MULTI-CHANNEL CORRELATION FILTERS WITH LIMITED BOUNDARIES: THEORY AND APPLICATIONS

HAMED KIANI GALOOGAHI

(M.Sc., Amirkabir University of Technology, 2009)

A THESIS SUBMITTED FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

DEPARTMENT OF COMPUTER SCIENCE
NATIONAL UNIVERSITY OF SINGAPORE

2014

Declaration

I hereby declare that this thesis is my original work and it has been written by me in its entirety. I have duly acknowledged all the sources of information which have been used in the thesis.

This thesis has also not been submitted for any degree in any university previously.

H Kiani

Hamed Kiani Galoogahi

20 August 2014

Acknowledgments

This dissertation would not have been possible without the support and guidance of many people who contributed and extended their valuable assistance in the preparation and completion of this study.

Foremost, I would like to express my greatest appreciations to my supervisor professor Terence Sim. This thesis would not have been possible without his help, support and patience, not to mention his advice and unsurpassed knowledge of the related subjects. He has been invaluable on both an academic and a personal level, for which I am extremely grateful.

I would like to acknowledge the financial, academic and technical support of the National University of Singapore and Department of Computer Science particularly in the award of a Postgraduate Research Scholarship that provided the necessary financial support for my research.

I am indebted to Prof. Simon Lucey for giving me the opportunity to work under him in Australia and exposing me to new problems in the field of computer vision.

Last but not least, I wish to thank all of my family for their unconditional support, encouragement and love throughout my life. My gratitude towards them is truly beyond words.

Contents

Abstract

Correlation filters have been widely used in computer vision for pattern detection and recognition. The core idea of all correlation filters is to learn a filter/template that produces desired correlation outputs when correlated with a set of training examples. Correlation filters exhibit a number of characteristics that make them interesting to the vision community, e.g. shift-invariance, robustness to noise, closed-form solutions and most importantly their memory and computation efficiencies. In spite of recent progress in correlation filters, there remains plenty of scope for new extensions and improvements of traditional correlation filters for vision problems. In this research, we introduce the following improvements to the correlation filter theory for vision applications. First, traditional correlation filters are limited to single-channel image representations (e.g. pixel intensities). We propose an extension to canonical correlation filter theory that is able to handle multi-channel signals/features, which refereed to as multi-channel correlation filters. This allows one to exploit modern image descriptors (e.g HOG and SIFT) to learn discriminative filters for challenging pattern classification and detection. Second, we demonstrate that multi-channel correlation filters can be directly applied to learn spatial-temporal patterns in videos with no extra memory and computation overheads. Third, traditional correlation filters employ shifted patches for filter training which implicitly are created by circular boundary effects. These shifted patches are not representative of real patches and can drastically reduce the discrimination power of the trained filter. We propose correlation filters with limited boundaries that can significantly reduce the number of patches affected by boundary effects. Finally, we propose to apply a set of multi-channel correlation filters with different spatial supports over a cascaded framework for coarse-to-fine facial landmark detection. We demonstrate the superior performance, memory and computation efficiencies of all the proposed techniques in this thesis over an extensive set of experiments including visual object tracking, object localization, human action recognition and robust facial landmark detection.

List of Figures

List of Tables

Chapter 1

Introduction

Pattern recognition has been becoming essential for thousands of practical vision applications ranging from robotics, surveillance, biometrics, image retrieval, scene understanding, video analysis, health care etc. The goal of visual pattern recognition is to automatically recognize instances of patterns (e.g. objects) in image or video. In spite of significant progress addressing this problem over the past few decades, current pattern recognition systems are not able to accurately and quickly recognize and categorize patterns under challenging real-world circumstances.

The trade-off between accuracy and computational efficiency is the central issue in all pattern recognition approaches. Accuracy can be adversely affected by photometric/geometric changes, background clutter, occlusion, large intra-class variations and inter-class similarities. Many efforts have been devoted to perform robust pattern recognition by using invariant and discriminative image descriptors (e.g. SIFT and HOG [Dalal and Triggs, 2005] [Lowe, 2004]), extracted from a huge amount of training examples, in conjunction with sophisticated statistical learning techniques such as Neural Networks, Support Vector Machines (SVMs) and various boosing techniques [Viola and Jones, 2001; Dalal and Triggs, 2005].

The main disadvantage with these techniques, however, is the large memory and computational overheads required for learning over modest size of training set. From a practical perspective, learning a SVM classifier using HOG features [Viola and Jones, 2001], which has been extensively employed for recognition tasks, incurs a memory cost linear in the number of samples. Whilst this seems reasonable at a glance, consider a simple example of storing $200,000$ 50×50 images in double precision. In the case of raw pixels this amounts to only 4 GB of storage, a manageable figure on current desktop hardware. Using Gabor filter banks of 40 channels (e.g. 5 scales and 8 orientations when using oriented edge energies), storage blows out to an untenable 160 GB. Strategies have been proposed to save storage complexity, however they are largely based on heuristic subsampling of the resolution of image descriptors, or the number of training samples.

Besides, nearly every state-of-the-art detector employs sliding window paradigm in the spatial domain to detect a pattern of interest in images [Dalal and Triggs, 2005; Felzenszwalb et al., 2010; Lampert et al., 2008; Viola and Jones, 2001]. In sliding window approaches, a classifier which is trained using a set of positive/negative training examples is evaluated on every possible window in a test image. The brute-force search of all sliding windows over a set of scales would be computationally intractable (e.g. 90000 evaluations to process a 150×150 image over 4 different scales) and, consequently, makes the detection process extremely slow [Vedaldi and Zisserman, 2012; H. Harzallah and Schmid, 2009].

Canonical correlation filters, on the other hand, enjoy mathematical simplicity, computational efficiency and tractable memory usage [Bolme et al., 2010]. Correlation filters, developed initially in the seminal work of Hester and Casasent [Hester and Casasent, 1980] are a method for learning a template/filter in the frequency domain which have been widely employed for pattern detection in images and videos. Although many variants have been proposed over the last three decades [Hester and Casasent, 1980; Kumar, 1986; Mahalanobis et al., 1987; Kumar, 2005; Bolme et al., 2009, 2010], the approach's central tenet is to learn a filter/template instead of a simple cropped example, that when correlated with a set of training signals, returns corresponding desired responses (typically a peak response at the location of the object, while the response in all other regions of the correlation plane are close to zero). Like correlation itself, one of the central advantages of correlation filters is that they attempt to learn the filter in the frequency domain due to computational efficiency of correlation in that domain. Shift invariance, high tolerance to noise, robustness to partial occlusions, stability against spatial translations and closed-form solution are the other interesting characteristics of correlation filters.

