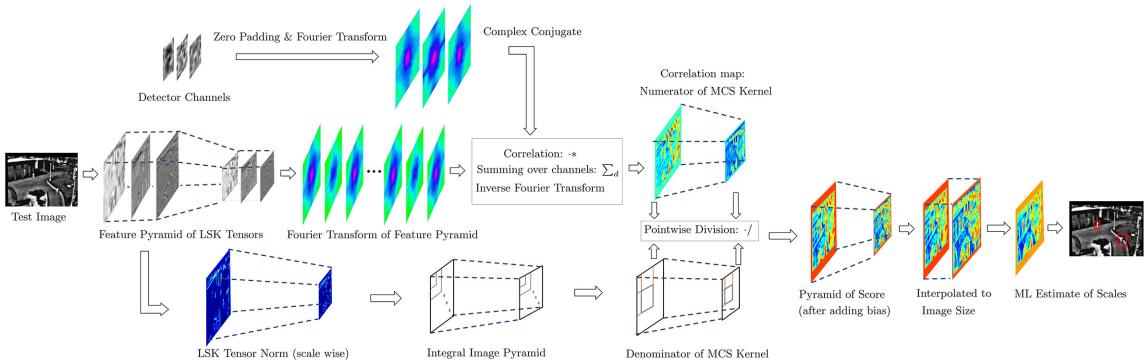


Figure 3.4: Multiscale detection technique involving construction of feature pyramid, computation of kernel function and maximum likelihood estimate of scale and location of pedestrian in target image

then at each position $\mathbf{x}_i$ of the sliding window over T we compute MCS (ρ) as follows:

$$\rho(\mathbf{F}_Q, \mathbf{F}_{T_i}) = \frac{\text{trace}(\mathbf{F}_Q^T \, \mathbf{F}_{T_i})}{\|\mathbf{F}_Q\| \|\mathbf{F}_{T_i}\|} \in [-1, 1]\,. \tag{3.2}$$

Higher the value of the MCS at location $\mathbf{x}_i$ in target image, greater is the likelihood of finding the object there.

3.4.1 Kernelization of matrix cosine similarity

The computation of MCS involves channel wise correlation followed by normalization. The following derivation shows MCS as a dot product between two normalized vectors. The dot product formulation will be the key to kernelizing MCS in the subsequent part of this section, and also to introducing the max-margin framework for designing the pedestrian later in this paper. To show that MCS represents a dot product we first reshape the feature matrices $\mathbf{F}_Q$ to $m \times n \times d$ and $\mathbf{F}_T$ to $M \times N \times d$. Paying a polite nod to this slight notational abuse, and noting that each i-th sliding window feature tensor $\mathbf{F}_{T_i} \in \mathbb{R}^{m \times n \times d}$, we

proceed to reframe the expression of MCS in (3.2) as follows —

$$\rho(\mathbf{F}_Q, \mathbf{F}_{T_i}) = \sum_d \sum_n \sum_m \frac{\mathbf{F}_Q(m,n,d)}{\|\mathbf{F}_Q\|} . \frac{\mathbf{F}_{T_i}(m,n,d)}{\|\mathbf{F}_{T_i}\|}, \tag{3.3}$$

$$= \langle \mathbf{F}_Q^\dagger, \mathbf{F}_{T_i}^\dagger \rangle, \tag{3.4}$$

$$= \text{colstack}(\mathbf{F}_Q^\dagger)' \ \text{colstack}(\mathbf{F}_{T_i}^\dagger), \tag{3.5}$$

$$= k(\tilde{\mathbf{F}}_Q, \tilde{\mathbf{F}}_{T_i}). \tag{3.6}$$

We have assumed $\mathbf{F}_Q^\dagger = \frac{\mathbf{F}_Q}{\|\mathbf{F}_Q\|}$, $\mathbf{F}_{T_i}^\dagger = \frac{\mathbf{F}_{T_i}}{\|\mathbf{F}_{T_i}\|}$, and the `colstack(·)` merely arranges the elements of the matrices $\mathbf{F}_Q^\dagger$ and $\mathbf{F}_{T_i}^\dagger$ in lexicographic fashion in a long column vector yielding $\tilde{\mathbf{F}}_Q$ and $\tilde{\mathbf{F}}_{T_i}$ respectively. In other words, we have the following property to hold true: $\tilde{\mathbf{F}}_Q, \tilde{\mathbf{F}}_{T_i} \in \mathbb{R}^{(m \cdot n \cdot d)}$.

3.4.2 Max-Margin Formulation with MCS kernel

With the kernel interpretation of MCS (3.6) let us introduce the max-margin framework in the present context. We follow the convention and label the features represented by $\tilde{\mathbf{F}}_j$ with $y_i \in \{-1, 1\}$ (pedestrain/background). Next, we use the C-SVM formulation to find the hyperplane $\mathbf{u}$ that best separates pedestrian from the background:

$$\begin{aligned} &\underset{\mathbf{u}, b, \psi_i}{\text{minimize}} \ \frac{1}{2}\|\mathbf{u}\|^2 + C \sum_{i=1}^{N} \xi_i, \\ &\text{subject to } \ y_i \ (\mathbf{u}'\tilde{\mathbf{F}}_i + b) \geq 1 - \xi_i, \xi_i \geq 0, \end{aligned} \tag{3.7}$$

where, $C > 0$ is a tradeoff between regularizations and constraint violation. Using the **Representer theorem** [98] we can always write a minimizing solution of (3.7) as $\mathbf{u}^* = \sum_{j=1}^{q} y_j \alpha_j \tilde{\mathbf{F}}_j$ that we insert in (3.7), and optimizing over $\alpha_1, \alpha_2, \ldots, \alpha_q \in \mathbb{R}^q$ instead of $\mathbf{u} \in \mathbb{R}^{(m \cdot n \cdot d)}$, we arrive at the following decision function:

$$f(\mathbf{F}_{T_i}) = \text{sgn}(\sum_{j=1}^{q} y_j \alpha_j k(\tilde{\mathbf{F}}_{T_i}, \tilde{\mathbf{F}}_j) + b). \tag{3.8}$$

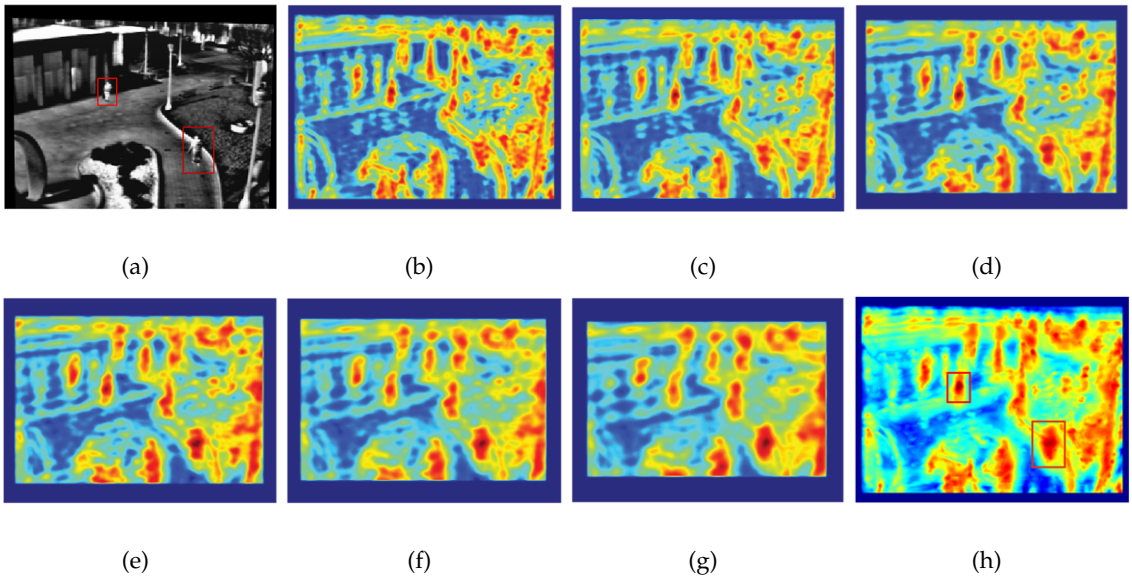


Figure 3.5: **Scale Estimation in Multiscale Detection**: Following detection, each scale of features in the feature pyramid yields a likelihood map showing detection score (b)-(g). Note, the boundary region in likelihood map is getting wider (filled with zeros) with scales because the rigid detector is getting bigger relative to target with decreasing target size. The individual likelihood maps of various sizes are rescaled with bilinear interpolation to a common size (shown here). Lastly, a maximum likelihood estimate at each pixel is carried out from all likelihood maps to obtain the final likelihood map (h) which upon thresholding and non-maximal suppression yields pedestrian location. The likelihood map supplying the maximum score becomes the scale associated with the detected bounding box. Blue means low score and dark red to reddish black denote high scores.

In short, q-kernel computations are needed to classify a point with a kernelized SVM and all of q-support vectors. Furthermore, simplification of (3.8) leads us to the following forms of the decision function:

$$f(\mathbf{F}_{T_i}) = \text{sgn}(\sum_{j=1}^{q} y_j \alpha_j \langle \mathbf{F}^{\dagger}_{T_i}, \mathbf{F}^{\dagger}_{j} \rangle + b), \tag{3.9}$$

$$= \text{sgn}(\langle \mathbf{F}^{\dagger}_{T_i}, \sum_{j=1}^{q} y_j \alpha_j \mathbf{F}^{\dagger}_{j} \rangle + b), \tag{3.10}$$

$$= \text{sgn}(\langle \mathbf{F}^{\dagger}_{T_i}, \bar{\mathbf{F}} \rangle + b). \tag{3.11}$$

The first step results from (3.4), second step results by virtue of linearity of the kernel, and third step designates the weighted sum of support vectors as $\overline{\mathbf{F}}$.

We conclude this section by drawing the attention of the reader to the fact that $\overline{\mathbf{F}} \in \mathbb{R}^{(m \cdot n \cdot d)}$ is our estimated detector that can be reshaped back to $m \times n \times d$ feature tensor. Searching this rigid detector in a bigger target T by repeated evaluation of the decision rule (3.4) is a computationally intensive task. In the next section we propose an efficient and faster means of accomplishing this objective.

3.5 Beyond Sliding Window: Efficient Pedestrian Search

A single channel sliding window of $m \times n$ searched in $M \times N$ target incurs a computational cost $\mathcal{O}(mnMN)$, and with d-channels $\mathcal{O}(dmnMN)$. Sliding window based computation of decision rule in the form of (3.10) results in time complexity $(O)(q^2T^2dm)$. Owing to the linearity of the kernel one can precompute the detector by combing all support vectors as shown in equation (3.11), reducing the runtime cost to $(O)(q^2T^2d)$.

Computation of (3.11) basically involves repeated execution of (MCS kernel) all through the target. A careful investigation reveals that numerator of MCS in (3.2) is a channelwise cross-correlation followed by summation across channels. For faster detection we precompute the Fourier transform of detector template. During runtime we perform Fourier transform of each target feature channel carrying out cross correlation in frequency domain, followed by summation across channel also in frequency domain (possible owing to the linearity of Fourier transform), and then reverting the summand back in spatial domain by inverse Fourier transform (also suggested in [90] recently). The whole process takes a total time of the order of $\mathcal{O}(dMN\log MN) + \mathcal{O}(dMN) + \mathcal{O}(dMN) + \mathcal{O}(MN\log MN) = \mathcal{O}(dMN\log MN)$. The last step in (3.11) involves normalization by $\|F_{T_i}\|^2$ which could be

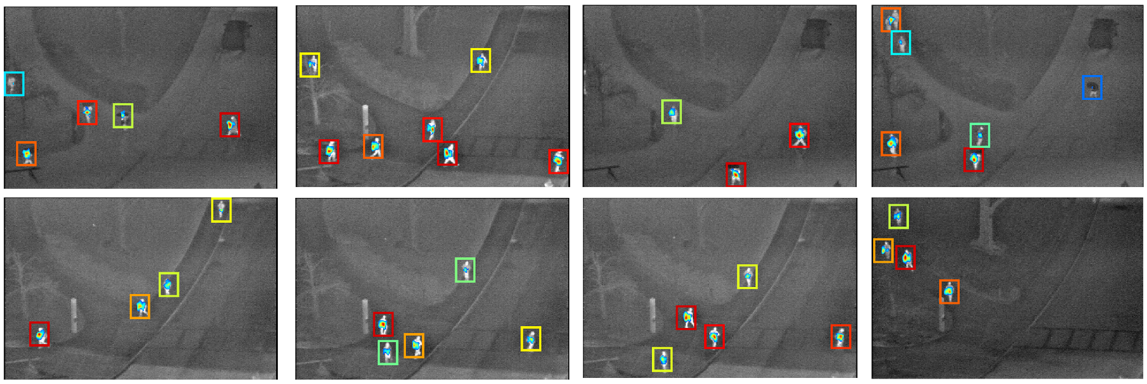


Figure 3.6: Detection results of our proposed methodology on OSU thermal dataset are shown in this figure at different times. The detection scores above the threshold are embedded inside the displayed bounding box. The convention of color map is maintained, i.e., a red bounding box indicates highest confidence and blue bounding box lowest confidence.

performed efficiently by precomputing an integral image of $\|F_{T_i}\|^2$ with time $\mathcal{O}(dMN)$, and retrieval of the normalization factor in constant time operation per window. Hence, the overall runtime cost maintains an order of $\mathcal{O}(dMN \log MN)$, in contrast to $\mathcal{O}(dMNmn)$ time in sliding window based technique. This is a substantial progress toward efficiency because the cost no longer relies on the rigid detector's template size! The complete methodology for evaluating the proposed MCS as a detection step is illustrated in details in Fig. 3.4.

3.6 Multiscale Detection Methodology

There are two approaches generally available for multiscale search of objects. One approach is to scale up the rigid detector and search for maximum scoring region. Such an approach is an attractive option because target image undergoes minimum transformation during runtime. However, from a purely theoretical standpoint this scaling up of a rigid

detector can have the uncanny effect of introducing undesired bias. It is not immediately obvious how and to what extent such issues will manifest in the present framework, and in case they do what could be probable way outs to mitigate such limitation. Hence, we have followed the second approach that involves target rescaling. Precisely, we did not rescale the target images as such, we computed features from the given image and resorted to feature scaling over the desired range of scales. In essence, we have constructed a feature pyramid of decreasing image size as described in Fig. 3.4.

The use of tensor features permits ready application of discrete Fourier transform and integral image techniques. Such techniques make the search of the root filter in multiple scales of target image very quick to compute without sacrificing much efficiency during runtime. We describe next how we infer the pedestrian's location and the scale of the detected bounding box in the following. It is important to note that for each scale we essentially obtain a likelihood map [1] as a result of detection (Fig. 3.5). Each pixel intensity in the likelihood map represents the value of scoring function of the proposed linear classifier at that particular scale. We rescale the likelihood maps of all scales to bring them to a common size (the largest scale in our case) before doing a maximum likelihood estimate at each pixel to choose the scale that is producing the maximum detection score. The final likelihood map is thresholded (usually at zero) following a non-maximum suppression step to output the pedestrian's location. The maximum scale associated with the pedestrian's location provides the size of the bounding box we need.

[1] not statistical likelihood per se, here we use the term *likelihood* to indicate the confidence of the detector in terms of the score

3.7 Experiments and Results

Images captured through thermal infrared cameras depend on various factors ranging from target surface to the nature of thermal imagers. Given such long list of dependencies it is reasonable to expect good number of thermal infrared datasets for the study and development of computer vision algorithms. However, despite the motivation and requirement the number of thermal image datasets is still not as many, and the size of the datasets where available is relatively small (osu-T). Moreover, some of the the baseline datasets which are typically used in developing and benchmarking algorithms lack a well established, common ground truth shared among the vision community (OSU-CT). It is prudent to remark here that the ready availability of such useful resources have facilitated steady improvement in the performance of vision algorithms for natural images (CalTech, INRIA pedestrians).