Interest in correlation filters has been reignited in the vision world through the recent work of Bolme et al. [2010] on Minimum Output Sum of Squared Error (MOSSE) correlation filters for object detection and tracking. This work was able to circumvent some of the classical problems with earlier correlation filters (e.g. over-fitting) and performed well in adaptive tracking under changes in rotation, scale, lighting and partial occlusion. A central strength of the MOSSE correlation filter is that it is extremely efficient in terms of required learning memory and computations. It was stated in Bolme et al. [2010] that the amount of memory required to learn a MOSSE filter is independent of the number of training images. This allows one to use a huge amount of training examples with no concern for memory limitation to learn a well-generalized correlation filter. In addition to the memory efficiency, the computational cost of detecting a pattern using correlation filters (specially in the Fourier domain) is very low, making this class of detectors very appropriate for real-time detection tasks (e.g. visual object tracking with a superior speed of 650 *fps* [Bolme et al., 2010]).

Current correlation filter, however, are not able to handle some classical issues in pattern recognition/detection which have been successfully addressed by the stat-of-the-art non-filter detectors which employ modern image descriptors (e.g. HOG and SIFT) along with discriminative machine learning techniques. The main objective of this thesis is to improve existing correlation filters for robust and invariant pattern detection/recognition in challenging situations, while preserving their superior detection speed, memory and computation efficiencies.

1.1 Thesis Contributions

In this thesis we present several improvements and extensions to current correlation filter techniques. More specifically, the main contributions of this thesis are as follows:

1. Traditional correlation filters have employed single-channel image representation (e.g. intensity) for filter learning. It has been well noted in the

vision literature that these simple features do not perform well for pattern detection/recognition in unconstrained circumstances. In the first part of this work, we propose an extension to canonical correlation filter theory that is able to learn well-generalized multi-channel correlation filters using invariant and discriminate multi-channel image descriptors, called Multi-Channel Correlation Filters (MCCF). We show how multi-channel correlation filter estimation in the frequency domain forms a sparse-banded linear system which can be efficiently solved through a novel variable reordering technique. Specifically, we demonstrate how our approach does not have a memory cost that is linear in the number of samples, allowing for substantial savings when learning detectors across large amounts of data.

2. In the second part of this thesis, we argue that shifted patches generated over the circular property of correlation drastically degrade the detection performance of current correlation filters. To deal with this drawback, a new correlation filter objective is proposed that can remarkably reduce the number of learning examples affected by boundary effects. Specifically, we demonstrate how this new objective can be efficiently optimized in an iterative manner through an Augmented Lagrangian Method (ALM) to take advantage of efficient correlation in the frequency domain. Moreover, we show that this new objective can be combined with MCCF's objective to address the problem of single-channel feature and the boundary effects simultaneously.

3. We evaluate the proposed correlation filters across a myriad of vision applications including, visual object tracking, object detection, facial landmark localization and video analysis. Particularly, we show that multi-channel correlation filters are not limited to images (space domain) and can be easily applied on video patterns (spatial-temporal domain) with no extra memory and computation overheads. we specifically evaluate the MCCF for human action recognition in video data, where the experimental result shows the superior detection speed and memory usage of our method

versus the state-of-the-art with very competitive recognition performance.

4. Finally, a cascaded correlation filters framework is proposed for global-to-local facial landmarks detection in face images with high stability against face pose and expression. Over the experiments, we demonstrate very competitive performance of our cascaded framework compared to current state-of-the-art landmark detectors, with superior detection speed, computational and memory efficiencies over the LFPW and BioID datasets.

1.2 Notations

In this thesis vectors are always presented in lower-case bold (e.g., $\mathbf{a}$), Matrices are in upper-case bold (e.g., $\mathbf{A}$) and scalars in italicized (e.g. a or A). $\mathbf{a}(i)$ refers to the ith element of the vector $\mathbf{a}$. All M-mode array signals shall be expressed in vectorized form $\mathbf{a}$. We shall be assuming $M = 2$ mode matrix signals (e.g. $2D$ image arrays) in nearly all our discussions throughout this work. A 2D matrix can be easily vectorized by concatenating its columns into a 1D vector. A M-mode convolution operation is represented as the $*$ operator. One can express a M-dimensional discrete circular shift $\Delta\boldsymbol{\tau}$ to a vectorized M-mode matrix $\mathbf{a}$ through the notation $\mathbf{a}[\Delta\boldsymbol{\tau}]$. The matrix $\mathbf{I}$ denotes a $D\times D$ identity matrix and $\mathbf{1}$ denotes a D dimensional vector of ones. A $\hat{}$ applied to any vector denotes the M-mode Discrete Fourier Transform (DFT) of a vectorized M-mode matrix signal $\mathbf{a}$ such that $\hat{\mathbf{a}} \leftarrow \mathcal{F}(\mathbf{a}) = \sqrt{D}\mathbf{F}\mathbf{a}$. Where $\mathcal{F}()$ is the Fourier transforms operator and $\mathbf{F}$ is the orthonormal $D \times D$ matrix of complex basis vectors for mapping to the Fourier domain for any D dimensional vectorized image/signal. We have chosen to employ a Fourier representation through this thesis due to its particularly useful ability to represent circular convolutions as a Hadamard product in the Fourier domain. Additionally, we take advantage of the fact that $\text{diag}(\hat{\mathbf{h}})\hat{\mathbf{a}} = \hat{\mathbf{h}} \circ \hat{\mathbf{a}}$, where $\circ$ represents the Hadamard product, and diag() is an operator that transforms a D dimensional vector into a $D \times D$ dimensional diagonal matrix. The role of filter $\hat{\mathbf{h}}$ or signal $\hat{\mathbf{a}}$ can be interchanged with this property. Any transpose operator $^\top$ on a complex vector or matrix additionally takes the complex conjugate in a similar fashion to the Hermitian adjoint [Kumar, 2005]. The operator $\text{conj}(\hat{\mathbf{a}})$ applies the complex conjugate to the complex vector $\hat{\mathbf{a}}$.

Chapter 2

Background

2.1 Signal Correlation

Correlation is a standard operation to measure signal similarity. The correlation between two given signals $\mathbf{z}$ and $\mathbf{h}$ is defined as:

$$\mathbf{y} = \mathbf{z} \star \mathbf{h} \tag{2.1}$$

where $\star$ denotes the correlation operator and $\mathbf{y}$ is the correlation output. The discrete correlation of two one-dimensional signals $\mathbf{z}$ and $\mathbf{h}$ is computed as:

$$(\mathbf{z} \star \mathbf{h})[x] = \sum_{k=-\infty}^{\infty} \mathbf{z}[k]\mathbf{h}[x+k] \tag{2.2}$$

which can be extended for two-dimensional signals as,

$$(\mathbf{z} \star \mathbf{h})(x,y) = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} \mathbf{z}[k,l]\mathbf{h}[x+k,y+l] \tag{2.3}$$

Signal correlation is similar to convolution except that one signal is time-reversed. In signal processing, convolution of two signals $\mathbf{z}$ and $\mathbf{h}$ is represented as:

$$\mathbf{y} = \mathbf{z} * \mathbf{h} \tag{2.4}$$

$$(\mathbf{z} * \mathbf{h})[x] = \sum_{k=-\infty}^{\infty} \mathbf{z}[k]\mathbf{h}[x-k] \tag{2.5}$$

where $*$ denotes the convolution operator. Similar to correlation, two-dimensional signal convolution can be defined as:

$$(\mathbf{z} * \mathbf{h})(x, y) = \sum_{dx=-\infty}^{\infty} \sum_{dy=-\infty}^{\infty} \mathbf{z}[dx, dy]\mathbf{h}[x - dx, y - dy] \tag{2.6}$$

where the signal $\mathbf{x}$ is time-reversed.