To mitigate this shortcoming new and large-scale thermal image datasets have been developed recently, e.g., LSI [99] and KAIST [100] and BU-TIV [101]. The image quality in LSI is reasonably decent and height of pedestrians range from 20 to 120 pixels, offering a good range of difficulty levels to develop pedestrian detectors. KAIST multispectral images are taken in real life setting. The image quality in thermal channel tends to degrade quickly with distance and it gets difficult to distinguish distant pedestrians from background with the infrared channel alone (KAIST images come with RGB color channels too). It would be an interesting and rewarding future direction to extend the proposed methodology toward leveraging the information present in thermal infrared and color channels jointly. A very useful benchmark study on KAIST multispectral video (mostly on ACF channels and its variants with boosting as learning methodology) is described in the paper [100]. The BU-TIV dataset has relatively high resolution images of human be-

ings for a wide range of visual recognition tasks like detection, single view and multi-view tracking. The objects for the detection task in the BU-TIV not only include pedestrians but also other classes like cars and bikes on a crowded street.

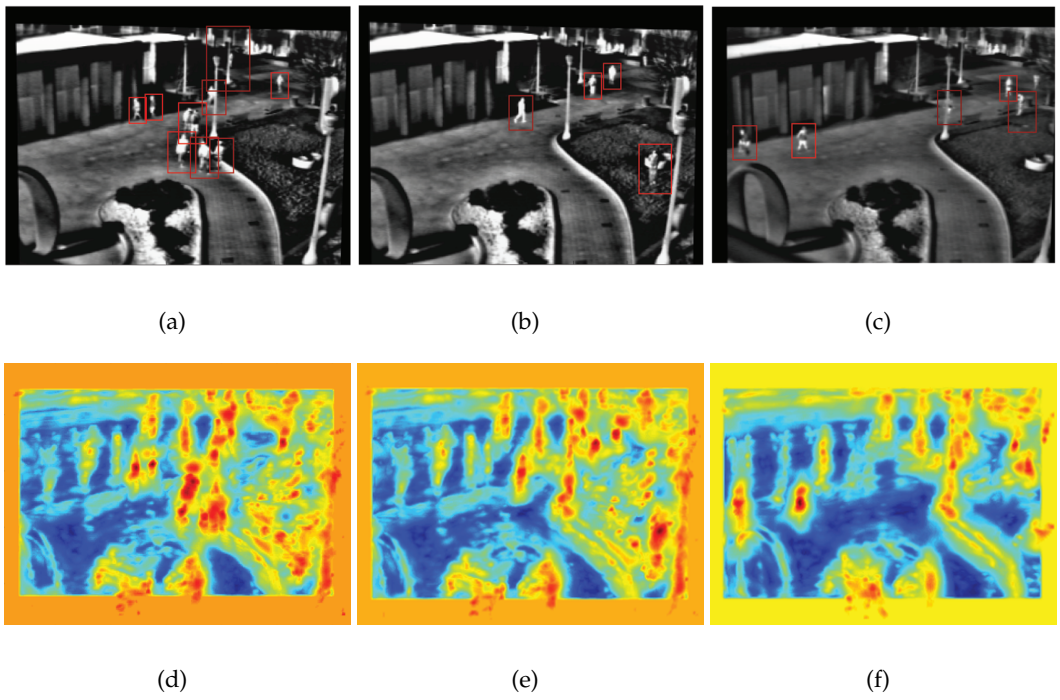
In this work we focus our attention on detecting pedestrians in three baseline datasets: OSU Thermal (OSU-T) [102], OSU Color Thermal (OSU-CT) [103], and LSI [99]. We restrict our study to thermal channels only. Color channels when available are ignored. The baseline dataset OSU-T contains pedestrians in still images. The other two datasets namely OSU-CT and LSI both are infrared video datasets. In addition to reporting results on the OSU-CT, we are going to release annotations of OSU-CT for the purpose of sharing with the community. To the best of our knowledge such ground truth for the study of pedestrian detection in OSU-CT is hitherto absent.

In the experimental setup we have first computed the three dimensional LSK descriptors where third dimension denotes number of descriptor channels. The number of descriptor channels is mostly 25 since we have considered 5×5 neighborhood around the central pixel. The LSK descriptors are projected on leading eigenspace (computed from full image), yielding again three dimensional LSK tensor features. The number of eigenvectors is typically three leading to height $\times$ width $\times$ channels tensors for each thermal image. Following axis aligned ground truth we extract feature tensors corresponding to each pedestrians. For training purpose we convert the feature tensor in a long column vector and write to a binary file with corresponding label (1 for pedestrians and -1 for background). It is important to remark that the column vectors, prior to writing into a binary file, are L2-normalized for the sake of our chosen kernel matrix cosine similarity. The feature computation is done once, and subsequently the computed features are used in training and hard mining with various parameter settings of the training algorithm.

For efficiency reason we extract the LSK descriptors with an elementary C-mexfile implementation. In the detection step simple MATLAB routines like built-in DFT functions are enough to shrink the training time to a considerable extent.

It is important to note that the algorithm proposed here as well as our runtime implementation do not assume any background model, nor any motion estimation for tracking. While such information can improve or even augment the space of probable inferences, we discard such domain specific information to evaluate the pedestrian detector with minimal supervision possible.

Our experimental framework follows the PASCAL VOC convention and we have used the development kit available on the website of PASCAL Visual Recognition Challenge. For describing evaluation criterion let us call the detection made by our algorithm as correct if DB (detected bounding box) overlaps sufficiently (must exceed 50%) with GB (ground truth): $\frac{\text{area}(DB \cap GB)}{\text{area}(DB \cup GB)} > 0.5$. As per PASACL evaluation guideline the highest confidence score of detection is matched first, and if the detector supplies multiple boxes to match with a single ground truth, the match with greatest overlap is considered in the evaluation process. An arbitrary tie breaker can be used in the rare event of the two bounding boxes overlapping with the ground truth by the same amount. As far as the measure for evaluating pedestrian detectors is concerned, such measure has gone through some recent modifications in the vision literature. Studying the trade off between missed detection and false positives has remained the well agreed way of describing detection performance. However, the vision community faced some issues with the way to normalize such measure. Going by most recent convention we define the miss rate by $FN/(TP+FN)$, and FP is normalized by total number of images in the data set leading to the quantity false positive per image (FPPI) [52]. This is in contrast to earlier notion of false positive per window



(a) (b) (c)

(d) (e) (f)

Figure 3.7: Top row shows multiscale detection on three frames from OSU-CT dataset. The scale best estimated is shown with the appropriate sized bounding box centered at the predicted location. The bottom row heat maps illustrate corresponding decision scores (maximum likelihood estimate across all six scales) obtained from classifier. The blue regions show less confidence and the red to reddish black shows high to very high confidence in detecting pedestrians. Note how well the proposed detector has managed to detect partially occluded people with reasonably fair accuracy without considering any sort of tracking information and/or background model.

(FPPW) as used to evaluate the pedestrian detector [34, 97]. Miss rate versus FPPI is also in contrast to precision and recall curves that are traditionally used in other applications of object detections. The reason is that in applications like autonomous driving, it is often the norm to fix the upper ceiling on acceptable FPPI rate independent of number of pedestrians in the image.

OSU Thermal Database (OSU-T): OSU thermal images [102] come from 10 se-

quences with a total number of 284 images all of which are 8-bit. The images are captured with a 75 mm lens camera mounted on rooftop of 8-story building with manual focusing. This dataset is not a video sequence because the images are captured in a non uniform fashion with a sampling rate less than 30Hz. The image size is 360 × 240 pixels. In total, the dataset has 984 pedestrians across all 10 sequences.

We have pooled images from all sequences and divided them uniformly for training and testing purpose. Such uniform distribution ensures proper variability both in training and test set. It is important because the images are captured in different weather condition including light rain. In total, 192 images are used for training and 92 for testing. There are 675 pedestrians (i.e., bounding boxes) in training set and 309 in test set.

It is common in wide area surveillance and satellite imaging to attribute less emphasis to scale. Objects at a distance do not appear to vary widely in sizes. Indeed, with a single scale rigid detector we have achieved reasonably good performance as shown in the Fig. 3.6. Note that the ground truth supplied varies in height and width across pedestrians. However, we have extracted a constant height × width bounding box (36 × 28 to be specific) around the center of the given ground truth rectangle for each pedestrian.

Since, the background does not change much we restrict the collection of hard negatives to a few images. An exhaustive search over full dataset will result repetitive entries of similar background structures which will harm the learning process. Doing this may introduce a bias with respect to the negative support vectors that may prove detrimental for the generalization purpose.

OSU Color Thermal Database (OSU-CT): OSU Color Thermal dataset [103] has a total of 17089 images (8-bit thermal and 24-bit color) and is a video sequence dataset. This dataset has a total of six sequences: the first three sequences have one background

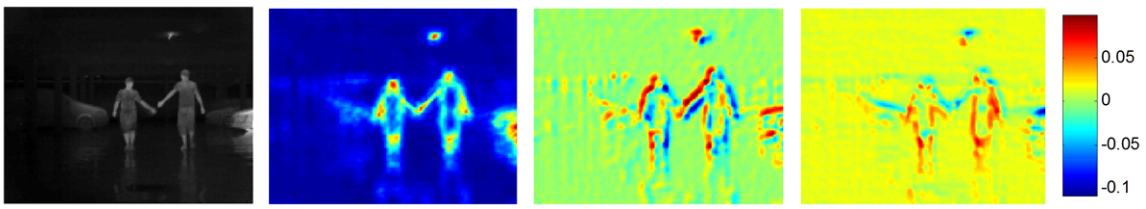


Figure 3.8: The thermal image on far left is shown in three LSK feature channels on right. Note how the first channel shows signal strength around body silhouette, whereas second channel tends to highlight horizontal to oblique structures. The third channel mostly models the vertical to near vertical structures.

pattern and the last three have a different background. Each of the six sequences has thermal as well as color channels. Since this dataset does not have a ground truth established for detection purpose, we have annotated the full data set for the evaluation of the proposed detector. From a purely pedestrian detection perspective (without taking into account background modeling and/or tracking algorithms) this dataset poses considerable challenge for annotation as well as detection task. The pedestrians seen from a distance often get occluded by physical structures or other pedestrians. We have approximated the annotated bounding box size to the best guess possible in such instance.

Since our objective is pedestrian detection in thermal infrared images we have discarded information from color channels and used only the infrared ones in the experiments. Similar to OSU-T, we have pooled images from all sequences and built our training and test set. Specifically, we have chosen every third image for test purpose and allotted the rest for the purpose of training. It is noted that the heights of pedestrians range from 14 to 60 pixels. In this case we have followed a multi-scale pedestrian detection approach with six scales altogether. We have learnt a rigid detector of size 24×16 and searched the target images over the following scales: 1.00, 1.30, 1.60, 1.90, 2.20, 2.50. The results of our

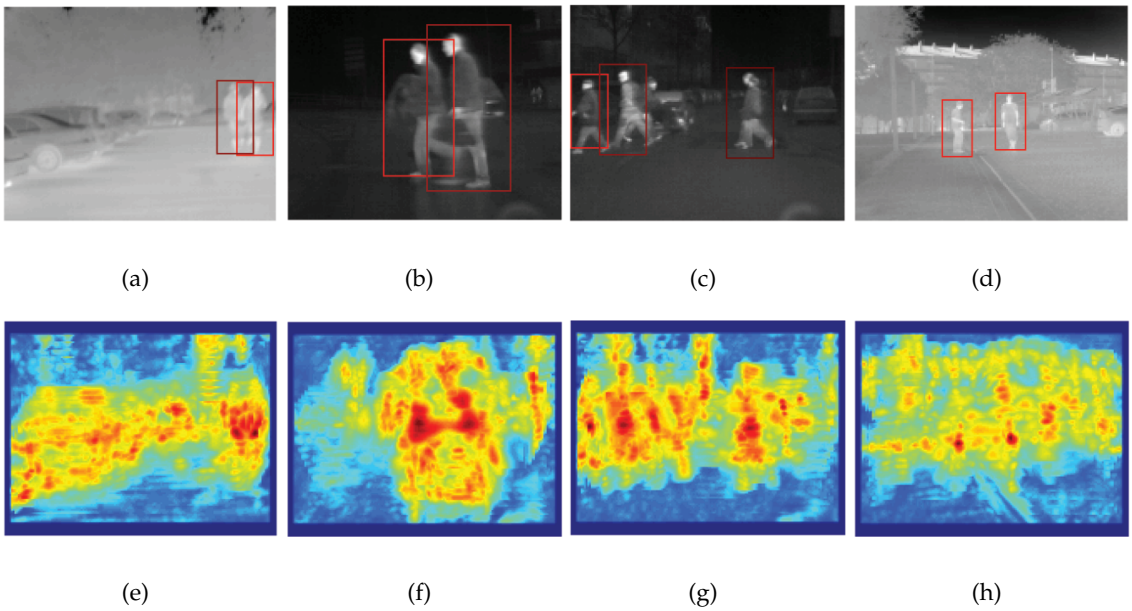


Figure 3.9: LSI Results show multiscale detection of pedestrians across wide range of scales. The estimated likelihood of pedestrian's location measured across all the scales is shown under each frame. As before, the dark red to reddish black denotes high to very high confidence of detector.

multi-scale detection along with the score map are shown in Fig. 3.7. The scores are shown as heat color map where blue denotes very low confidence whereas dark red to red-black denotes high to very high confidence in predicting a pedestrian.

Like in OSU-T, the background remains stationary here. But repeated occlusions, pedestrian's proximity with other pedestrians and physical structures on the ground, and noisy nature of the images keep giving false positives requiring a hard mining step. We applied hard mining in a restricted setting similar to OSU-T, with uniformly but sparsely sampled training images from the full training set.

LSI Far Infrared Pedestrian Database: This dataset comes with two flavors: classification setup as well as detection setup. We have focused on the detection set that is

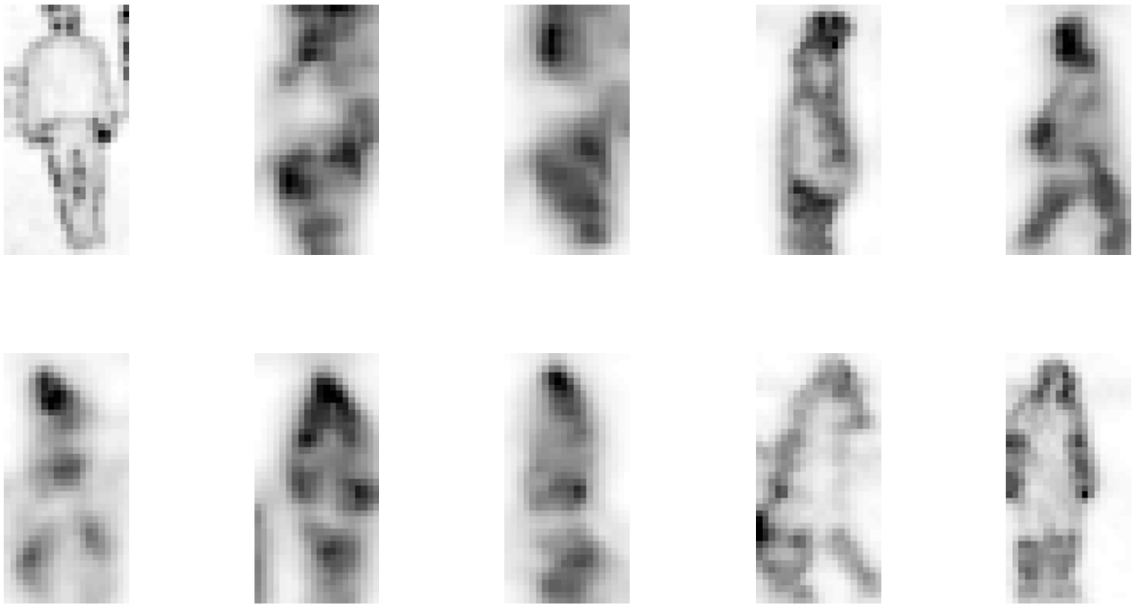


Figure 3.10: The shape of pedestrian is prominent positive support vectors shown in the form of first LSK feature channel. More importantly, the positive support vectors show how the linear kernel has succeeded to learn a set of widely different poses of pedestrians.

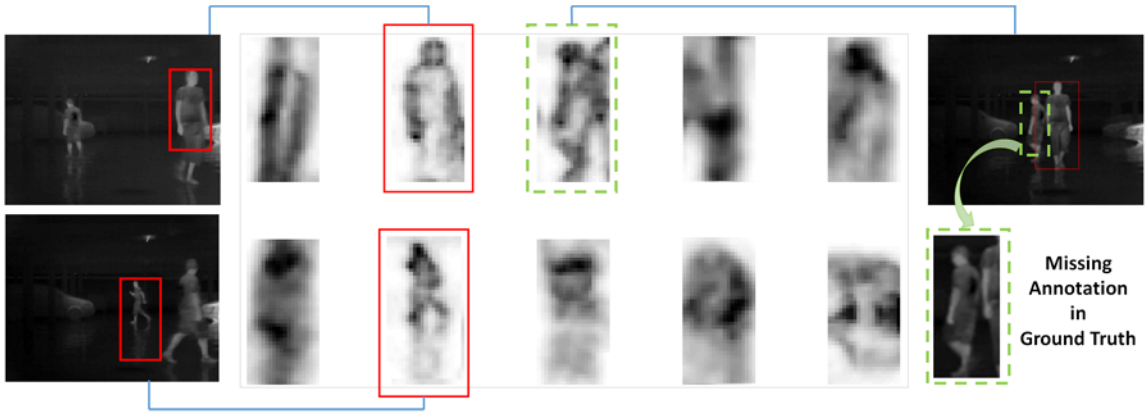


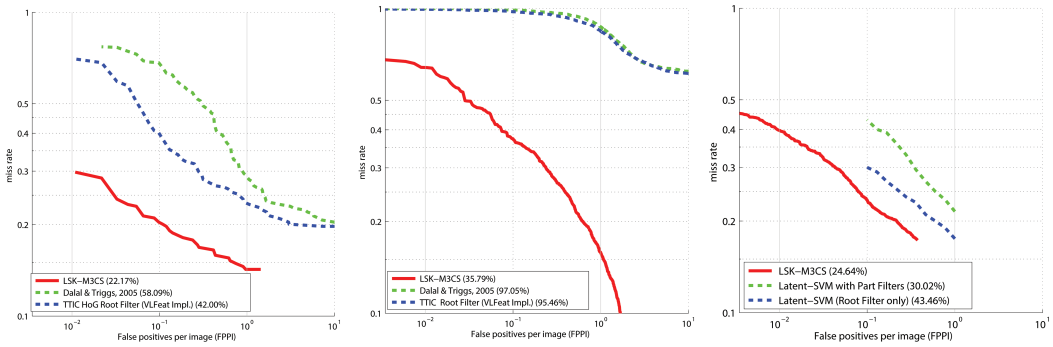
Figure 3.11: Negative support vectors are shown to have come from hard mining step where undersized or oversized detection have resulted into false negatives (on left). On right, we show an instance where the correct detection is made but absence of such annotation in ground truth has forced this example into being a false negative.

further divided in two subsets, training and test set. The training set has 3225 positive images and 1601 negative images. The test set includes 3279 positive and 4859 negative images. The images are 164 pixels wide and 129 pixels tall. Since the intensities of LSI images roughly range from 31000 to 35000 (they are 16 bit images), we have scaled the intensities to 0-255 without noticeable loss in performance. Following the practice followed in out experiments we have collected negative examples initially from randomly sampled negative windows in positive images. Later, the hard negatives have come from wrong prediction in the training images. The hard mining stage is executed once [34]. As a consequence, we have categorically ignored the negative images in the LSI training set, however, we evaluate our model on full test including both positive and negative images.

Fig. 3.8 show the LSK feature channels corresponding to a thermal image. The LSK features characteristically decomposes the gray scale image in contour, horizontal and vertical segments. The learning scheme following next builds up on this feature representation. In Fig. 3.10 we show the positive support vectors by displaying the first channel of LSK tensor feature. One can notice the wide range of poses captured in the learning process of the support vectors. Fig. 3.11 illustrates negative support vectors which have resulted from hard mining step after being either under or over detected bounding box. The same figure also illustrates an example where missing annotation in ground truth pushes it into hard negative set. This shows that the proposed methodology is robust to noise and outliers present in the ground truth.

3.7.1 Results & Discussions

There are usually two approaches available when it comes to computing features, namely, feature engineering and feature learning. Boosting, sparse coding and recently



(a) (b) (c)

Figure 3.12: Miss rate versus false positives per image (FPPI) for the three datasets: (a) OSU-T, (b) OSU-CT and (c) LSI thermal dataset.

convolutional neural network are learning methodologies one can apply for learning the features. On the other hand, histogram features, especially HOG has dominated the object detection scenario in the first decade of this century leading to the success of deformable part model.

In our work, however, two of the datasets, namely OSU-T and OSU-CT have pedestrians so small that explicit modeling of parts is not a feasible idea to apply. We have implemented a HOG based linear SVM using the library VLFeat as a baseline for comparison purpose. HOG implemented in VLFeat comes in two forms, one being the originally proposed in [34] and the other is dimension-reduced form used in [15]. The proposed detector works superior to HOG based linear SVM both on OSU-T and OSU-CT achieving lowest miss rate (Fig. 3.12(a) and (b) respectively). However, the extremely occluded nature of pedestrians in OSU-CT has made the detection task challenging for both HOG and LSK with MCS kernel. Fig. 3.12(c) shows the performance on LSI dataset in comparison with HOG root filter and Latent-SVM with parts [15]. We refer the reader to [99] where the

authors have pointed out how introduction of parts in Latent-SVM introduces a derogatory performance on LSI as it often leads to confusion among the part detector in absence of robust texture in the low resolution and noisy thermal images. The proposed feature with our chosen MCS kernel has been able to achieve minimum miss rate as shown in Fig. 3.12.

3.8 Conclusion

In this chapter we have extended and investigated the use of LSK tensors for pedestrian detection task in thermal infrared images. In earlier work LSK was used for one shot detection, with matrix cosine similarity as decision rule. Here we have shown matrix cosine similarity as a kernel function and using this kernel we have proposed a max-margin framework for learning a rigid detector for pedestrian detection. Since we are dealing with tensor features, a discrete fourier transform based computation of decision rule in each channel allows faster search of pedestrians. The proposed methodology is compared with other state of the art detectors known to perform well with visible range sensors on the publicly available data sets of thermal infrared images. We have shown how robust estimation of local manifold which lies at the heart of LSK computation as well as matrix cosine similarity as decision rule deal with the characteristic challenges of noisy and low resolution thermal images delivering state of the art performance.

Chapter 4

On the Role of Dimensionality Reduction in Object Detection

Abstract – In this chapter we present a study of the role of dimensionality reduction in object detection. It is shown that a good subspace not only leads to better discrimination between object and background but also makes the detection process efficient. It is particularly important to note that the Local Steering Kernel (LSK) coefficients reside in a low dimensional manifold. This is understandable given the brief discussion on dimensionality reduction with PCA in earlier chapters. The substantial energy of the eigenspectrum has been shown to be stored in the few leading eigenvalues. As a consequence, the right subspace, quite predictably, would return us the few channels of salient features.

While PCA does a decent job of quickly computing a set of salient feature channels by dimensionality reduction, we here propose another technique called Laplacian Eigenmap that stores much more detailed image representation in the projected feature space. We apply such subspace learning technique in the one-shot object detection task

reporting improved detection performance.

4.1 Introduction

Dimensionality reduction in the context of this thesis can be broadly categorized in the two branches: linear dimensionality reduction and multilinear dimensionality reduction. Linear dimensionality reduction works with the vectorial representation of features. This is the classical approach to dimensionality reduction technique when the features, even if they have the natural form of matrix or higher order tensors, are converted to long vectors. Following this technique the dimensionality reduction amounts to embedding such long vectors into a lower dimensional subspace. Note the subspace learning algorithm is agnostic to the natural shape of the feature data, which originally could be matrix or higher order tensors. Multilinear algebra addresses this limitation of linear dimensionality reduction technique by providing a mathematical framework in which the features can retain their *multidimensional shape* (e.g., higher order matrices or tensors), and still we will be able to reduce the dimensions of the tensors. In other words, multilinear subspace learning treats each of the several dimensions of a tensor separately while holding other dimensions constant, and gives us, a way to decorrelate each dimension from others.

4.2 Linear Dimensionality Reduction

Linear dimensionality reduction techniques can be broadly categorized in two branches: unsupervised and supervised. The unsupervised dimensionality reduction techniques does not consider class labels. The subspace learning takes place with the opti-

mization of various objective functions that aim at preserving one or other property of the dataset (e.g., maximizing total variation of the dataset as done by PCA). Some of the examples in this category include the following:

- Principal component analysis
- Kernel principal component analysis
- Multidimensional scaling
- Locality linear embedding
- Locality preserving projection

It is a well known fact that that all of the aforementioned techniques ultimately lead to an eigenvalue problem. However, a relatively less known observation is that the objective function in each of the dimensionality reduction techniques mentioned above can be derived from a unifying perspective of a least square based error minimization principle. An excellent description of this perspective is available in [104] authored by Fernando De la Torre.

On the other hand supervised dimensionality reduction techniques take class labels into account. Their supervised nature tend to make them bit more expensive to compute than their unsupervised counterpart but such cost, in general, also yields better detection performance. Some of the supervised subspace learning techniques known to have performed well in the object detection domain include Linear Discriminant Analysis (LDA) and Partial Least Square Analysis.

In the following we present a study related to the unsupervised dimensionality reduction technique [79, 80] with an application to solve one shot object detection task.

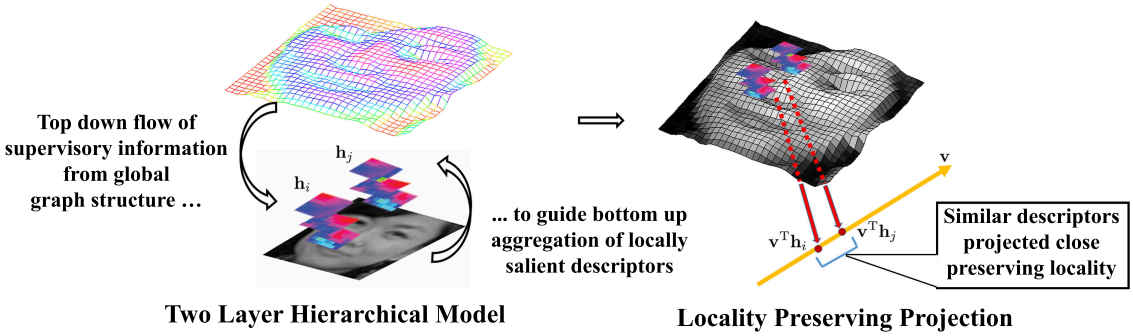


Figure 4.1: *Laplacian Object*: computing a query subspace that preserves intrinsic image geometry — on left, the proposed two-layer hierarchical model is shown where top layer of global context (in the form of an affinity graph) guides the bottom up aggregation of local information from low level descriptors. On right, locality preserving projection [1] with the graph Laplacian is used as a mathematical framework to represent the two-layer hierarchy.

Since, there is no scope for training in one shot detection the supervised algorithms are not feasible. Also, non-linear algorithms capture better data representation than linear ones do. We seek a linear approximation to a nonlinear subspace learning (Laplacian Eigenmap) that serves our objective.

4.2.1 *Laplacian Object*: A Framework for Locally Salient Feature Computation

We begin the model description by assuming an $m \times n$ gray-scale query image Q and $M \times N$ sized target image T. We visualize a gray-scale image as the parameterized image surface $\mathcal{S}(\mathbf{x}_i) = \{\mathbf{x}_i, z(\mathbf{x}_i)\}$, where $\mathbf{x}_i$ denotes the 2D coordinate vector $\mathbf{x}_i = [x_{i_1}, x_{i_2}]^{\mathrm{T}}$, having intensity $z(\mathbf{x}_i)$. We compute local image descriptors (e.g., SIFT [9], LSK [86]) densely at every pixel, that makes the number of descriptors from Q as mn, and from T as MN. The descriptor at location $\mathbf{x}_i$ is denoted as a l-dimensional vector $\mathbf{h}_i \in \mathbb{R}^l$. The descriptor vectors $\mathbf{h}_{Qi}$ for query, and $\mathbf{h}_{Ti}$ for target, are stacked column wise to define the de-

scriptor matrix for query as $\mathbf{H}_Q = [\mathbf{h}_{Q1}, \mathbf{h}_{Q2}, \ldots, \mathbf{h}_{Qmn}] \in \mathbb{R}^{l \times (mn)}$, and the same for target as $\mathbf{H_T} = [\mathbf{h}_{T1}, \mathbf{h}_{T2}, \ldots, \mathbf{h}_{TMN}] \in \mathbb{R}^{l \times (MN)}$.

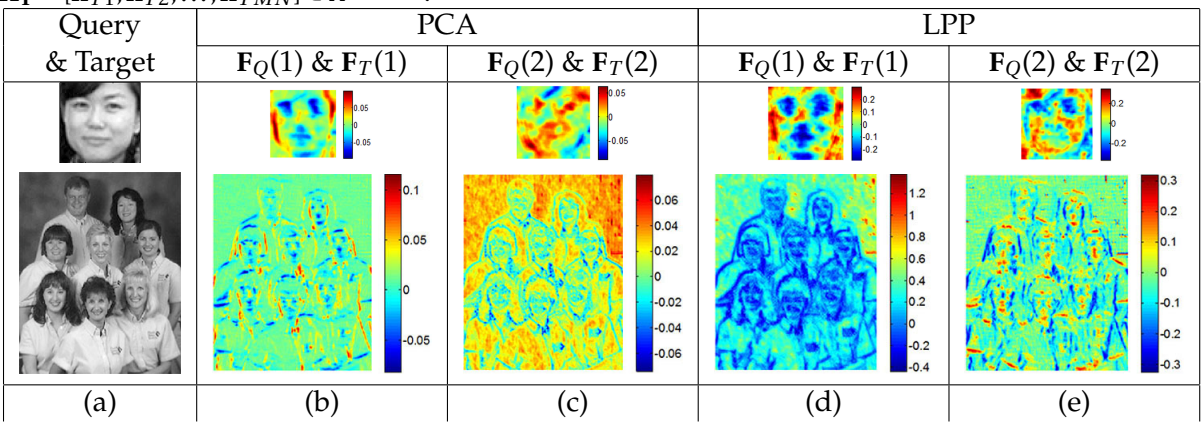


Figure 4.2: Salient features shown after dimensionality reduction of LSK descriptors: (a) query & target images, (b)-(c) salient query (target) features $\mathbf{F}_Q$ ($\mathbf{F}_T$) learnt by projecting descriptors $\mathbf{H}_Q$ ($\mathbf{H}_T$) along two dominant principal components, (d)-(e) same LSK descriptors projected along two dominant eigenvectors of LPP (one can notice finer local details in these features)

To distill the redundancy resulting from dense computation of descriptors we embed $\mathbf{H}_Q$ in a global graph structure, represented by an affinity matrix $\mathbf{K} \in \mathbb{R}^{mn \times mn}$, that takes into account the spatial relationship among the descriptors. Our goal is to estimate a low dimensional but discriminatory subspace $\mathbf{v}$ from Q such that the query descriptors $\mathbf{h}_{Qi}$, when projected on $\mathbf{v}$, respect the local geometric pattern. In other words, if $\mathbf{h}_{Qi}$ and $\mathbf{h}_{Qj}$ are closely spaced over the image manifold $\mathscr{S}$ then their projections $\mathbf{v}^{\mathrm{T}}\mathbf{h}_{Qi}$ and $\mathbf{v}^{\mathrm{T}}\mathbf{h}_{Qj}$ on a subspace $\mathbf{v}$ should be close as well (Fig. 4.1). The theory of locality preserving projection (LPP) [105] ensures this criterion by minimizing the following objective function for $|bf v$ with fixed $\mathbf{K}_{ij}$ —

$$J_{\mathrm{LPP}} = \frac{1}{2}\sum_{ij}(\mathbf{v}^{\mathrm{T}}\mathbf{h}_{Qi} - \mathbf{v}^{\mathrm{T}}\mathbf{h}_{Qj})^2\mathbf{K}_{ij}. \tag{4.1}$$

The objective function J_{LPP} with our proposed affinity measure $\mathbf{K}_{ij}$ incurs heavy penalty if

neighboring descriptors $\mathbf{h}_{Qi}$ and $\mathbf{h}_{Qj}$ are mapped *far apart*. Each element of graph affinity matrix $\mathbf{K}$, denoted as $\mathbf{K}_{ij}$, indicates the pairwise affinity between descriptors $\mathbf{h}_{Qi}$ and $\mathbf{h}_{Qj}$ of the query Q. Upon simplification, (4.1) leads to the following penalty (for details refer to [105]):

$$J_{\text{LPP}} = \mathbf{v}^{\text{T}}\mathbf{H}_Q\mathbf{L}\mathbf{H}_Q^{\text{T}}\mathbf{v}. \tag{4.2}$$

Defining the diagonal matrix $\mathbf{D}_{ii} = \sum_j \mathbf{K}_{ij}$, the matrix $\mathbf{L} = \mathbf{D} - \mathbf{K}$ is known as the graph Laplacian. Recent research shows success of graph Laplacian in effective exploration of local patterns in massive graphical networks [106, 107]. Here we have studied the related Laplacian eigenmap [108, 109] to embed the informative but redundant descriptors into a low dimensional but salient feature space (Fig. 4.1).