2.2 The Convolution Theorem

The convolution theorem states that convolution in the spatial domain can be computed as Hadamard product in the Fourier domain [Bracewell and Bracewell, 1986],

$$\mathbf{y} = \mathbf{z} * \mathbf{h} = \mathcal{F}^{-1}(\hat{\mathbf{z}} \circ \hat{\mathbf{h}}) \tag{2.7}$$

The power of the above equation lies in its computational efficiency, as it simplifies the computationally intensive operations of convolution in the spatial domain (including circular-shift, multiplication and summation) with an element-wise Hadamard product in the Fourier domain [see Section 2.3 for more details].

Using the convolution theorem, correlation in the Fourier domain can be defined as:

$$\mathbf{y} = \mathbf{z} \star \mathbf{h} = \mathcal{F}^{-1}(\hat{\mathbf{z}} \circ conj(\hat{\mathbf{h}})) \tag{2.8}$$

where $conj(.)$ is the complex conjugate that reverses a signal, and used to ensure that the operation is correlation not convolution.

2.3 Correlation Complexity

According to Equation 2.2, correlating two signals of length N and P in the spatial domain takes $\mathcal{O}(NP)$. Likewise, the complexity of correlating two images of size $N \times M$ and $P \times Q$ is $\mathcal{O}(NMPQ)$. Using the Convolution theorem, correlation can be performed by Hadamard product of two signals in the Fourier domain. The FFT/IFFT and Hadamard product can be respectively computed in $\mathcal{O}(N \log N)$ and $\mathcal{O}(N)$ [Cooley and Tukey, 1965]. This results in a total computation of $\mathcal{O}(N \log N)$ for one-dimensional correlation. Similarly, we can show that the two-dimensional correlation in the Fourier domain can be performed in $\mathcal{O}(NM \log NM)$.

In the spatial domain, two-dimensional correlation can be computed faster as long as one signal is separable, meaning it can be defined as outer product of two vectors [Breiman, 1996]. A two-dimensional signal $\mathbf{H}$ is separable if $\mathbf{H} = \mathbf{h}_1 \otimes \mathbf{h}_2$, where $\otimes$ indicates the outer product of two vectors. Thus, the separable spatial correlation between two two-dimensional signals $\mathbf{H}$ and $\mathbf{Z}$ is calculated as:

$$\begin{aligned} \mathbf{Y} = \mathbf{Z} \star \mathbf{H} = \mathbf{Z} \star (\mathbf{h}_1 \otimes \mathbf{h}_2) &= (\mathbf{Z} \star \mathbf{h}_1) \star \mathbf{h}_2 \\ &= (\mathbf{Z} \star \mathbf{h}_2) \star \mathbf{h}_1 \end{aligned} \tag{2.9}$$

For example, some basic image processing filters such as 3×3 Sobel operator can be broken into two 3×1 and 1×3 one-dimensional vectors, $\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 & -1 \end{bmatrix}$. In this case, a two-dimensional correlation in the spatial domain can be performed in two steps for a total cost of $\mathcal{O}(NM(P+Q))$. This involves correlating the image with the first vector in $\mathcal{O}(NMP)$, and then correlating the output with the second vector in $\mathcal{O}(NMQ)$.

From the discussion above, it is necessary to have prior knowledge about signals

Correlation schema	Complexity
Spatial Correlation	$\mathcal{O}(NMPQ)$
Separable Correlation	$\mathcal{O}(NM(P+Q))$
Fourier Correlation	$\mathcal{O}(NM \log NM)$

Table 2.1: Comparing the complexity of three different correlation schemas.

to choose the appropriate correlation/convolution schema. For example, spatial correlation of an $M \times N$ image with a $P \times Q$ filter is faster than correlation in Fourier domain if $P \times Q < \log NM$. Likewise, separable correlation is always faster than spatial correlation. A comparison of computational costs of three different correlation schemas are shown in Table 2.1, where the computation of spatial correlation depends on both the size of the image and the filter, while the complexity of FFT correlation is based on the size of the image, assuming that the image is larger than the filter.

2.4 Detection by Template Matching

In practice, correlation calculates the inner product of two signals and provides a measure of signal similarity/dissimilarity which has been commonly used in many signal processing problems. Assume that two signals $\mathbf{h}$ and $\mathbf{z}$ are given, where $\mathbf{h}$ is the offset version of $\mathbf{z}$ by an unknown time lag and both are affected by random noise. The problem is to estimate the time lag between these two signals by signal matching. This can simply done by signal correlation, $\mathbf{y} = \mathbf{z} \star \mathbf{h}$. The correlation output $\mathbf{y}$ is a new signal whose values are the inner product between the signal $\mathbf{h}$ and all possible translated versions of $\mathbf{z}$. Therefore, each discrete value of $\mathbf{y}$ shows how similar that part of signal $\mathbf{z}$ is to the signal $\mathbf{h}$. The global maximum (peak) in $\mathbf{y}$ finds the match between of the signals $\mathbf{h}$ and $\mathbf{z}$ and indicates the amount of time offset.

In vision communities, signal correlation is used as a common solution to many pattern detection/matching problems. The basic idea of detection-by-correlation is that the pattern of visual appearance can be captured by an image template. Like one-dimensional correlation, correlation between a template and an input image returns a new image where each pixel value indicates the amount of similarity between the template and the shifted versions of the image. By thresholding the local maximums over the correlation output, one can simply determine the presence and location of the pattern of interest in images.

Current template matching techniques differ mainly in the way they design the correlation template. The simplest way is cropping an example of the pattern from training images. This, however, performs well when all images are captured under heavily controlled situations, to ensure that the template and images have very similar appearance. This is applicable in some applications such as industrial monitoring, video stabilization and rigid image registration where images are taken under fixed or slightly different viewpoint and lighting condition, and have similar visual characteristics in common.

This, however, is not practical in more complicated applications. In object detection, for example, images belong to the same class of objects might look drastically different (intra-class variations), even under the same imaging conditions. On the other hand, images taken from different objects of different classes under different imaging conditions can appear to be very similar (inter-class similarities).

To deal with these difficulties, therefore, correlation filters have been proposed to learn templates/filters from a set of training samples, instead of cropping a single raw example from a training image, where the template is referred to as a *correlation filter*. In the next subsection, we will briefly review current correlation filter techniques and their advantages and disadvantages for pattern detection and matching.

2.5 Overview of Existing Correlation Filters

To date, several correlation filters techniques have been proposed. In this section, we will review more common techniques in the literature and evaluate their advantages and disadvantage for pattern detection, localization and recognition.