To prevent abnormally high values of $\mathbf{D}_{ii}$ (which means unusually greater "importance" to descriptor $\mathbf{h}_i$) in (4.2), the constraint $\mathbf{v}^{\text{T}}\mathbf{H}_\mathbf{Q}\mathbf{D}\mathbf{H}_\mathbf{Q}^{\text{T}}\mathbf{v} = 1$ is imposed on $\mathbf{D}$. Minimizing J_{LPP} with respect to the aforementioned constraint we obtain the following optimization problem:

$$\min_{\mathbf{v}} \mathbf{v}^{\text{T}}\mathbf{H}_Q\mathbf{L}\mathbf{H}_Q^{\text{T}}\mathbf{v} \text{ subject to } \mathbf{v}^{\text{T}}\mathbf{H}_Q\mathbf{D}\mathbf{H}_Q^{\text{T}}\mathbf{v} = 1. \tag{4.3}$$

The projection vector $\mathbf{v}$ that minimizes the above is given by the minimum eigenvalue solution to the following generalized eigenvalue problem:

$$\mathbf{H}_Q\mathbf{L}\mathbf{H}_Q^{\text{T}}\mathbf{v} = \lambda\mathbf{H}_Q\mathbf{D}\mathbf{H}_Q^{\text{T}}\mathbf{v}. \tag{4.4}$$

The desired set of eigenvectors which builds our low dimensional LPP subspace comprises the trailing d eigenvectors computed as a solution of (4.4). We collect the set of d eigenvectors as columns of $\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_d] \in \mathbb{R}^{l \times d}$. Since the descriptors are densely computed they typically lie on a lower dimensional manifold. As a consequence, we can expect d to be quite small in comparison to the dimension l of the descriptors. In practice, d is selected

to be a small integer, and it turns out that this small set of eigenvectors is good enough to discriminate the query from the background clutter. The descriptor matrices $\mathbf{H}_Q$ and $\mathbf{H}_T$, when projected on $\mathbf{V}$, lead to salient features that preserve locality as guaranteed by the objective function (4.1). The locally salient features $\mathbf{F}_Q$ and $\mathbf{F}_T$, for query Q and target T respectively, are defined by the following equations:

$$\mathbf{F_Q} = \mathbf{V}^{\mathrm{T}}\mathbf{H}_Q \in \mathbb{R}^{d\times(mn)}; \mathbf{F_T} = \mathbf{V}^{\mathrm{T}}\mathbf{H}_T \in \mathbb{R}^{d\times(MN)}. \tag{4.5}$$

The salient query (target) features $\mathbf{F}_Q$ ($\mathbf{F}_T$) learnt with PCA and LPP are shown in Fig. 4.2. The results in the figure(s) demonstrate that LPP is able to preserve greater amount of details in the projected features than PCA. During detection the detailed contour and inherent spatial geometry captured in LPP result in better localization. The reason why LPP features inculcate more information compared to PCA features [6] lies in the construction of respective objective functions as explained below.

Descriptors $\mathbf{h}_{Qi}$ and $\mathbf{h}_{Qj}$ typically encode geometric information in various channels, but if they **share** similar local geometry they would likely have a high pairwise similarity term $\mathbf{K}_{ij}$. Consequently, a high $\mathbf{K}_{ij}$ would penalize the cost function in case $\mathbf{h}_{Qi}$ and $\mathbf{h}_{Qj}$ are projected far apart. The pairwise similarity term ensures that the local continuity of the fine edge structure would be preserved on the projected subspace.

PCA does not care which descriptor comes from where – it retains the global geometric structure of the data (as evident from the mean term $\overline{h}$ below) without providing any room for preserving local details. It maximizing the following objective function,

$$J_{\mathrm{PCA}} = \sum_i (\mathbf{v}^{\mathrm{T}}\mathbf{h}_{Qi} - \mathbf{v}^{\mathrm{T}}\overline{\mathbf{h}})^2 \tag{4.6}$$

The graph encoded by $\mathbf{K}_{ij}$ in (4.1) provides further insights into the working principles of LPP and PCA. We denote the total number of pixels mn in query by n_q. Suppose

we connect $\mathbf{x}_i$ with all other pixels $\mathbf{x}_j$ of the query image obtaining a complete graph with constant weights $\mathbf{K}_{ij} = \frac{1}{n_q^2}, \forall \mathbf{x}_i, \mathbf{x}_j$. Then $\mathbf{L} = \mathbf{D} - \mathbf{K} = \frac{1}{n_q^2}\mathbf{I} - \frac{1}{n_q^2}\mathbf{e}\mathbf{e}^{\mathrm{T}}$, where $\mathbf{e}$ is a vector of all ones. Under this graph construction, and denoting mean $\overline{h} = \frac{1}{n_q}\sum_i \mathbf{h}_{Qi}$ we get,

$$\mathbf{H}_Q \mathbf{L} \mathbf{H}_Q^T = \frac{1}{n_q}\mathbf{H}_Q(\mathbf{I} - \frac{1}{n_q}\mathbf{e}\mathbf{e}^{\mathrm{T}})\mathbf{H}_Q^{\mathrm{T}}, \tag{4.7}$$

$$= \frac{1}{n_q}\mathbf{H}_Q\mathbf{H}_Q^{\mathrm{T}} - \frac{1}{n_q^2}(\mathbf{H}_Q\mathbf{e})(\mathbf{H}_Q\mathbf{e})^{\mathrm{T}}, \tag{4.8}$$

$$= \frac{1}{n_q}\sum_i \mathbf{h}_{Qi}\mathbf{h}_{Qi}^{\mathrm{T}} - \frac{1}{n_q^2}(n\bar{\mathbf{h}})(n\bar{\mathbf{h}})^{\mathrm{T}}, \tag{4.9}$$

$$= \frac{1}{n_q}\sum_i (\mathbf{h}_{Qi} - \overline{\mathbf{h}})(\mathbf{h}_{Qi} - \overline{\mathbf{h}})^{\mathrm{T}}. \tag{4.10}$$

This happens to be the covariance matrix of the data set that is used in PCA (J_{PCA} upon simplification boils down to $J_{\mathrm{PCA}} = \mathbf{v}^{\mathrm{T}}\left[\sum_i (\mathbf{h}_{Qi} - \overline{\mathbf{h}})(\mathbf{h}_{Qi} - \overline{\mathbf{h}})^{\mathrm{T}}\right]\mathbf{v}$). The analysis above (and also in [105]) suggests when we care about global structure of LSK descriptor space we connect each descriptor location (i.e., each pixel) to all others in the graph construction, and project the descriptors along the direction of maximal variance. When we seek to preserve local information in reduced dimension we connect each pixel to its immediate neighborhood, and project the descriptors along the direction that minimizes local variation.

A fact of theoretical interest is the subtle difference between Laplacian eigenmap [108] and LPP [105]. The former provides us a non-linear manifold, and a (linear) projection operation like (4.5) does not hold true in case of Laplacian eigenmap. LPP, as described in [105], can be seen as a linear approximation to the non-linear Laplacian eigenmap allowing us to project the query and target features as shown in (4.5).

Local Desciptors & Affinity Matrix From a Unifying Geodesic Perspective

The proposed detection framework is general enough to use with any local descriptor (e.g., SIFT [9], HOG [34]). However, we advocate LSK descriptor because it is

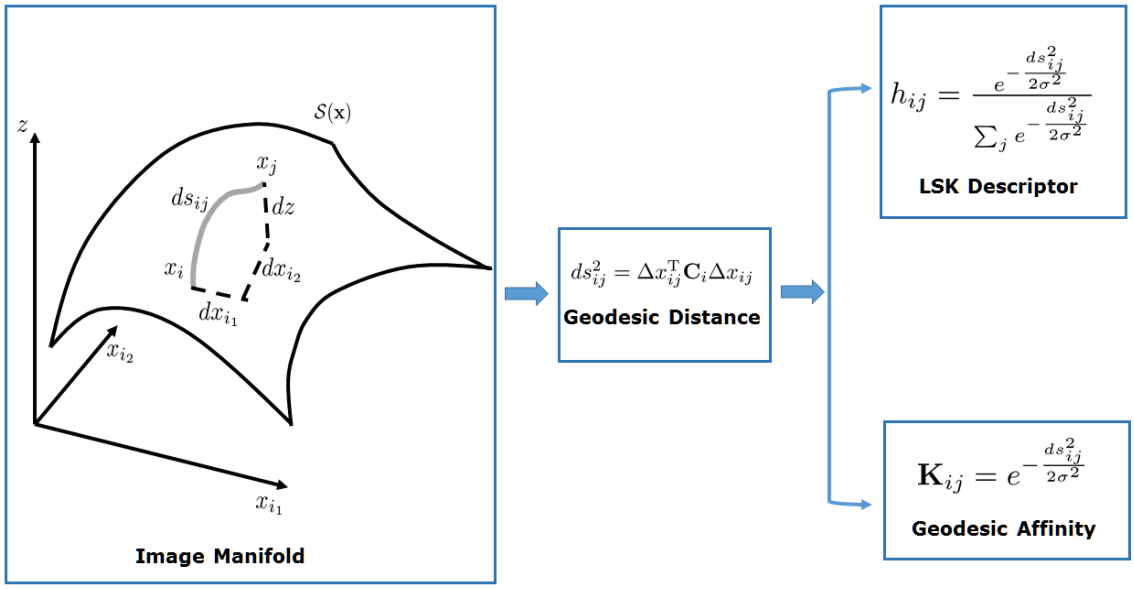


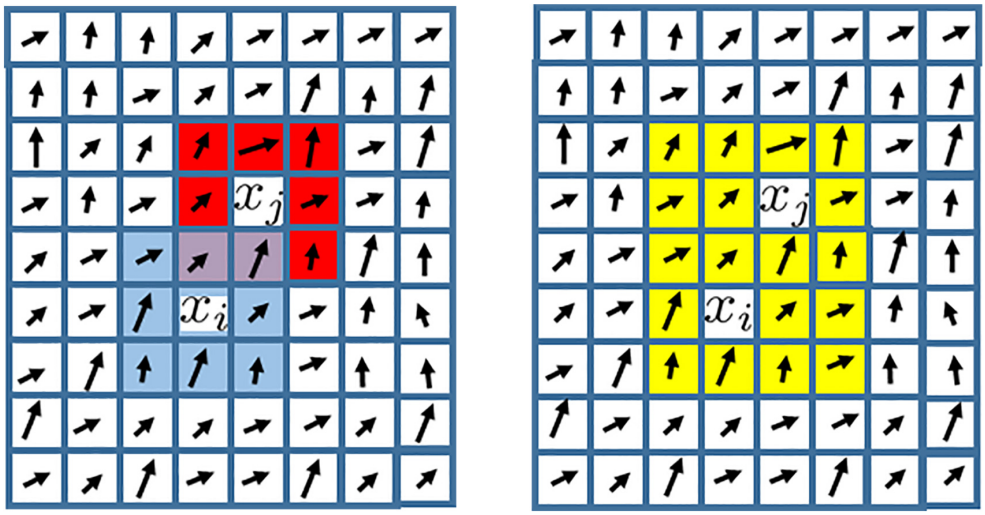
Figure 4.3: Unifying geodesic framework: the geodesic distance (ds_{ij}) between the points $\mathbf{x}_i$ and $\mathbf{x}_j$ on the image manifold $\mathscr{S}(\mathbf{x})$ is used to derive both the LSK descriptors and affinities on the right

specifically designed for one shot object detection [6, 71]. In fact, the geodesic interpretation behind LSK descriptors introduced in [71] motivates us to present an unifying geometric perspective to connect the definitions of LSK descriptor $\mathbf{h}_i$ and the graph affinity $\mathbf{K}_{ij}$.

The local geodesic distance (Fig. 4.3) between the two neighboring descriptor locations $\mathbf{x}_i$ and $\mathbf{x}_j$ on the image manifold $\mathscr{S}(\mathbf{x}_i)$ is approximated [71] by the differential arc length ds_{ij} as follows (see Section 2.2 for details):

$$ds_{ij}^2 = dx_{i_1}^2 + dx_{i_2}^2 + dz^2 \approx \Delta x_{ij}^{\mathrm{T}} \mathbf{C}_i \Delta x_{ij}. \tag{4.11}$$

The approximation involves the following discretizations: $dx_{i_1} \approx \Delta x_{i_1 j_1} = x_{j_1} - x_{i_1}$, and $dx_{i_2} \approx \Delta x_{i_2 j_2} = x_{j_2} - x_{i_2}$ (i.e., $\Delta x_{i_1 j_1}$ and $\Delta x_{i_2 j_2}$ representing displacements along the two



(a) (b)

Figure 4.4: Estimation of covariance matrix $\mathbf{C}$ from local gradients (shown with black arrows): (a) For LSK descriptors we estimate $\mathbf{C}_{\Omega_i}$ from (4.13) using the support patch Ω_i corresponding to $\mathbf{x}_i$ as shown in (blue) color. Note, Ω_j (in red) corresponding to $\mathbf{x}_j$ is different from Ω_i. To make $\mathbf{K}_{ij}$ symmetric (b) shows the rule adopted for defining a common support for $\mathbf{x}_i$ and $\mathbf{x}_j$ using the patch Ω_{ij} (shown in yellow) over which $\mathbf{C}_{\Omega_{ij}}$ is estimated (4.15).

image-axes in Fig. 4.3). Also, we assume $\Delta x_{ij} = [\Delta x_{i_1 j_1} \quad \Delta x_{i_2 j_2}]^{\mathrm{T}}$, and the matrix $\mathbf{C}_i$ denotes the local gradient covariance matrix (also called as steering matrix in [72]) computed at $\mathbf{x}_i$.

Computation of LSK descriptors

The descriptor $\mathbf{h}_i$ denotes a multidimensional vector $\mathbf{h}_i = (h_{i1}, h_{i2}, \ldots, h_{ij}, \ldots, h_{p^2})$ computed at pixel $\mathbf{x}_i$ over a $p \times p$ local window. In Section 2.2, we defined the general term h_{ij} as a measure of similarity between two descriptor locations $\mathbf{x}_i$ and $\mathbf{x}_j$,

$$h_{ij} = \frac{e^{-\frac{ds_{ij}^2}{2\sigma^2}}}{\sum_{j=1}^{p^2} e^{-\frac{ds_{ij}^2}{2\sigma^2}}}, \quad j = 1, 2, \ldots, p^2, \tag{4.12}$$

where ds_{ij} is approximated as in (4.11). The normalization in the denominator is carried out by summing the local geodesic similarities over all the neighbors of $\mathbf{x}_i$ in its $p \times p$ local neighborhood. LSK descriptors when normalized to a unit vector become robust to illumination changes.

We also described in Section 2.2 how noisy single pixel estimate of $\mathbf{C}_i$ in (4.11) can be replaced by more robust $\mathbf{C}_{\Omega_i}$ computed from a patch. With reference to that, we estimate $\mathbf{C}_{\Omega_i}$ in a reliable fashion by first computing the derivatives of the image signal $z(\mathbf{x}_i)$ over a patch Ω_i of pixels centered at pixel $\mathbf{x}_i$ (Fig. 4.4(a)). As shown in equation (2.2) in Section 2.2, a subsequent eigen decomposition step leads us to the following which we rewrite here for the ease of continuation to subsequent concepts,

$$\begin{aligned} \mathbf{C}_{\Omega_i} \quad &= (\sqrt{\nu_1 \nu_2} + \varepsilon)^{\theta} \cdot \\ &\quad \left(\frac{\sqrt{\nu_1} + \tau}{\sqrt{\nu_2} + \tau} \mathbf{u_1} \mathbf{u_1}^{\mathrm{T}} + \frac{\sqrt{\nu_1} + \tau}{\sqrt{\nu_2} + \tau} \mathbf{u_2} \mathbf{u_2}^{\mathrm{T}}\right), \end{aligned} \tag{4.13}$$

where, ν_1 and ν_2 are eigenvalues of $\mathbf{C}_{\Omega_i}$ corresponding to eigenvectors $\mathbf{u}_1$ and $\mathbf{u}_2$, respectively. Also in the derivation above, $\varepsilon, \tau, \theta$ are regularization parameters to avoid numerical

instabilities and kept constant throughout all the experiments in this paper at 10^{-7}, 1 and 0.1 respectively.

Building the Graph Laplacian with Geodesic Affinities

Next, we build a graph structure from Q with descriptors representing the graph nodes. The edges in the graph denote affinities between neighboring descriptor locations $\mathbf{x}_i$ and $\mathbf{x}_j$ as follows,

$$\mathbf{K}_{ij} = \begin{cases} e^{-\frac{ds_{ij}^2}{2\sigma^2}} & \text{when } \|\mathbf{x}_i - \mathbf{x}_j\|_2 \leq \gamma, \\ 0 & \text{otherwise ,} \end{cases} \tag{4.14}$$

where σ is a smoothing parameter (kept same in (4.12) for LSK descriptor), and γ a radius within which we limit the affinity computation. The choice of γ is not too critical as long as it covers decent neighborhood size (typically 3 to 5 pixel radius). Setting γ too high increases the computational burden, and may involve derogatory confluence of too many and irrelevant neighborhood information. In fact, as also observed in [105], too much aggregation of information collected over a bigger neighborhood may invariably affect LPP's embedding performance.

Unfortunately, $\mathbf{K}_{ij}$ defined above is non symmetric. However, the derivation of LPP subspace in (4.2) assumes a symmetric affinity matrix $\mathbf{K}$. To understand why $\mathbf{K}_{ij}$ is non symmetric we note computing $ds_{ij} = \Delta x_{ij}^{\mathrm{T}} \mathbf{C}_{\Omega_i} \Delta x_{ij}$ following the definition of $\mathbf{C}_{\Omega_i}$ in (4.11) makes $ds_{ij} \neq ds_{ji}$. This is because the support Ω_i of $\mathbf{C}_{\Omega_i}$ is centered at $\mathbf{x}_i$ (Fig. 4.4(a)), and similarly, support Ω_j of $\mathbf{C}_{\Omega_j}$ is centered at $\mathbf{x}_j$, hence, $\mathbf{C}_{\Omega_i} \neq \mathbf{C}_{\Omega_j}$. Therefore, to ensure $\mathbf{K}_{ij}$ to be symmetric we make the supports of $\mathbf{C}_{\Omega_i}$ and $\mathbf{C}_{\Omega_j}$ common as shown by a circumscribing rectangle in Fig. 4.4(b). The common support is denoted by Ω_{ij}, and we write the corresponding gradient covariance matrix as $\mathbf{C}_{\Omega_{ij}}$. It follows directly that

$ds_{ij} = ds_{ji} = \Delta x_{ij}^{\mathrm{T}} \mathbf{C}_{\Omega_{ij}} \Delta x_{ij}$, and the final expression of affinity becomes the following:

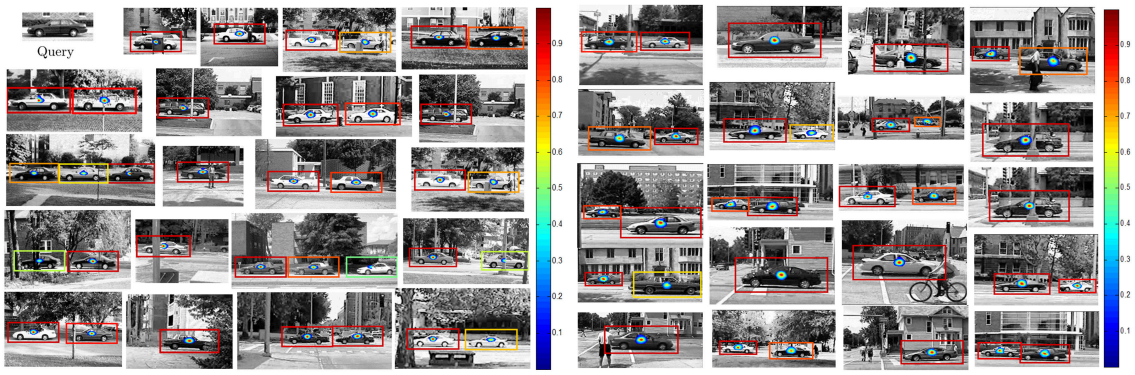
$$\mathbf{K}_{ij} = \begin{cases} e^{-\frac{\Delta x_{ij}^{\mathrm{T}} \mathbf{C}_{\Omega_{ij}} \Delta x_{ij}}{2\sigma^2}} & \text{when } \|\mathbf{x}_i - \mathbf{x}_j\|_2 \leq \gamma, \\ 0 & \text{otherwise .} \end{cases} \tag{4.15}$$

Note, to ensure symmetry a straightforward averaging ($\mathbf{K}_{ij} :== \frac{\mathbf{K}_{ij}+\mathbf{K}_{ji}}{2}$, and $\mathbf{K}_{ji} :== \frac{\mathbf{K}_{ij}+\mathbf{K}_{ji}}{2}$) does not work well in practice because such average oversmooths the dominant structure pattern over the image manifold.

To summarize, since each LSK channel captures a specific orientation pattern, different edge orientations manifest themselves in different LSK channels along with relative signal strength of the edges (in terms of ν_1, ν_2 in (4.13)). Flat regions receive low values in all the channels, but edge structures show up as high values in appropriate channel depending on the orientation. Getting all the directional information of 81 channels (when $p = 9$ form (4.12)) in 5 or 6 low dimension is difficult: when PCA does this job it tends to show the fine structures like eyes, nose, or mouth of faces (or contour of parts in case of generic objects) as blobs in high contrast regions, and completely misses the relatively faint parts in low contrast region (Fig. 3). In contrast, LPP is able to retain the delicate image geometry relatively better. The reason is by virtue of pairwise similarity LPP keeps the projected descriptors close, if similar, thereby maintaining local continuity of fine details. Viewed from an alternative perspective, LPP projects the high dimensional LSK channels along a direction that minimizes the weighted local variance following a least square framework (4.1), preserving geometric details in lower dimension.

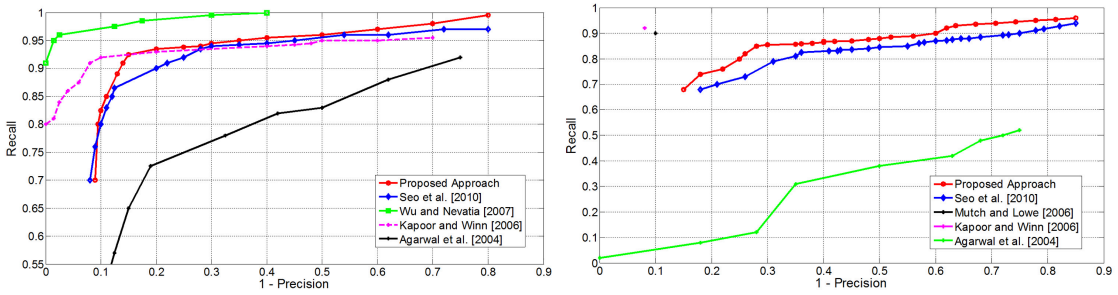
Experimental Setup, Results & Discussions

In this section, we evaluate the proposed low dimensional features along with runtime performance of the proposed detection methodology. All the experiments are



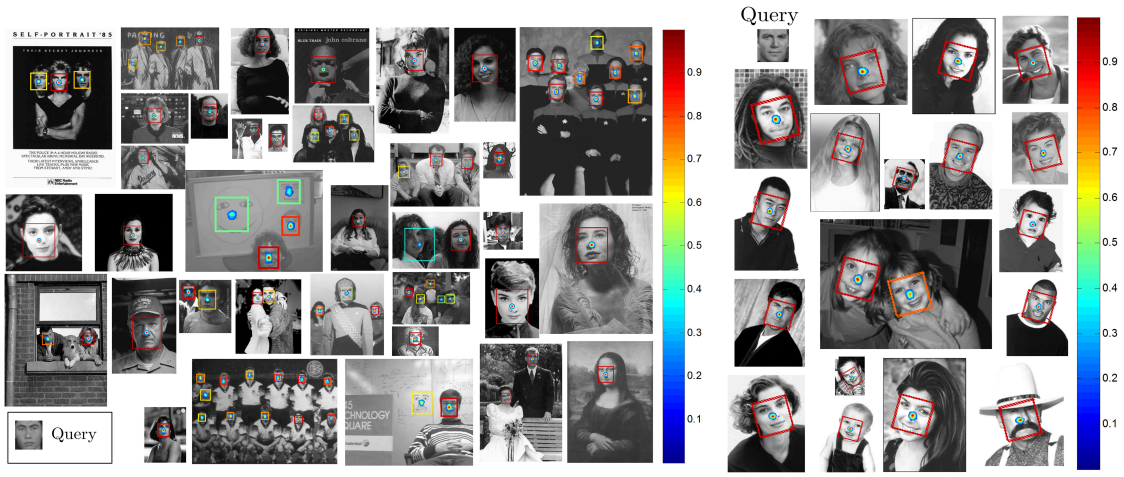
(a) (b)

Figure 4.5: Example detections on UIUC car test set [2] are shown here. (a) Single scale car detection (the query image is shown top left), and (b) Multiscale car detection (the same query image as used in single scale experiment is used here). The FDR α is set at 1%. The $f(\rho)$ values above the threshold τ corresponding to α is embedded inside the displayed bounding box. A red bounding box indicates highest resemblance to query image, and for other colors the colormap shown right depicts relative resemblance.



(a) (b)

Figure 4.6: Precision recall curves obtained from the evaluation of our proposed methodology on UIUC single scale car test set (left), and UIUC multiscale car test set (right) in comparison to other training based state of the arts [3, 4, 2, 5] as well as training-free state of the art methodology [6].



(a) (b)

Figure 4.7: Face detection in MIT-CMU face data set [7] is illustrated in the figure above. (a) Example detections along with scale estimation are shown using a query face (bottom left). (b) Sample detections along with pose estimation are shown when the scales as well as orientations of the query both vary in target images. In both the experiments, the FDR α is set at 1% to determine the threshold τ. The thresholded $f(\rho)$ is shown inside the bounding box. The correct bounding box results from the maximum likelihood estimate of probable set of scales and orientation. The colormap on right is a mapping between color of bounding box and the measure of resemblance in case of multiple detection; the red means highest resemblance.

done in a standard desktop machine with 8 GB RAM, Intel Core i7-2600 CPU @3.40 GHz using standard MATLAB functions with no GPU support. Of course, our proposed methodology is general enough to avail of the benefit of GPU computation which would result in even shorter computation time.

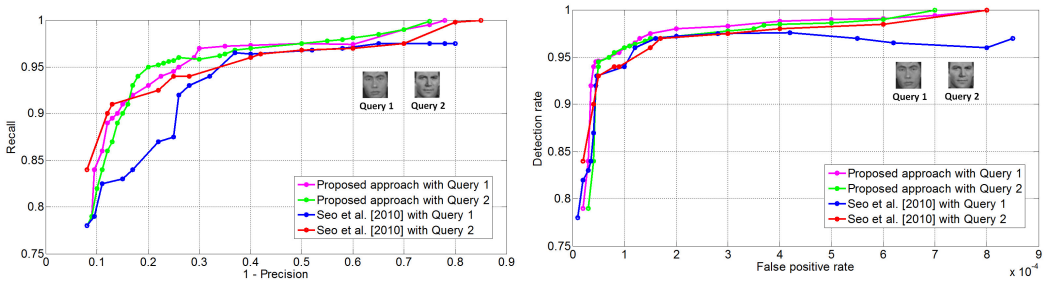
In the first part, we have evaluated our methodology on three benchmark data sets: UIUC car data set [2], MIT-CMU face data set [7], Shechtman's general object data set [8], and Caltech 101 data set [110, 111, 112]. The input to the algorithm is a query image with a single dominant object present, e.g., face or car, plus a target image. The typical out-

put of our proposed methodology is a set of bounding boxes drawn around the detected object of interest. By bounding box we mean the smallest possible rectangle drawn around the detected object in the target image. We evaluate the object detection algorithm following the criterion described in [2]: if the detected region overlaps considerably with the ground truth we accept the output of the algorithm as true positive (or correct detection). Otherwise, the detection is regarded as false alarm. With each pair of recall and precision value collected by varying the false discovery rate α, we draw precision-recall and/or receiver operating characteristics (ROC) curves. Also, for the purpose of comparison with other competing detection methodologies we report *detection equal error rate* which is same as recall rate when recall is equal to precision.

UIUC Car Data Set This gray-scale image data set comprises training (500 car and 500 noncar images) and test sets. The test set contains car images at i) same scale (with 170 images of 200 cars, some images having multiple cars of size approximately 100 × 40 pixels matching closely with the size of the cars in the training sets), and at ii) multiple scales (with 108 images of 139 cars at various sizes where the ratio of scales between largest and smallest car being around 2.5). Since this paper focuses on one shot object detection task, we use a query car image (randomly selected) from the training set as our query object.

The LSK descriptors at pixel $\mathbf{x}_i$ are computed over a 9×9 patch centered at $\mathbf{x}_i$ yielding 81-dimensional local descriptors $\mathbf{h}_i$. The smoothing parameter σ for computing LSK is set to 1.0; the value of σ in the estimation of pairwise affinity $\mathbf{K}_{ij}$ (4.15) between $\mathbf{h}_i$ and $\mathbf{h}_j$ also remains the same. Following locality preserving projection we reduce the dimension of LSK descriptors from $l = 81$ to $d = 5$ by choosing the 5 trailing eigenvectors of (4.4). It is observed that selecting more eigenvectors does not produce noticeable change

in the detection performance. Performing a significance test by setting the FDR $\alpha = 1\%$ we obtain the threshold τ for each test example. Fig. 4.5(a) shows an example of single scale car detection. In case of multiscale detections, as already mentioned in Section 4.2, we do not enforce any feature transform on target features $\mathbf{F}_T$. The query features $\mathbf{F}_Q$ are scaled as much as 2.5 times for robust detection of objects. We use 0.5 times to 2.5 times scaling of query features by a step size of 0.2. The detection performance of multiscale analysis is shown in Fig. 4.5(b). The performance of our algorithm is reported after aggregating the results of multiple query images. For a particular threshold τ we obtain a set (for each randomly selected query image from the training set) of precision-recall values for the whole query set which we average to obtain a single precision-recall pair, and next, by varying τ we draw the precision-recall curve in Fig. 4.6(a)-4.6(b). The overall performance shows improvement as a consequence of preserving locality in derived features. The proposed approach presented here has also been compared with training based approaches in Table 4.1. The results show that our training-free methodology has been able to take the performance of one-shot detection close to some of the training-based ones.