2.5.1 Synthetic Discriminant Functions

The first correlation filter technique, called Synthetic Discriminant Functions (SDF), was proposed to learn correlation filters form a set of training images [Casasent and Chang, 1986]. Given a set of N vectorized training images and their corresponding desired outputs $\{\mathbf{x}_i, u_i\}_{i=1}^{N}$, the objective of SDF is training a filter $\mathbf{h}$ that satisfies a set of hard constraints on the correlation outputs,

$$\mathbf{x}_i^\top \mathbf{h} = u_i, \quad for \;\; i = 1, ..., N. \tag{2.10}$$

when the number of training images are less than the number of constraints (filter's dimension), there are many filters that satisfy the hard constraints in Equation 2.10. To learn a single filter, therefore, the SDF technique requires the filter $\boldsymbol{h}$ to be a linear combination of all training images.

Therefore, the optimal SDF filter is obtained by minimizing the following objective function in the spatial domain:

$$\begin{aligned} \mathbf{h}^* &= \arg\min \mathbf{h}^\top \mathbf{h} \qquad (2.11) \\ &\text{s.t.} \quad \mathbf{X}^\top \mathbf{h} = \mathbf{u} \end{aligned}$$

where $\mathbf{u} = [u_1, u_2, ..., u_N]$ denotes the desired correlation outputs corresponding to training images, $u_i = 1$ for positive and $u_i = 0$ for negative examples, and $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N]$ is a $D \times N$ matrix containing all N vectorized training images of length D. The optimal SDF filter in the spatial domain is calculated as:

$$\mathbf{h}^* = \mathbf{X}(\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{u} \tag{2.12}$$

Complexity and memory analysis: The memory required to train a SDF correlation filter over N training images of length D is $\mathcal{O}(ND)$, because one needs to hold all training examples X in memory. The training complexity consists a computational cost of $\mathcal{O}(N^3)$ for a $N \times N$ matrix inversion.

2.5.2 Minimum Average Correlation Energy Filters

Later on, the Minimum Average Correlation Energy Filter (MACE) technique was introduced to address the problem of smooth peaks in SDF. It learns a filter that produces a sharp peak at the origin of the correlation plane [Mahalanobis et al., 1987]. This is done by minimizing average correlation energy in the Fourier domain subject to the hard constraints on correlation output in Equation 2.10:

$$\begin{aligned} \hat{\mathbf{h}}^* &= \arg\min \hat{\mathbf{h}}\hat{\mathbf{D}}\hat{\mathbf{h}} \\ \text{s.t.} & \quad \hat{\mathbf{X}}^\top \hat{\mathbf{h}} = \mathbf{u} \end{aligned} \tag{2.13}$$

The optimal solution of a MACE filter is calculated as:

$$\hat{\mathbf{h}}^* = \hat{\mathbf{D}}^{-1}\hat{\mathbf{X}}(\hat{\mathbf{X}}^\top\hat{\mathbf{D}}^{-1}\hat{\mathbf{X}})^{-1}\hat{\mathbf{u}} \tag{2.14}$$

where $\hat{\mathbf{X}}$ contains vectorized training images in the Fourier domain and $\hat{\mathbf{D}}$ is a diagonal matrix containing their average power spectrum. It has been shown that the MACE filter suffers from sensitivity to noise and typically produces sharp peaks for the images it was trained on (over-training).

Complexity and memory analysis: Inspecting Equation 2.14, one can see that the memory required to train a MACE correlation filter is $\mathcal{O}(ND)$, where N and D respectively indicate the number and length of training images, $X_{D\times N}$. The overall computations for training a MACE filter are:

$$\underbrace{N^3}_{\text{matrix inversion}} + \underbrace{D^3}_{\text{matrix inversion}} + \underbrace{ND\log D}_{\text{Fourier transforms}} = \mathcal{O}(\max\{D^3, N^3\}) \qquad (2.15)$$

2.5.3 Minimum Variance Synthetic Discriminate Functions

The Minimum Variance Synthetic Discriminate Function (MVSDF) technique dealt with SDF noise sensitivity by minimizing noise covariance while satisfying the hard constraints on correlation outputs [Kumar, 1986]. The MVSDF filter was formulated the same as the MACE filter except that the matrix $\hat{\mathbf{D}}$ in Equation 2.14 contains the noise power spectrum. Because estimating the color of noise often is not practical, the noise is assumed to be white. In this case, the matrix $\hat{\mathbf{D}}$ becomes the identity matrix $\mathbf{I}$ and SDF and MVSDF filters are equivalent (except that the SDF is formulated in the spatial domain, but the MVSDF is defined in the Fourier domain), and still suffer from smooth peaks, poor generalization and over-training.

2.5.4 Optimal Trade-off Filters

Finally, Optimal Trade-off Filters (OTF)[Refregier, 1991] are a trade-off between the peak sharpness of MACE and the noise tolerance of MVSDF by defining the matrix $\hat{\mathbf{D}}$ in Equation 2.14 as:

$$\hat{D} = \alpha \hat{D}_1' + (1 - \alpha)\hat{D}_2' \qquad (2.16)$$

where $\hat{\mathbf{D}}_1^{'}$ and $\hat{\mathbf{D}}_2^{'}$ are respectively the matrix $\hat{\mathbf{D}}$ in MACE and SVMDF, and $0 \leqslant \alpha \leqslant 1$ is the trade-off variable. For α equal to 0 and 1, the OTF filter respectively is equivalent to SVMDF and MACE filters. The computational cost and memory usage of MVSDF and OTF techniques are same as the MACE correlation filter.

2.5.5 Unconstrained Correlation Filters

All the aforementioned correlation filters including SDF, MACE, MVSDF and OTF are similar in the way that they train a single correlation filter subject to a set of hard constraints on correlation outputs. This resulted in poor generalization for unseen images and over-fitting for training images [Kumar, 2005]. It has been proposed by [Mahalanobis et al., 1987] that these limitations can be improved by eliminating the hard constraints, and instead requiring the filter to produce a high average response to all training images in the Fourier domain. This technique is generally called unconstrained correlation filters and is defined as:

$$\hat{\mathbf{h}}^{*} = \hat{\mathbf{D}}^{-1}\hat{\mathbf{m}} \tag{2.17}$$

where $\hat{\mathbf{m}}$ is the average of training images in the Fourier domain and $\hat{\mathbf{D}}$ is defined the same as in Equation 2.16. This technique is also known as a Maximum Average Correlation Height (MACH) filter [Mahalanobis et al., 1987]. According to Equations 2.17 and (2.16), if $\alpha = 0$ then $D = I$ and the MACH filter is the average of training images, and the Unconstrained Minimum Average Correlation Energy filter (UMACE) in the case of $\alpha = 1$ [Mahalanobis et al., 1994]. The main issue of unconstrained correlation filters is that these techniques do not employ non-target examples for filter training. Besides, this technique does not determine the correlation outputs for target examples. These may lead to strong peaks upon non-target patches and, consequently, affect the detection/localization accuracy.