(a) (b)

Figure 4.8: Evaluation of proposed detection technique on MIT-CMU face data set in comparison to [6]: (a) precision-recall curve, (b) ROC curve

MIT-CMU Face Data Set This is also a gray-scale image data set and we have evaluated our methodology on the same subset of images as done by Seo *et al.* [6]. The motivation behind using MIT-CMU data set is to subject our algorithm to severe scale changes (up to as much as a scale factor of 5), and large in-plane rotation of the query pattern as well. The test set consists of 43 gray-scale images (list given in [6]) containing a total of 149 frontal faces, occurring at various scales, and 20 gray-scale images having faces at unusually large ($> 60°$) angular orientation. The query faces used for detection as shown in Fig. 4.7(a) and 4.7(b) each has a size 61×61. As has been done in [6], we do not resize the target image to bring it at the same scale as that of the query. Instead, we engage in a multiscale and multiorientation search for the query to achieve correct detection optimizing over its pose parameters. Specifically, for a particular scale we search over all angular orientations, from $0°$ to $360°$, with an interval of $30°$. Parameters like smoothing parameters (h), LSK descriptor size (9×9), number of eigenvectors d for dimensionality reduction, and FDR α remain the same as the ones used in UIUC car data set.

Figures 4.7(a) and 4.7(b) show the efficacy of our proposed method. We are able to detect faces at various scales with a tight bounding box. The rotated faces are also detected with correct localization and adequate orientation — the displayed results show the correct angle estimated for the oriented face. Since our aim is to detect visually similar instances we have also been able to detect faces drawn on a white board. To show that our algorithm is not specific to a given query face, we have used multiple query faces (following the experimental framework of [6]) to detect similar instances in target images. Following the evaluation scheme we have presented our result in the form of precision-recall and ROC curves in Fig. 4.8(a)-4.8(b) with two query faces.

General Object Data Set In this section, we present the performance of our algo-

Table 4.1: Detection Equal Error Rates on UIUC cars and MIT-CMU faces (multiscale and multi-orientation)

Datasets	Proposed Approach	Training Based Approaches				
		Agarwal *et al.* [2]	Mutch & Lowe [5]	Kapoor & Winn [4]	Lampert *et al.* [88]	Wu & Nevatia [3]
UIUC Single Scale	90.76	77.08	99.94	94.00	98.5	97.6
UIUC Multiscale	79.01	44.00	90.60	93.50	98.60	-
MIT-CMU Faces	91.24	-	-	-	-	-

rithm on color images. In Shechtman and Irani's general object data set [8] we have applied the proposed concepts to match pose *symbols* of humans with relevant human poses in general photographs (Fig. 4.9). Several challenging query and target pairs are taken from categories like flowers, heart symbols, peace symbols, and faces (Fig. 4.10). We follow the similar parameter settings like previous experiments except being little cautious with FDR ($\alpha = 0.5\%$) to deal with false positives in a more conservative fashion. To study color information one can consider several color space models like RGB, YCbCr, and CIE L*a*b*. Our experiments support the findings already reported in [6] and [8], that CIE L*a*b* color model is the most discriminatory. In fact, the luminance channel alone is sufficient to distinguish the object from the clutter in most of the cases as shown in precision-recall and ROC curves in Fig. 4.11(a)-4.11(b). In [6], Seo et al., have proposed the use of Canonical Cosine Similarity (CCS) to combine all three color channels for improved detection performance. We endorse their view but at the same time we note that the resulting performance gain as seen from precision-recall and ROC curves is not terribly significant. This comes as no surprise because the structural information (excellently captured by LSK descriptors) alone is enough to compare the visual geometry of query and target, and it is readily available in the luminance channel.

We have compared the performance of the proposed features with other state of

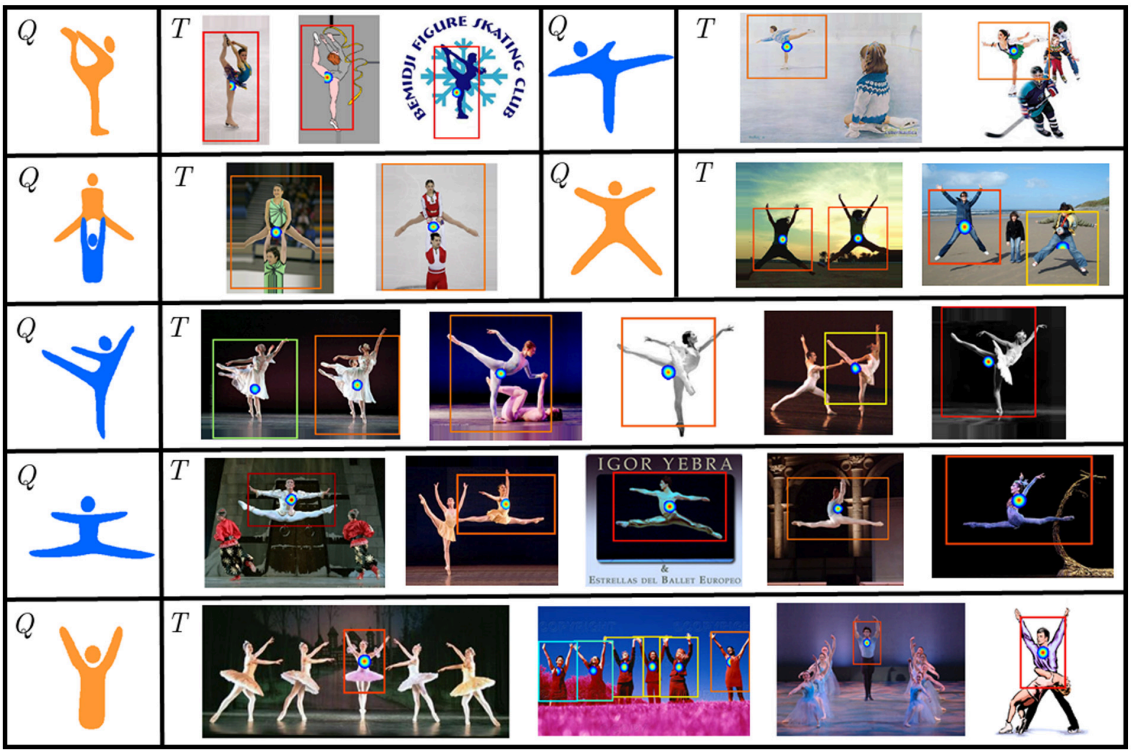


Figure 4.9: The human pose symbols as query objects are detected in real life photographs in Shechtman and Irani's general object data set [8]. The query symbols are displayed in the Q panel, and corresponding target detections are shown on right in the T panel. We set the FDR α at 0.5% to deal with false positives conservatively.

the art descriptors like *GLOH* [10], *Shape Context* [11], *SIFT* [9] using the implementation in [10]. We have computed all the local descriptors as densely as possible. To facilitate a fair comparison among the descriptors we have maintained the proposed way of matching in the rest of the detection process. In other words, we have carried out the experiment on the data set by replacing the LPP features with these descriptors but keeping the rest of the steps the same. The proposed graph-based dimensionality reduction technique is able to robustly capture the local image structure as clearly visible in the performance curves of

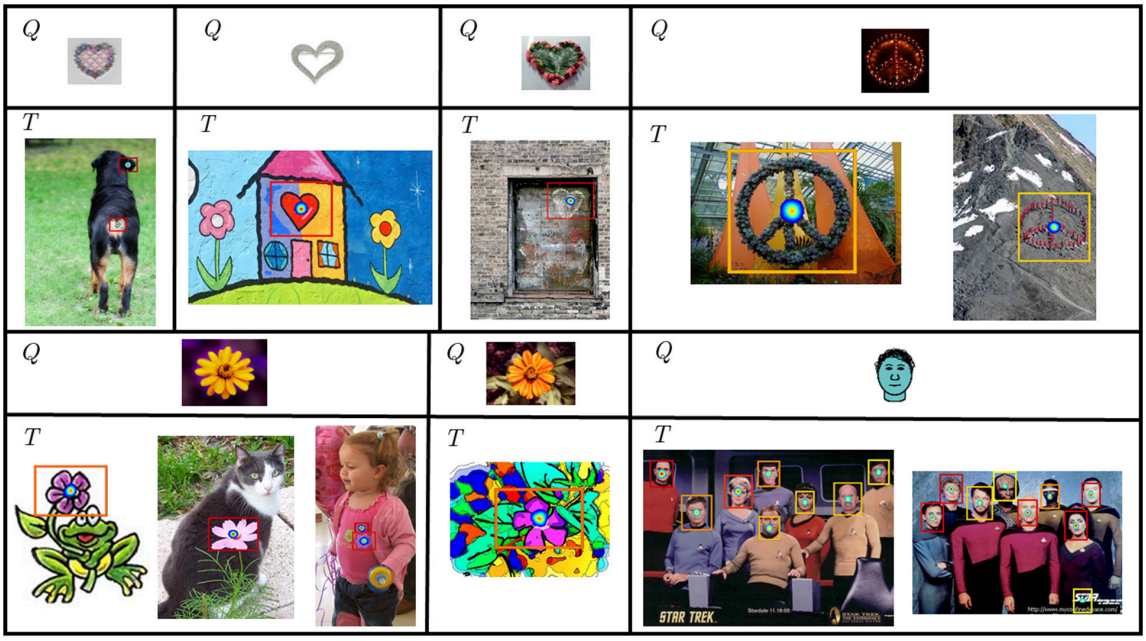
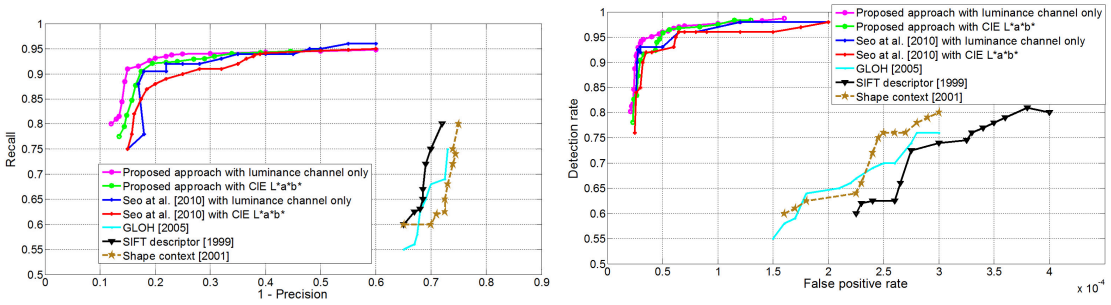


Figure 4.10: More examples from Shechtman and Irani's general object data set [8]: Query objects (heart symbol, peace symbol, flower and sketch of human face) are displayed in Q panel; setting the FDR $\alpha = 0.5\%$ we show the corresponding detections in T panels just underneath the relevant Q panels.

Fig. 4.11(a)-4.11(b) with different parameter settings of τ.

The proposed detector achieves a detection equal error rate of 84.4% on this data set. In contrast, self-similarity descriptor when densely computed and used in our matching framework yields a detection rate of 79.2%. Saliency based pruning technique to remove redundant and noisy features followed by nearest neighbor voting based matching [113] improves the detection to 82.7% but that gain comes with considerable computational cost as also observed by Chatfield *et al.* [113]. Besides pruning of features, another factor that further increases the runtime is computation of self-similarity at multiple scales over a Gaussian image pyramid. Note, the performance stated above does not contradict the



(a) (b)

Figure 4.11: Evaluation of proposed detection technique on Shechtman-Irani general object data set [8]: on left, precision-recall curves are shown, and on right the ROC curves show the performance of the proposed algorithm along with [6], SIFT [9], GLOH [10], and Shape Context [11]. Experiment is conducted using only luminance channel as well as all CIE L*a*b* channels. In case of CIE L*a*b* channels, canonical cosine similarity [6] is used to fuse information from three channels.

reported 86% detection rates of self-similarity by Schechtman *et al.* [8] because we did not implement the star-graph based ensemble matching that they used in conjunction with self-similar descriptors. In fact, it is not directly evident how ensemble matching performs in terms of false positive rate as well as computational efficiency when compared with MCS based detection, because [8] did not mention the false positive rate (corresponding to reported detection rate) and computation time. Therefore, our evaluation makes the proposed methodology more practical as Figure 4.11 provides explicitly the estimates of false alarm versus detection tradeoff. The runtime analysis is discussed in Section V-F.

Performance Analysis of Embedding Techniques

Considering our proposed contribution it becomes imperative to study the performance benefit of the locality preserving embedding in contrast to raw LSK channels as well as other embedding technique. For that purpose we have introduced Table 4.2 that

Table 4.2: Detection Rates of Raw LSK and Projected Features

Data Sets	LSK All Channels [6]	LSK + PCA [6]	Proposed Approach
UIUC Car (Single Scale)	83.92	87.13	90.76
UIUC Car (Multiscale)	73.33	75.47	79.01
MIT-CMU Faces	84.76	86.58	91.24
General Object Data [8]	81.58	83.35	84.41

shows results produced by different components related to the present detector. The proposed LPP features derived from LSK, shown in fourth column of Table 4.2, works superior to PCA projections of LSK [6] as mentioned in third column.

It is also noted in the second column of Table 4.2 that using raw LSK feature channels (with the implementation of [6]) affects the detection performance as also observed by [6]. This is because too much channel information has a derogatory influence on the detector. In contrast, the locality preserving projection aggregates channel information from all LSK channels in such a way that the object contours become very prominent in lower subspace, and the non-contour parts tend to get smoothed out (Fig. 3(d)-(e)). This makes sense because $\mathbf{K}_{ij}$ in locality preserving cost function (1) is built with local aggregation of gradient vectors (Fig. 5(b)), and the subsequent dimensionality reduction causes strong contours in dominant projections.

Also worth mentioning is the computational benefit that comes with maintaining a low number of feature channels as a result of the embedding, and in the subsequent sections we will see how a few discriminatory feature channels (typically four or five) aids rapid processing of an image in real time.

4.3 Multilinear Dimensionality Reduction

The tool for understanding tensors is multilinear or tensor algebra. Multilinear algebra allows us to explore not only feature channels (third dimension of a third order tensor), but all of the three dimensions individually (i.e., height, width, and depth). The techniques involved come under the broad umbrella of "Supervised Tensor Learning" [114].

Suppose that we have N feature examples altogether combining the positive as well as negative exmaples. In the context of Chapter 3, that means, each such feature vector corresponds to a sliding window location that represents either a pedestrian or a background. For vector features the maximum margin framework is given by the following loss function in the primal form [1]:

$$\underset{\mathbf{W},b}{\text{minimize}}\ \frac{1}{2}\|\mathbf{W}\|^2 + C\sum_{i=1}^{N}\max\{0, 1 - y_i(\langle \mathbf{W}, \mathbf{F}_i\rangle + b)\}. \tag{4.16}$$

Hence, following the discussion in Chapter 3, the dual form of the aforementioned maximum margin framework reduces to the following:

$$\begin{aligned}\underset{\alpha}{\text{maximize}}\ &\sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i,j}\alpha_i\alpha_j y_i y_j \langle \mathbf{F}_i, \mathbf{F}_j\rangle_F, \\ &\text{subject to } 0 \le \alpha_i \le C \text{and } \textstyle\sum \alpha_i y_i = 0, \text{and } \alpha \in \mathbb{R}^N.\end{aligned} \tag{4.17}$$

An important observation here is that if the examples $\mathbf{F}$ come from an $m \times n \times d$ sliding window, we estimate m.n.d + 1 parameters in the primal, and N parameters in the dual. But neither the primal and nor the dual in the formulations above have machineries to explore the individual mode (or dimension) when the features $\mathbf{F}$ actually have tensor form.