Complexity and memory analysis: According to Equation 2.17, unconstrained correlation filters just compute average of training examples in the Fourier domain, instead of holding all training images in memory. This causes substantial memory saving versus SDF, MACE, MVSDF and OTF techniques, a constant $\mathcal{O}(D)$ compared to $\mathcal{O}(ND)$ which is linear in the number of training examples. Training an unconstrained correlation filter over N training examples takes $\mathcal{O}(D^3)$ computations:

$$\underbrace{D^3}_{\text{matrix inversion}} + \underbrace{ND\log D}_{\text{Fourier transforms}} = \mathcal{O}(D^3). \tag{2.18}$$

2.5.6 Average of Synthetic Exact Filters

Recently, Bolme et al. [2009] introduced a new correlation filter technique called the Average of Synthetic Exact Filters (ASEF) [Bolme et al., 2009]. The ASEF technique trains correlation filters that return Gaussian-like correlation output when correlated upon training images, with a peak value of 1.0 at target location (i', j') and near to zero values elsewhere. A Gaussian function was employed to define the $(i,j)^{th}$ location of correlation output as:

$$\mathbf{y}_{i,j} = e^{-\frac{(i-i')^2+(j-j')^2}{\delta^2}} \tag{2.19}$$

where δ specifies the spatial bandwidth of the Gaussian function.

The main idea behind ASEF is that for each vectorized training image $\mathbf{x}_i$ and its corresponding desired output $\mathbf{y}_i$, there is a unique filter $\mathbf{h}_i$ that exactly transforms $\mathbf{x}_i$ into $\mathbf{y}_i$ such that:

$$\mathbf{y}_i = \mathbf{x}_i \star \mathbf{h}_i \tag{2.20}$$

According to the Convolution theorem, the above equation can be expressed in the Fourier domain as:

$$\hat{\mathbf{y}}_i = \hat{\mathbf{x}}_i \circ conj(\hat{\mathbf{h}}_i) \tag{2.21}$$

the optimal filter $\hat{\mathbf{h}}_i$ is called the *Exact* filter, and is calculated as:

$$\hat{\mathbf{h}}_i = \frac{\hat{\mathbf{y}}_i}{conj(\hat{\mathbf{x}}_i)} \tag{2.22}$$

Given a set of Exact filters trained based on N training images $\{\hat{\mathbf{x}}_i\}_{i=1}^N$ and their corresponding outputs $\{\hat{\mathbf{y}}_i\}_{i=1}^N$, Bolme et al. [2009] computed the Average of Synthetic Exact Filter (ASEF) as,

$$\hat{\mathbf{h}} = \frac{1}{N}\sum_{i=1}^{N}\hat{\mathbf{h}}_i = \frac{1}{N}\sum_{i=1}^{N}\frac{\hat{\mathbf{y}}_i}{conj(\hat{\mathbf{x}}_i)} \tag{2.23}$$

2.5.7 Minimizing the Output Sum of Squared Error Filters

The major limitation of the ASEF filter is that thousands of training examples are required to produce a well-generalized correlation filter [Bolme et al., 2010]. More recently, Bolme et al. [2010] proposed that this drawback of ASEF can be improved by learning a correlation filter that minimizes the sum of squared error between the desired and the actual correlation outputs in the Fourier domain, instead of averaging a set of Exact filters. This technique called Minimizing the Output Sum of Squared Error (MOSSE) is defined as:

$$\begin{aligned}\hat{\mathbf{h}} &= \arg\min_{\hat{\mathbf{h}}}\sum_{i=1}^{N}\|\hat{\mathbf{x}}_i \circ conj(\hat{\mathbf{h}}) - \hat{\mathbf{y}}_i\|_2^2 \qquad (2.24)\\ &= \arg\min_{\hat{\mathbf{h}}}\sum_{i=1}^{N}\|diag(\hat{\mathbf{x}}_i)^\top\hat{\mathbf{h}} - \hat{\mathbf{y}}_i\|_2^2\end{aligned}$$

where *diag(.)* converts a vector into a diagonal matrix. By solving the above objective function, the optimal filter $\hat{\mathbf{h}}$ is calculated in the closed-form of:

$$\hat{\mathbf{h}} = [diag(\hat{\mathbf{s}}_{xx})]^{-1}\sum_{i=1}^{N}diag(\hat{\mathbf{x}}_i)\hat{\mathbf{y}}_i = \frac{\hat{\mathbf{s}}_{xy}}{\hat{\mathbf{s}}_{xx}} \tag{2.25}$$

where,

$$\hat{\mathbf{s}}_{xx} = \sum_{i=1}^{N} \hat{\mathbf{x}}_i \circ conj(\hat{\mathbf{x}}_i) \quad \& \quad \hat{\mathbf{s}}_{xy} = \sum_{i=1}^{N} \hat{\mathbf{y}}_i \circ conj(\hat{\mathbf{x}}_i) \tag{2.26}$$

are the average auto-spectral and cross-spectral energies of the training images.

Complexity and memory analysis: According to Equations 2.25 and 2.23, training MOSSE and ASEF correlation filters requires $\mathcal{O}(D)$ constant memory independent of number of training examples. Since, these methods just need to hold the average of Exact filters (ASEF) and the average auto-spectral and cross-spectral energies of training images (MOSSE) in memory. From a complexity perspective, these techniques enjoy low computations of $\mathcal{O}(ND \log D)$ to compute the Fourier transforms of N training examples ($\mathcal{O}(D \log D)$ for each example).

As mentioned above, the ASEF filter generalizes well if it is provided with a large number of training examples. This disadvantage of ASEF technique is illustrated by Figure 2.1, where ASEF and MOSSE filters are evaluated on the problem of right eye localization in face images. The localization rate is presented as a function of number of training examples (the details of training and testing is clearly described in Section 5.1.8). According to Figure 2.1(a), the ASEF technique performs does not perform well when a few training images are provided and its localization rate gradually improves when the amount of training images increases. In contrast, the MOSSE technique achieves a high localization rate using only four training images and its accuracy is slightly changed by increasing the size of training set. Some examples of the trained filters used in this evaluation are shown in Figure 2.1(b).

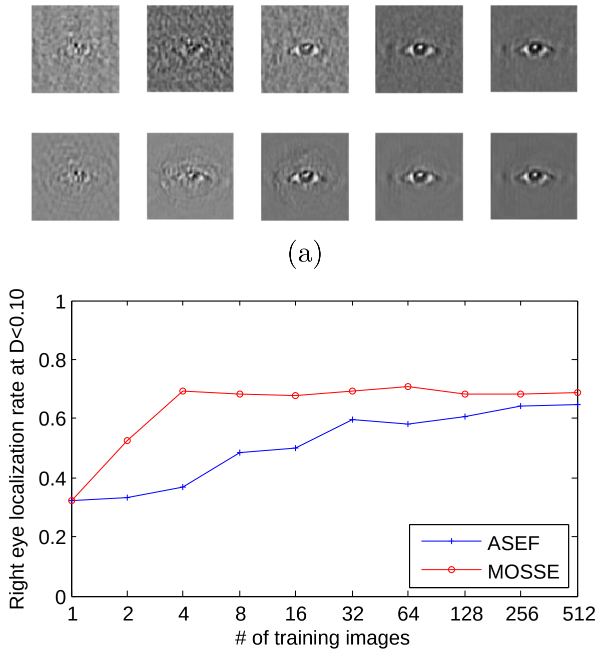
Although the low generalization of the ASEF is improved by the MOSSE technique, there still remain some drawbacks which have not been addressed by the MOSSE and ASEF methods. First, both of these methods explicitly use circular shifted versions of target patches as non-target examples. These shifted patches are produced over the circular property of correlation operation and are not representative of real-world non-target patches, as illustrated for the right eye in Figure 2.2. This may result in training correlation filters which produce strong peaks over real non-target patches. Second, MOSSE and ASSEF techniques directly train correlation filters using raw image intensities, which are not robust to lighting changes, intra class variations, and inter class similarities. A full explanations of these two disadvantages will be provided in the following chapters.

In Figure 2.3, we illustrate three examples of right eye localization using MOSSE correlation filter. The left example shows a successful localization, where the peak upon the right eye is the maximum peak over entire correlation plane. A wrong eye localization is shown in the middle example, where the maximum correlation peak occurs upon the left eye. In this evaluation we realized that most of the failure cases are localizing the left eye instead of the right eye, since the visual pattern of these two eyes looks very similar (inter class similarity), and the MOSSE technique does not exploit the left eye patches as non-target examples in the training process. Another type of wrong localization is shown by the right example, where a non-eye patch is wrongly selected as the right eye. Most likely because the intensity values of the right eye are blurred/changed by the eyeglass. This wrong localization can be avoided by using illumination invariant and more discriminative image descriptors for filter training.

2.5.8 Nonlinear Correlation Filters

All the aforementioned correlation filters are nothing but a linear approximation to map a set of training images to their corresponding outputs. It has been noted in the vision literature that this linear modeling cannot efficiently capture large pattern variations. Nonlinear learning schemas, therefore, have been introduced for challenging pattern detection/recognition.

Polynomial filters is the first technique that introduced nonlinearity in correlation filters [Mahalanobis and Kumar, 1997; Alkanhal and Vijaya Kumar, 2003]. A set of nonlinear functions is employed to transform input data points into



(a)

(b)

Figure 2.1: (a) Examples of trained ASEF (first row) and MOSSE (second row) correlation filters using 2 (leftmost), 8, 32, 128 and 512 (rightmost) training images, (b) The right eye localization rate of ASEF and MOSSE versus the size of training set.

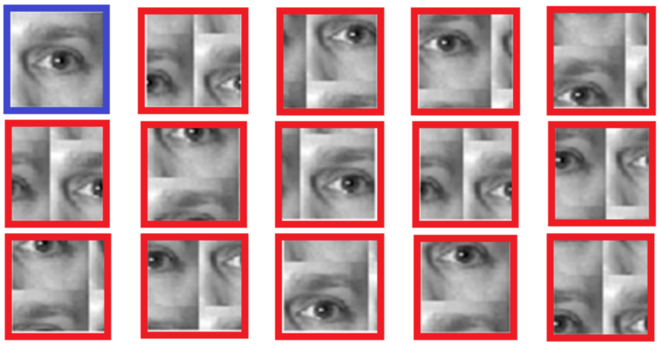


Figure 2.2: The shifted versions of the cropped right eye patch which are explicitly produced over the circular property of correlation operation. The first example (top left) with a blue border shows the training patch with zero circular shift (no shift) and is used as a target example. The other examples with a red border are shifted versions of the target patch and are used as non-target examples over the training process. The shifted patches which we refer ao as synthetic patches are not representative of real non-target patches stemming from different parts of images. Training correlation filters by these shifted patches may result in strong peaks at non-target patches, decreasing the robustness against translation.

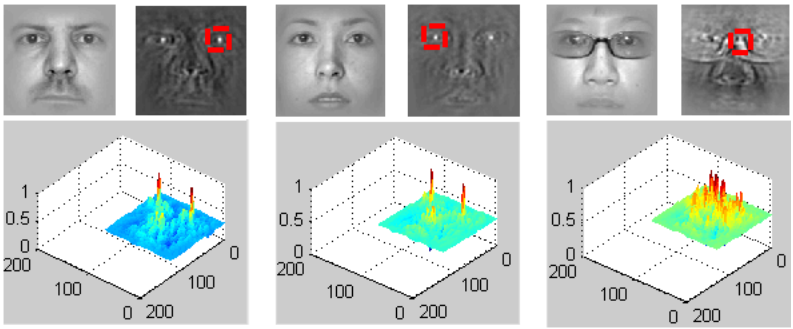


Figure 2.3: Three examples of eye localization by MOSSE correlation filter. The left example shows a successful localization. The middle example shows wrong detection of the left eye instead of the right, caused by the visual similarity of the right and left eyes (inter class similarity). The right example illustrates wrong eye localization where a non-eye patch is wrongly selected as the right eye, since the intensity values of the right eye patch are changed by the eyeglass and lighting conditions.

nonlinear feature spaces. This technique jointly trains a set of correlation filters $\{\mathbf{h}_i\}_{i=1}^N$ that returns a set of desired outputs when correlated with nonlinear versions of the training images:

$$\mathbf{y} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{h}_i \star f_i(\mathbf{x}) \tag{2.27}$$

where $\{f_i(.)\}_{i=1}^N$ indicates the N nonlinear functions.

The discrimination power of this nonlinear technique, however, comes at the cost of large features dimensionality and extra computations. Because of this, kernel based correlation filters have been proposed, where vector correlations are replaced by the inner product of two kernel functions satisfying Mercer's condition [Jeong et al., 2006].

Using kernel tricks for pattern detection is not new in vision communities and have been widely exploited to introduce nonlinear versions of many linear classification schemas. The first kernel correlation filter employed arbitrary kernels for the challenging problem of face recognition [Xie et al., 2005]. To save computational costs, this method assumed that all training images are pre-aligned and normal cross-correlation was replaced by the inner product of two kernel functions. This, however, drastically reduces the shift-invariance of correlation filters, which is an essential for robust object detection and tracking.

The correntropy filters introduced nonlinear extensions to the MACE and SDF techniques, while preserving the shift-invariance property [He et al., 2011; Jeong et al., 2006; Jeong and Principe, 2006; Jeong et al., 2009]. The kernel of this technique was selected to be the Gaussian kernel which can be easily connected to nonlinear correntropy functions. It has been shown in [Jeong et al., 2009] that directly training a correntropy-kernel filter is not computationally efficient, therefore, a fast approximation of the filter was introduced using a fast Gaussian transform.