[1] We have assumed the features $\mathbf{F}$ are $L2$ normalized

Actually, optimization of (4.16) or (4.17) leads us to the following form of the final classifier.

$$\underbrace{f(\mathbf{F};\alpha)}_{dual} = \sum_{j=1}^{q} \alpha_j y_j \langle \mathbf{F}_j, \mathbf{F} \rangle + b, \tag{4.18}$$

$$= \langle \underbrace{\sum_{j=1}^{q} \alpha_j y_j \mathbf{F}_j}_{\widetilde{\mathbf{W}}}, \mathbf{F} \rangle + b, \tag{4.19}$$

$$= \langle \widetilde{\mathbf{W}}, \mathbf{F} \rangle + b, \tag{4.20}$$

$$= \underbrace{f(\mathbf{F};\widetilde{\mathbf{W}})}_{primal}. \tag{4.21}$$

Multilinear algebra provides us a window to treat the features in their original form. That means, the tensor to vector transformation of features (by rearranging the elements of a tensor in a long vector) is no more required, and we can henceforth represent our features (with a slight abuse of notation) $\mathbf{F}_i \in \mathbb{R}^{m \times n \times d} \forall\, i$. In essence, multilinear algebra provides an inner glimpse of each mode (i.e., dimension) of the tensor — the roles that they supposedly play (e.g., causal factors like illumination and pose [115]). The idea here is to apply the linear model $\mathbf{a}'\mathbf{w} + b$ but separately in each dimension [114]. This becomes possible by constraining the template tensor $\mathbf{W} \in \mathbb{R}^{m \times n \times d}$ to be a sum of R rank-1 tensors — a direct result of CANDECOMP/PARAFAC (CP) decomposition (an analogy of SVD) applicable to higher order tensors. This is given by the following,

$$\mathbf{W} \approx \sum_{r=1}^{R} \mathbf{w}_r^{(1)} \circ \mathbf{w}_r^{(2)} \circ \mathbf{w}_r^{(3)} \text{ (CANDECOMP/PARAFAC)}, \tag{4.22}$$

$$\approx \mathbf{w}^{(1)} \circ \mathbf{w}^{(2)} \circ \mathbf{w}^{(3)} \text{ (Rank} - 1 \text{ approximation)}. \tag{4.23}$$

where '$\circ$' represents the tensor outer product [116]. It is worth pointing out that $\mathbf{w}^{(1)} \in \mathbb{R}^m, \mathbf{w}^{(2)} \in \mathbb{R}^n$, and $\mathbf{w}^{(3)} \in \mathbb{R}^d$. Using the notion of Frobenius inner product $\langle \cdot \rangle_F$ (that represents a generalization of inner product from vector to tensors), the classifier in the context

of the tensor features takes the following shape:

$$f(\mathbf{F};\mathbf{W}) = \langle \mathbf{F}, \mathbf{W} \rangle_F + b, \tag{4.24}$$

$$= \langle \mathbf{F}, \mathbf{w}^{(1)} \circ \mathbf{w}^{(2)} \circ \mathbf{w}^{(3)} \rangle_F + b, \tag{4.25}$$

$$= \mathbf{F} \prod_{i=1}^{3} \times_i \mathbf{w}^{(i)} + b, \tag{4.26}$$

$$= f(\mathbf{F};\mathbf{w}^{(1)},\mathbf{w}^{(2)},\mathbf{w}^{(3)}). \tag{4.27}$$

Combining (4.24)-(4.27) with (4.23) leads us to the following optimization framework:

$$\underset{\mathbf{w}^{(1)},\mathbf{w}^{(2)},\mathbf{w}^{(3)}}{\text{minimize}} \frac{1}{2}\|\mathbf{W}\|^2 + C\sum_{j=1}^{N} \max\{0, 1 - y_j(\mathbf{F}_j \prod_{i=1}^{3} \times_i \mathbf{w}^{(i)} + b)\}, \tag{4.28}$$

$$\mathbf{w}^{(1)} \in \mathbb{R}^m, \mathbf{w}^{(2)} \in \mathbb{R}^n, \mathbf{w}^{(3)} \in \mathbb{R}^d, b \in \mathbb{R}.$$

Note, we need to learn $\mathbf{w}^{(1)}, \mathbf{w}^{(2)}$ and $\mathbf{w}^{(3)}$. Since, the objective function (4.28) is no more convex, there is no way we can not approach the global optimum of the objective. A usual practice involves employing alternate projection [114] (or coordinate descent minimization) algorithm which optimizes (4.28) with respect to $\mathbf{w}^{(1)}$ holding $\mathbf{w}^{(2)}$ and $\mathbf{w}^{(3)}$ constant. The same process of optimization one set of parameters holding others constant is applied to $\mathbf{w}^{(2)}$ and $\mathbf{w}^{(3)}$, in succession, leading to an iterative minimization of (4.28).

It is important to note that we estimate $m \cdot n \cdot d$ parameters in primal formulations of vector features, and N parameters in the dual representation. However, the multilinear algebra while treating features as tensors permits us to optimize $m + n + d$ parameters. This is a huge difference particularly when we need to keep the model complexity low to avoid overfitting. In fact, the comparison of model complexity between the linear and multilinear algebra further indicates which learning procedure to employ and under what condition. In practice, when the data dimensionality is high and the number of examples N is low it is worth considering minimization (4.28). The low complexity will no doubt

guard the solution from potential overfitting.

In our dataset the number of examples N is comparable to, or even greater than, the complexity of $\mathbf{W}$. This comes as no surprise because the pedestrians on an average look small in our dataset. At the same time, a reasonably high number of them, along with an equally good number of challenging background examples, motivate us to use a rigid template tensor with decent enough complexity. It appears that number of parameters available in multilinear support tensor machine would be to few to handle the variations present in the dataset. Of course, a trade-off can be made by experimenting with an increasing R in the rank-1 approximation of $\mathbf{W}$. However, it is not clear at this point whether such endeavor is justified in exchange of the much simpler but effective linear model. So, (4.28) in our case may ruin the detector much of its discriminatory power, especially when the number of examples goes high. In principle, we have followed (4.16)-(4.17) for learning the detector parameters but preserved the shape of the detector during prediction so as to leverage the exact acceleration of matrix cosine similarity with multichannel discrete Fourier transform (Chapter 2).

4.4 Conclusion

In this chapter we have studied visual similarity between two images which could lead to robust and efficient object detection. Given a single query object, searching the same in a bigger image is a hard task given various pose and scale variations that the object undergoes. We have addressed such concerns in this work. Typically, the descriptors traditionally used in visual recognition for encoding image geometry have important information in all the channels. Extracting a useful gist of them without sacrificing the descriptor's discriminative power is not straight forward. To address such concern, we have

studied a graph based dimensionality reduction method by combining local signal patterns with global context, preserving discriminative details of image patterns for one shot object detection and concurrent pose estimation. The algorithm described is quite general; one can integrate the methodology with any descriptors depending on the application in hand. The results with LSK descriptors show LPP improves detection in comparison to PCA by being aware of local structure, thereby making correct estimate of the object location, its scale, and orientation.

Chapter 5

Conclusions and Future Scope

5.1 Future Work

Though the basic elements like Local Steering Kernel tensors and maximum margin matrix cosine similarity have been shown to perform well on big thermal datasets of pedestrians, there is of course nothing preventing their extension to other areas of visual recognition. One immediate and natural extension could be tracking problems. It is particularly difficult to track pedestrians in poorly illuminated conditions. Things are therefore challenging with thermal infrared datasets where distinguishing foreground from background always offer various challenges. Occlusion in crowded scenarios compound that difficulty.

We also explore feature learning mechanism in the context of convolutional neural network. Recently, learning features by means of deep architecture has shown commendable progress in locating pedestrians. We explore the prospect of our methodology in this context in the second part.

5.2 Tracking

Tracking algorithms can broadly be classified in two categories: generative and discriminative. Generative methods model the appearance of the target, often in the form of a template as done by Jepson et al. [117]. Recently, $l1$-norm optimized based sparse coding methodology has also been tried, following a generative model, with success to develop tracker robust to occlusion and noise [118].

A different class of technique approaches the tracking problem by posing it as a classification task. This kind of model, also known as discriminative trackers, classify the target in a relatively bigger window of interest that includes a part of the background too [119, 120]. Where generative models emphasize on the modeling aspects on the target only without paying heed to the background, the discriminative tracker on the other hand considers the background too in course of the classification task.

Tracking approaches include a bottom up stage where contours (e.g., active contours in condensation algorithm [121]) or templates [122] are learnt, or updated , in order to deal with the changing appearance of the object in subsequent frames. Simultaneously, a top down approach is also needed, which is a part of the motion model, to associate data in the current frame with that in the previous frame(s).

5.2.1 Probable Approach to Tracking in Thermal Videos

The suggested approach should consist of two phases. In the first phase we are interested to employ a trained appearance model. The appearance model does the job of automatically detecting the pedestrians. Note there could be multiple of them and some of them might be heavily occluded. In the second phase, the appearance model needs to be continuously updated. In other words, we would seek an adaptive appearance model to

deal with pose changes. The frequency of updates depends on the application in hand and how fast the pedestrians change appearance (scales, left or right orientation for examples). We would like to investigate the use of LSK tensor as appearance model in developing a pedestrian tracker.

While the appearance model updated this way would help us deal with change in pedestrian poses, a motion model would restrict the zone of search and solve the data association problem. The data association would help us solve temporary occlusions, both partial and full. Extensive research has been done on the design choices for building a good motion model (e.g., [121]), and the Bayesian framework of particle filtering still remains popular for the successful development of an object tracker. In this context, we intend to study the current research results so far [123, 124]that aim at minimizing *visual drift* of the target object.

5.3 Feature Learning for Pedestrian Detection

Recently, with the rise of deep learning the various aspects of object detection (or specifically, pedestrian detection [125]) like part modeling, occlusion handling, deformation modeling which earlier required careful designs can be solved directly in an end to end learning scenario. For example, Ouyang et al. [126] have explored such a deep architecture with an application to pedestrian detection in natural images. Their reported system jointly learns parts, deformation and occlusion in a unified framework. Some recent results [127] have however suggested that for pedestrian detection a carefully designed rigid detector can go a long way than relatively complex deformable part models [15]. This is a promising direction that we would like to explore in our investigation. Specifically, we are interested in learning templates for a rigid detector which would be able to handle

occlusion and pose alterations but would be free from the complexities of part designs.

The work by [126] is further extended by Girshick et al. [128] to use geometric relationship as available in the framework of deformable part model to be used along with the deep learning architecture. They have suggested that deformable part models can readily be cast in the design of convolutional neural network. This is interesting as our proposed root filter is inspired by the design of deformable part model. A connection like [128] also inspires us to explore similar possibility in the present context.

5.4 Summary and Conclusion

In this thesis, we have advocated a fundamentally different representation of images in the form of tensors toward efficient and effective object detection task. We have showed the excellent performance of LSK coefficients to achieve that representation. The use of Matrix Cosine Similarity to build scalable, general machine learning system is another important aspect of this thesis. The experimental results on challenging data have demonstrated that the proposed framework outperforms state of the art methods. Following is the chapter wise summary of this thesis.

Chapter 1 This chapter discussed the ideas that prevailed in the history of object detection. The lessons learnt are then used to point out the limitation of current feature representation and lack of robust detection strategy to handle noisy, thermal images, or decide detection in restricted environment like one shot cases. We summarized the contributions in the concluding part of the thesis.

Chapter 2 The fundamental concepts and ideas were introduced in this chapter. Notably, feature tensor representation with LSK and computing decision rule with matrix cosine similarity (MCS) are discussed. Following the expensive brute force computation

of MCS, we showed how the inefficiencies could be improved for better detection performance. We proposed an exact acceleration of MCS computation and no approximation happened is allowed anywhere in the computation pipeline.

Chapter 3 Here we proposed a methodology to use Local Steering Kernel (LSK) tensors as low-level descriptors for detecting pedestrians in far infrared images. LSK is specifically designed to deal with intrinsic image noise and pixel level uncertainty (proving more robust and descriptive than HOG). Our second contribution was the introduction of a new image similarity kernel in the popular max-margin framework of support vector machines that results in a relatively short and simple training phase for building a rigid template as pedestrian detector. Our third and last contribution was to replace the sluggish but *de facto* sliding window based detection methodology with multichannel discrete Fourier transform, facilitating very fast and efficient pedestrian localization.

Chapter 4 In this chapter we presented a study of the role of dimensionality reduction in object detection. While PCA did a decent job of quickly computing a set of salient feature channels by dimensionality reduction, we proposed a graph based dimensionality reduction technique called Laplacian Eigenmap that stored much detailed image representation in projected feature space. We applied such subspace learning technique in the one shot object detection task reporting improved detection performance compared to PCA results.

Chapter 5 In the past chapters we described the individual components of a complete object detection methodology. In this chapter we considered potential extension of such approach to other uses. One immediate and useful extension of the our detection architecture should be tracking pedestrians in thermal infrared videos. It would be of particular interest to research community to see how the LSK tensors and associated matrix

cosine similarity as similarity function responded to the challenges of tracking pedestrians in low resolution thermal videos, often in heavily occluded scenarios. In view of recent success with feature learning that pushed the state of the art in many visual recognition challenges, we also explored the possibility of feature learning in our proposed framework.

We conclude this thesis by an introspection of the winning strands that were prominent in the history of object detection and discussed in details in the introduction of the thesis. Many of the winning ideas like careful feature representation, mid-level representations like templates, correlation as an effective and efficient form of decision rule for detection (that existed in the form of linear SVM) were used in successful building of our proposed detector. We have followed the lessons learnt. However, in course of our design we improvised on those lessons as well as keeping many of them unaltered there they performed reasonably well. Therefore, the proposed thesis will be seen advocating tensor representation as an improved from of feature representation, taking deviation from histogram statistics and resorting to more powerful modeling of local structure, maintaining mid-level template designs, and suggesting a multichannel design of efficient correlation computation. An important question that remains to be answered is how the ideas explored in this thesis merge with the rapidly changing landscape of visual recognition. We wish to seek the answer to such question in the scope of suggested future work.

Bibliography

[1] X. He and P. Niyogi, "Locality preserving projections," in *Advances in Neural Information Processing Systems*, 2004, pp. 153–160.

[2] S. Agarwal, A. Awan, and D. Roth, "Learning to detect objects in images via a sparse, part-based representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 11, pp. 1475–1490, 2004.

[3] B. Wu and R. Nevatia, "Simultaneous object detection and segmentation by boosting local shape feature based classifier," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.

[4] A. Kapoor and J. Winn, "Located hidden random fields: Learning discriminative parts for object detection," in *European Conference on Computer Vision*. Springer, 2006, pp. 302–315.

[5] J. Mutch and D. J. Lowe, "Multiclass object recognition with sparse, localized features," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2006, pp. 11–18.

[6] H. J. Seo and P. Milanfar, "Training-free, generic object detection using locally adaptive regression kernels," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1688–1704, 2010.

[7] H. A. Rowleya, S. Baluja, and T. Kanade, "Neural network based face detection,"

IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 20, no. 1, pp. 23–38, 1998.

[8] E. Shechtman and M. Irani, "Matching local self-similarities across images and videos," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.

[9] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[10] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615–1630, 2005.

[11] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 4, pp. 509–522, 2002.

[12] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1. IEEE, 2001, pp. I–511.

[13] P. Viola and M. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, 2004.

[14] P. Felzenszwalb, D. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2008, pp. 1–8.

[15] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010.

[16] W. E. L. Grimson, D. P. Huttenlocher, *et al.*, *Object recognition by computer: the role of geometric constraints*. The MIT Press, 1990.

[17] A. Yali and D. Geman, "A computational model for visual selection," *Neural computation*, vol. 11, no. 7, pp. 1691–1715, 1999.

[18] E. Bienenstock, S. Geman, and D. Potter, "Compositionality, mdl priors, and object recognition," *Advances in neural information processing systems*, pp. 838–844, 1997.

[19] M. Revow, C. Williams, and G. Hinton, "Using generative models for handwritten digit recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 6, pp. 592–606, 1996.

[20] D. Mumford, "Pattern theory: a unifying perspective," in *First European congress of mathematics*. Springer, 1994, pp. 187–224.

[21] S. C. Zhu, Y. Wu, and D. Mumford, "Filters, random fields and maximum entropy (frame): Towards a unified theory for texture modeling," *International Journal of Computer Vision*, vol. 27, no. 2, pp. 107–126, 1998.