The work done by [Henriques et al., 2012] exploited a kernel correlation filter for real-time object tracking. The main advantage of this technique is using all possible shifted versions of input images to learn a shift-invariant correlation filter. This is done by expressing correlation using Circulant matrix in the Fourier domain, and integrating this new form with Circulant kernel functions. By solving a kernel regularized least squares (KRLS) in the Fourier domain, this technique trained a kernel correlation filter in an efficient computation of $O(N^2 \log N)$.

Due to limitations of the above techniques, some recent researches have focused to combine canonical correlation filters with non-filter discriminative classifiers [Thornton et al., 2004; Rodriguez et al., 2013]. For example, the paper [Thornton et al., 2004] discussed that the poor generalization of MACE and OTSDF techniques are mainly caused by wrong peaks at the sidelobes, because, they do not control the spatial distribution of correlation energy over the correlation plane. Therefore, a SVM-like objective function was proposed to maximize the separation margin between the peak at the origin of the correlation plane and the values of sidelobes. The correlation filter was simply trained using SVM optimization over non-shifted (class 1) and circular shifted versions of object patches (class -1). The main limitation of this method was training a linear SVM which is very time consuming and requires a huge memory footprint. Similarly, Maximum Margin Correlation Filter technique simultaneously combined the good generalization of SVM classifier and the shift-invariance of MOSSE [Rodriguez et al., 2013] in a closed-form objective function with an optimal solution. This technique, however, suffers from memory and computation issues.

2.6 Non-filter Object Detection and Recognition Approaches

The challenges of object detection are mainly caused by large intra-class variations, huge objects diversity, background clutter and uncontrolled imaging conditions. To tackle these challenges, non-filter object detection techniques have been propose to distinguish object classes by learning discriminative machine learning techniques over modern image representations. In this section, we briefly review the most common object detection and recognition framework in the vision literature.

2.6.1 Appearance based object representation

Appearance based techniques employ visual cues such as color, texture, shape, and parts to distinguish foreground from the background. The earlier techniques exploited color cues for object representation and detection. First, Swin and Ballard [Swain and Ballard, 1991] introduced a simple object detection schema by means of color histograms, which was invariant to rotation, translation and scaling. It, however, showed a high sensitivity to illumination changes and occlusions. Recent approaches proposed more complicated color descriptors to tackle both illumination changes and geometric variations [Shahbaz Khan et al., 2012; Salas and Tomasi, 2011]. The discriminative power of these approaches, however, gained in the high computational complexities and memory usage.

Object detection by shape points was also investigated in earlier approaches. The Chamfer and Hausdorff distances are commonly used for shape matching, where a shape similarity is defined as the average distance between points on template shape and the nearest points on the image shape [Borgefors, 1988; Huttenlocher et al., 1993]. The simple point matching, however, was not able to handle background clutter and object deformation.

Parts based object detection, on the other hand, employ primitive objects parts (instead of points) and their spatial relationships for discriminative object representation. The early part based techniques used hand-defined part templates and spatial constraints for object detection. These techniques, however, often fail if the object parts or spatial constraints are ambiguous. More complicated techniques, contrary, have focused to automatically learn parts appearances and their spatial relationships over a large training set of part-annotated object images [Felzenszwalb et al., 2010; Azizpour and Laptev, 2012; Gall and Lempitsky, 2013]. These methods, however, requires a huge processing time and memory for learning and data annotation.

Texture cues provide rich visual cues for object detection. The Histogram of Oriented Gradients (HOG) with linear SVM classifier has been successfully applied to human detection [Dalal and Triggs, 2005]. In general, HOG captures discriminative texture by histogramming gradient orientation of local cells into corresponding orientation bins weighted by gradient magnitude. The local cells are then grouped into normalized blocks to provide strong illumination invariance. The normalized histograms of negative and positive examples are used to train a linear SVM classifier for object/background classification. Some recent extension of HOG descriptor for object detection can be found in [Déniz et al., 2011; Dalal et al., 2006; Laptev, 2009].

Gabor filter banks is also commonly used to extract textural features, due to its capabilities to capture salient visual properties such as spatial localization, orientation selectivity and spatial frequency. A bank of Gabor filters with different scales and orientations is often used to extract Gabor magnitudes. These magnitudes represent high frequencies texture and can be directly used to train a classifier (e.g. linear SVM) for object detection/recognition. The main disadvantage with these techniques is the large memory and computational complexity required for learning over modest size of training set.

2.6.2 Sliding Window

Sliding window is a common technique used to detect objects/patterns in images and videos. This technique involves scanning the image with a sliding window, from left to right and top to bottom, and classifying each sub image as either object or background classes. The main issue of sliding window techniques is the amount of time required to evaluate thousands of sub images for object/non-object classification. Some approaches employ simple classifiers or similarity measures such as histogram intersection kernels [Maji et al., 2008] and χ^2 kernels [Vedaldi and Zisserman, 2012] to speed up the classification process. They, however, suffer from low discriminative power to distinguish challenging patterns from background.

Recent approaches improved the original idea of the Viola and Jones Cascade classifier [Viola and Jones, 2001] for more accurate and faster sliding window based detection [Cevikalp and Triggs, 2012; G. Gualdi, 2012]. In general, Cascade classifiers consist of a series of binary classifiers where each categorizes sub-images either as either the foreground or background. A sub-window which is evaluated as object will be passed to the next classifiers for more evaluation. The cascade classifier will move on to the next sliding window if the current window is classified as background.

The cascade levels are constructed by Adaboost training [Freund and Schapire, 1995], where a set of weak classifiers build a strong classifier. In Adaboost algorithm, early classifiers are trained using a huge set of negative examples (which may not be always available) and a small set of positive ones such that a large number of negative sub-images can be quickly rejected with extremely low false positive rate. The late cascade levels, on the other hand, are used to classify ambiguous sub-images passed from previous levels. This training schema allows the algorithm to reject a a high percentage of negative sub-images in the early levels and save processing time for the late levels to efficiently distinguish hard background sub-images form the foreground.

2.7 Literature Summary and Comparison

We have surveyed existing correlation filters and their advantages and disadvantages for pattern detection and matching in images. In Table 2.2 and 2.3, we summarize all the current correlation filters techniques mentioned in this thesis including:

- Constrained correlation filters including SDF, MACE, MVSDF and OTF
- Unconstrained correlation filters including MACH and UMACE
- Optimized correlation output filters including ASEF and MOSSE
- Nonlinear correlation filters

and the new approaches proposed in this thesis,

- Multi-Channel Correlation Filters (MCCF)
- Correlation Filters with Limited Boundaries (CF with LB)
- Multi-Channel Correlation Filters with Limited Boundaries (MCCF with LB)

All of these techniques in Table 2.2 and 2.3 train correlation filters for pattern detection/matching. They, however, are basically different in terms of image representation, correlation output, training examples, learning strategy, computational complexity and memory usage which are clearly specified for each technique in Table 2.2 and 2.3 as follows.