[22] A. Yali, *2D object detection and recognition: Models, algorithms, and networks*. MIT Press, 2002.

[23] P. Felzenszwalb and D. P. Huttenlocher, "Efficient matching of pictorial structures," in *IEEE conference on Computer Vision and Pattern Recognition*, vol. 2. IEEE, 2000, pp. 66–73.

[24] M. Fischler and R. Elschlager, "The representation and matching of pictorial structures," *IEEE Transactions on Computers*, vol. 22, no. 1, pp. 67–92, 1973.

[25] P. Felzenszwalb and D. Huttenlocher, "Pictorial structures for object recognition," *International Journal of Computer Vision*, vol. 61, no. 1, pp. 55–79, 2005.

[26] M. W. M. C. Burl and P. Perona, "A probabilistic approach to object recognition using local photometry and global geometry," in *European Conference on Computer Vision (ECCV)*. Springer, 1998, pp. 628–641.

[27] R. Fergus, "Visual object category recognition," Ph.D. dissertation, University of Oxford, 2005.

[28] A. Yuille, P. Hallinan, and D. Cohen, "Feature extraction from faces using deformable templates," *International Journal of Computer Vision*, vol. 8, no. 2, pp. 99–111, 1992.

[29] S. Sclaroff, "Deformable prototypes for encoding shape categories in image databases," *Pattern Recognition*, vol. 30, no. 4, pp. 627–641, 1997.

[30] B. Leibe, A. Leonardis, and B. Schiele, "Combined object categorization and segmentation with an implicit shape model," in *European Conference on Computer Vision (ECCV)*. Springer, 2004.

[31] B. Leibe, A. Leonardis, and B. Schiele, "Robust object detection with interleaved categorization and segmentation," *International Journal of Computer Vision*, vol. 77, no. 1-3, pp. 259–289, 2008.

[32] S. Agarwal and D. Roth, "Learning a sparse representation for object detection," in *European Conference on Computer Vision (ECCV)*, 2002, pp. 113–128.

[33] A. Mohan, C. Papageorgiou, and T. Poggio, "Example-based object detection in images by components," *IEEE transactions on pattern analysis and machine intelligence*, vol. 23, no. 4, pp. 349–361, 2001.

[34] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1. IEEE, 2005, pp. 886–893.

[35] D. Ramanan, "Learning to parse images of articulated bodies," in *Advances in neural information processing systems*, 2006, pp. 1129–1136.

[36] D. Ramanan and C. Sminchisescu, "Training deformable models for localization," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1. IEEE, 2006, pp. 206–213.

[37] M. P. Kumar, A. Zisserman, and P. H. S. Torr, "Efficient discriminative learning of parts-based models," in *IEEE International Conference on Computer Vision*. IEEE, 2009, pp. 552–559.

[38] L. Bourdev and J. Malik, "Poselets: Body part detectors trained using 3d human pose annotations," in *International Conference on Computer Vision (ICCV)*, 2009.

[39] L. Bourdev, S. Maji, T. Brox, and J. Malik, "Detecting people using mutually consistent poselet activations," in *European Conference on Computer Vision (ECCV)*, 2010.

[40] M. B. Blaschko and C. H. Lampert, "Learning to localize objects with structured output regression," in *European conference on computer vision (ECCV)*. Springer, 2008, pp. 2–15.

[41] S. Nowozin and C. H. Lampert, "Structured learning and prediction in computer vision," *Foundations and Trends® in Computer Graphics and Vision*, vol. 6, no. 3–4, pp. 185–365, 2011.

[42] Y. Yang and D. Ramanan, "Articulated human detection with flexible mixtures of parts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 12, pp. 2878–2890, 2013.

[43] C. H. Lampert, *Kernel methods in computer vision*. Now Publishers Inc, 2009.

[44] T. Malisiewicz and A. A. Efros, "Recognition by association via learning per-exemplar distances," in *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2008, pp. 1–8.

[45] X. Ren and D. Ramanan, "Histograms of sparse codes for object detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3246–3253.

[46] P. Felzenszwalb, R. Girshick, and D. McAllester, "Cascade object detection with deformable part models," in *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2010, pp. 2241–2248.

[47] L. Bourdev and J. Brandt, "Robust object detection via soft cascade," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2. IEEE, 2005, pp. 236–243.

[48] P. Dollár, Z. Tu, P. Perona, and S. Belongie, "Integral channel features," 2009.

[49] P. Dollár, P. Welinder, and P. Perona, "Cascaded pose regression," in *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2010, pp. 1078–1085.

[50] P. Dollár and L. C. Zitnick, "Structured forests for fast edge detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 1841–1848.

[51] L. C. Zitnick and P. Dollár, "Edge boxes: Locating object proposals from edges," in *European Conference on Computer Vision (ECCV)*. Springer, 2014, pp. 391–405.

[52] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: An evaluation of the state of the art," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 4, pp. 743–761, 2012.

[53] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: A benchmark," in *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 304–311.

[54] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[55] C. Szegedy, A. Toshev, and D. Erhan, "Deep neural networks for object detection," in *Advances in Neural Information Processing Systems*, 2013, pp. 2553–2561.

[56] D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov, "Scalable object detection using deep neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2147–2154.

[57] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "Overfeat: Integrated recognition, localization and detection using convolutional networks," *arXiv preprint arXiv:1312.6229*, 2013.

[58] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 580–587.

[59] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Region-based convolutional networks for accurate object detection and segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 1, pp. 142–158, 2016.

[60] G. Gkioxari, B. Hariharan, R. Girshick, and J. Malik, "R-cnns for pose estimation and action detection," *arXiv preprint arXiv:1406.5212*, 2014.

[61] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik, "Simultaneous detection and segmentation," in *European Conference on Computer Vision*. Springer, 2014, pp. 297–312.

[62] R. Girshick, "Fast r-cnn," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1440–1448.

[63] K. E. A. V. Sande, J. R. R. Uijlings, T. Gevers, and A. W. M. Smeulders, "Segmentation as selective search for object recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2011, pp. 1879–1886.

[64] J. R. R. Uijlings, K. E. A. V. Sande, T. Gevers, and A. W. M. Smeulders, "Selective search for object recognition," *International Journal of Computer Vision*, vol. 104, no. 2, pp. 154–171, 2013.

[65] X. Wang, M. Yang, S. Zhu, and Y. Lin, "Regionlets for generic object detection," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 10, pp. 2071–2084, 2015.

[66] J. Hosang, R. Benenson, P. Dollár, and B. Schiele, "What makes for effective detection proposals?" *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 4, pp. 814–830, 2016.

[67] M.-M. Cheng, Z. Zhang, W.-Y. Lin, and P. Torr, "Bing: Binarized normed gradients for objectness estimation at 300fps," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 3286–3293.

[68] W. Kuo, B. Hariharan, and J. Malik, "Deepbox: Learning objectness with convolutional networks," in *IEEE International Conference on Computer Vision*, 2015, pp. 2479–2487.

[69] Y. Wang and M. Hebert, "Model recommendation: Generating object detectors from few samples," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1619–1628.

[70] H. J. Seo and P. Milanfar, "Action recognition from one example," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 5, pp. 867–882, 2011.

[71] H. J. Seo and P. Milanfar, "Face verification using the lark representation," *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 4, pp. 1275–1286, 2011.

[72] H. Takeda, S. Farsiu, and P. Milanfar, "Kernel regression for image processing and reconstruction," *IEEE Transactions on Image Processing*, vol. 16, no. 2, pp. 349–366, 2007.

[73] C. Kim and P. Milanfar, "Visual saliency in noisy images," *Journal of vision*, vol. 13, no. 4, pp. 5–5, 2013.

[74] O. Z. A. Tefas and I. Pitas, "Visual object tracking based on local steering kernels and color histograms," *IEEE Transactions on Circuits and Systems for video technology*, vol. 23, no. 5, pp. 870–882, 2013.

[75] X. You, Q. Li, D. Tao, W. Ou, and M. Gong, "Local metric learning for exemplar-based object detection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 8, pp. 1265–1276, 2014.

[76] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification (2Nd Edition)*. Wiley-Interscience, 2000.

[77] M. C. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.

[78] P. Milanfar, "A tour of modern image filtering: New insights and methods, both practical and theoretical," *IEEE Signal Processing Magazine*, vol. 30, no. 1, pp. 106–128, 2013.

[79] S. K. Biswas and P. Milanfar, "Laplacian object: One shot object detection with locality preserving projections," in *IEEE Conference on Image Processing*. IEEE, 2014.

[80] S. K. Biswas and P. Milanfar, "One shot detection with laplacian object and fast matrix cosine similarity," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 3, pp. 546–562, 2016.

[81] J. J. Hull, "A database for handwritten text recognition research," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 16, no. 5, pp. 550–554, 1994.

[82] M. M. Tatsuoka, *Multivariate analysis,*. Macmillan, 1988.

[83] T. Caliński, M. Krzyśko, and W. Wołyński, "A comparison of some tests for determining the number of nonzero canonical correlations," *Communications in Statistics, Simulation and Computation*, vol. 35, no. 3, pp. 727–749, 2006.

[84] Y. Benjamini and Y. Hochberg, "Controling the false discovery rate: A practical and powerful approach to multiple testing," *Journal of the Royal Statistical Society. Series B*, vol. 57, no. 1, pp. 289–300, 1995.

[85] F. Devernay, "A non-maxima suppression method for edge detection with sub-pixel accuracy," *Technical report, INRIA*, no. RR-2724, 1995.

[86] H. J. Seo and P. Milanfar, "Static and space-time visual saliency detection by self-resemblance," *Journal of vision*, vol. 9, no. 12, p. 15, 2009.

[87] R. Sznitman and B. Jedynak, "Active testing for face detection and localization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 10, pp. 1914–1920, 07 2010.

[88] C. H. Lampert, M. B. Blaschko, and T. Hofmann, "Beyond sliding windows: Object localization by efficient subwindow search," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[89] C. H. Lampert, M. B. Blaschko, and T. Hofmann, "Efficient subwindow search: A branch and bound framework for object localization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 12, pp. 2129–2142, 2009.

[90] C. Dubout and F. Fleuret, "Exact acceleration of linear object detectors," in *European Coneference on Computer Vision (ECCV)*. Springer, 2012, pp. 301–311.

[91] J. P. Lewis, "Fast normalized cross-correlation," in *Vision Interface*, vol. 10, no. 1, 1995, pp. 120–123.

[92] J. R. Lee and W. A. Nicewander, "Thirteen ways to look at the correlation coefficient," *The American Statistician*, vol. 42, no. 1, pp. 59–66, 1988.

[93] C. H. Lampert, "Detecting objects in large image collections and videos by efficient subimage retrieval," in *IEEE International Conference on Computer Vision (ICCV)*, 2009, pp. 987–994.

[94] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, 2003, pp. 1470–1477.

[95] R. Benenson, M. Omran, J. Hosang, and B. Schiele, "Ten years of pedestrian detection, what have we learned?" in *European Conference on Computer Vision*. Springer, 2014, pp. 613–627.

[96] D. Geronimo, A. M. Lopez, A. D. Sappa, and T. Graf, "Survey of pedestrian detection for advanced driver assistance systems," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 7, pp. 1239–1258, 2010.

[97] M. Teutsch, T. Mueller, M. Huber, and J. Beyerer, "Low resolution person detection with a moving thermal infrared camera by hot spot classification," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. IEEE, June 2014, pp. 209–216.

[98] B. Scholkopf and A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Cambridge, MA, USA: MIT Press, 2001.

[99] D. Olmeda, C. Premebida, U. Nunes, J. Armingol, and A. Escalera, "Lsi far infrared pedestrian dataset," 2013. [Online]. Available: http://e-archivo.uc3m.es/handle/10016/17370

[100] S. Hwang, J. Park, N. Kim, Y. Choi, and I. S. Kweon, "Multispectral pedestrian detection: Benchmark dataset and baseline," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1037–1045.

[101] Z. Wu, N. Fuller, D. Theriault, and M. Betke, "A thermal infrared video benchmark for visual analysis," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2014, pp. 201–208.

[102] J. W. Davis and M. A. Keck, "A two-stage template approach to person detection in thermal imagery." in *Proc. Workshop on Applications of Computer Vision*. IEEE, 2005.

[103] J. W. Davis and V. Sharma, "Background-subtraction using contour-based fusion

of thermal and visible imagery," *Computer Vision and Image Understanding*, vol. 106, no. 2, pp. 162–182, 2007.

[104] F. D. L. Torre, "A least-squares framework for component analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 6, pp. 1041–1055, 2012.

[105] X. He, S. Yan, Y. Hu, P. iyogi, and H.-J. Zhang, "Face recognition using laplacianfaces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 3, pp. 328–340, 2005.

[106] M. W. Mahoney, L. Orecchia, and N. K. Vishnoi, "A local spectral method for graphs: With applications to improving graph partitions and exploring data graphs locally," *Journal of Machine Learning Research*, vol. 13, no. 1, p. 2339, 2012.

[107] S. Maji, N. K. Vishnoi, and J. Malik, "Biased normalized cuts," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 2057–2064.

[108] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering." in *Advances in Neural Information Processing Systems*, vol. 14, 2001, pp. 585–591.

[109] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural Computation*, vol. 15, no. 6, pp. 1373–1396, 2003.

[110] F. F. Li, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories," in *IEEE CVPR Workshop of Generative Model Based Vision (WGMBV)*, 2004.

[111] F. F. Li, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories," *Computer Vision and Image Understanding*, vol. 106, no. 1, pp. 59–70, 2007.

[112] F. F. Li, R. Fergus, and P. Perona, "One-shot learning of object categories," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 4, pp. 594–611, 2006.

[113] K. Chatfield, J. Philbin, and A. Zisserman, "Efficient retrieval of deformable shape classes using local self-similarities," in *Proceedings of IEEE ICCV Workshops*, 2009, pp. 264–271.

[114] D. Tao, X. Li, X. Wu, W. Xindong, W. Hu, and S. J. Maybank, "Supervised tensor learning," *Knowledge and information systems*, vol. 13, no. 1, pp. 1–42, 2007.

[115] M. A. O. Vasilescu and D. Terzopoulos, "Multilinear analysis of image ensembles: Tensorfaces," in *European Conference on Computer Vision*. Springer, 2002, pp. 447–460.

[116] T. G. Kolda and B. W. Bade, "Tensor decompositions and applications," *SIAM review*, vol. 51, no. 3, pp. 455–500, 2009.

[117] A. D. Jepson, D. J. Fleet, and T. F. El-Maraghi, "Robust online appearance models for visual tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 10, pp. 1296–1311, 2003.

[118] X. Mei and H. Ling, "Robust visual tracking using $l1$ minimization," in *IEEE International Conference on Computer Vision (ICCV)*. IEEE, 2009, pp. 1436–1443.

[119] S. Avidan, "Support vector tracking," *IEEE transactions on pattern analysis and machine intelligence*, vol. 26, no. 8, pp. 1064–1072, 2004.

[120] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 25, no. 5, pp. 564–577, 2003.

[121] M. Isard and A. Blake, "CondensationâĂŤconditional density propagation for visual tracking," *International Journal of Computer Vision*, vol. 29, no. 1, pp. 5–28, 1998.

[122] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 24, no. 5, pp. 603–619, 2002.

[123] B. Babenko, M.-H. Yang, and S. Belongie, "Visual tracking with online multiple instance learning," in *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 983–990.

[124] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-learning-detection," *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 7, pp. 1409–1422, 2012.

[125] P. Sermanet, K. Kavukcuoglu, S. Chintala, and Y. LeCun, "Pedestrian detection with unsupervised multi-stage feature learning," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3626–3633.

[126] W. Ouyang and X. Wang, "Joint deep learning for pedestrian detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 2056–2063.

[127] R. Benenson, M. Mathias, T. Tuytelaars, and L. V. Gool, "Seeking the strongest rigid detector," in *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2013, pp. 3666–3673.

[128] R. Girshick, F. Iandola, T. Darrell, and J. Malik, "Deformable part models are convolutional neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 437–446.