Image Representation. Prior correlation filter techniques are limited to pixel intensities for filter training. This form of image representation is not discriminative enough to efficiently cover large objects variation, background clutter and different imaging conditions. Therefore, we will propose a new correlation filter objective to handle multi-channel image descriptors (e.g. HOG [Dalal and Triggs, 2005]) for robust pattern detection under uncontrolled circumstances.

Desired Correlation Outputs. All the current techniques (except unconstrained correlation filters) learn a filter that returns a set of desired correlation outputs when correlated over corresponding training images. The constrained correlation filters and most of nonlinear techniques define the correlation outputs as scaler values of 1 for positive and -1 for negative training examples. We argued in the previous section that this can result to consume huge memory usage for nonlinear techniques and produce noisy peaks for linear correlation filters such as SDF, MAC and OTF.

Recent correlation filters such as ASEF and MOSSE use Gaussian correlation outputs (two-dimensional) with the same size of the filter and training images ($D = T$) with a peak located upon the center of target images and zero values elsewhere. In this case, the accuracy of trained filters is affected by boundary effects and unbalanced synthetic negative examples. We will show in the following chapters that these limitations can be efficiently solved by training filters whose size is substantially smaller than training images ($T \ll D$), where D and T respectively indicate the size of training images and the correlation filter. The correlation output in our new approach is a Gaussian function but with a same size of the training images (much larger than the filter size).

Training Examples. Existing techniques are similar in the way that they all employ target (positive) examples cropped from a training set. But, they use different strategies to use non-target (negative) examples in the training procedure. The techniques with scaler correlation outputs use cropped non-target patches as negative training examples. These techniques, therefore, learn correlation filters which are not shift-invariant. The unconstrained approaches do not make use of non-target examples during the training process. This may lead to producing undesired peaks over non-target patches.

The ASEF and MOSSE techniques, on the other hand, employ shifted versions of cropped target patches which are implicitly produced by the circular property of the correlation operation. We will show that these shifted negative examples, which we referred to as *synthetic examples*, suffer from boundary effects and are not representative of real-word non-target patches. A new technique called correlation filters with limited boundaries will be introduced which is able to densely produce non-target patches from training images which are not affected by boundary effects and truly represent all possible non-target patches in the training images.

Incremental Learning and Online Adaption. Similar to ASEF and MOSSE, our new techniques are able to perform incremental learning in a computationally efficient manner and manageable memory usage. We will show superior results of our method (correlation filters with limited boundary) for object tracking, where the filter is quickly adapted for robust tracking against lighting, pose, appearance and scale changes over ongoing frames.

Learning by Multi Targets Per Image. ASEF and MOSSE techniques are able to handle training images with multiple target instances, by defining Gaussian-like correlation outputs with multiple peaks where each peak is located upon one target instance. This, however, trains correlation filter whose size is same as the size of training images, which is often much larger than the object of interest. Another strategy would be using local patches with a single object cropped from multi-target training images. This, however, increases sensitivity to translation. We will show that our method, correlation filters with limited boundary, is able to handle multi-target training images and train correlation filters with same size as the target of interest.

		Image Representation	**Correlation Output**	**Training Examples**	**Incremental Learning**	**Multi-targets per image**
Prior Correlation Filters	SDF, MACE, MVSDF and OTF (Sec. 2.5.1, 2.5.2, 2.5.3, 2.5.4) [Casasent and Chang, 1986] [Mahalanobis et al., 1987] [Kumar, 1986] [Refregier, 1991]	single channel (intensity)	scalar	cropped target/non-target examples	N	N
	UMACE and MACH (Sec. 2.5.5) [Mahalanobis et al., 1987] [Mahalanobis et al., 1994]	single channel (intensity)	N/A	cropped target examples	N	N
	ASEF and MOSSE (Sec. 2.5.6, 2.5.7) [Bolme et al., 2009] [Bolme et al., 2010]	single channel (intensity)	vector ($D = T$)	cropped target and shifted non-target examples	Y	Y
	Nonlinear correlation filters (Sec. 2.5.8): [Mahalanobis and Kumar, 1997] [Alkanhal and Vijaya Kumar, 2003] [Jeong et al., 2006] [Jeong et al., 2009] [Jeong and Principe, 2006] [Henriques et al., 2012] [Thornton et al., 2004] [He et al., 2011] [Rodriguez et al., 2013] [Xie et al., 2005]	single channel (intensity)	scalar	cropped target/non-target examples	N	N
Our Techniques	CF with LB (Chap. 5)	single channel (intensity)	vector ($D > T$)	target and real non-target examples	Y	Y
	MCCF (Chap. 3)	multi channel	vector ($D = T$)	cropped target and shifted non-target examples	Y	Y
	MCCF with LB (Chap. 5)	multi channel	vector ($D > T$)	target and real non-target examples	Y	Y

Table 2.2: Summarizing existing correlation filter techniques and the new approaches proposed in this thesis.

		Time Complexity	**Memory**
Prior Correlation Filters	SDF (Sec. 2.5.1) [Casasent and Chang, 1986]	$\mathcal{O}(N^3)$	$\mathcal{O}(ND)$
	MACE, MVSDF and OTF (Sec. 2.5.2, 2.5.3, 2.5.4) [Mahalanobis et al., 1987] [Kumar, 1986] [Refregier, 1991]	$\mathcal{O}(\max\{N^3, D^3\})$	$\mathcal{O}(ND)$
	UMACE and MACH (Sec. 2.5.5) [Mahalanobis et al., 1987] [Mahalanobis et al., 1994]	$\mathcal{O}(D^3)$	$\mathcal{O}(D)$
	ASEF and MOSSE (Sec. 2.5.6, 2.5.7) [Bolme et al., 2009] [Bolme et al., 2010]	$\mathcal{O}(ND\log D)$	$\mathcal{O}(D)$
	Nonlinear correlation filters (Sec. 2.5.8): [Mahalanobis and Kumar, 1997] [Alkanhal and Vijaya Kumar, 2003] [Jeong et al., 2006] [Jeong et al., 2009] [Jeong and Principe, 2006] [Henriques et al., 2012] [Thornton et al., 2004] [He et al., 2011] [Rodriguez et al., 2013] [Xie et al., 2005]	N/A	$\mathcal{O}(ND)$
Our Techniques	CF with LB (Chap. 5)	$\mathcal{O}([N+K]T\log T)$	$\mathcal{O}(T)$
	MCCF (Chap. 3)	$\mathcal{O}(DC^3 + NDC^2)$	$\mathcal{O}(C^2D)$
	MCCF with LB (Chap. 5)	$\mathcal{O}(KT(C^3 + NC^2))$	$\mathcal{O}(C^2T)$

Table 2.3: The computational complexity and memory usage of existing correlation filters techniques and the new approaches proposed in this thesis. N and D respectively denote the number of training examples and the length (size) of each example (trained filter). K refers to the number of iterations in ADMM solver. T refers to the size of search window, where $T > D$ in the correlation filters with limited boundaries. The number of channels is denoted by C.

Chapter 3

Multi-Channel Correlation Filters

In computer vision it is now rare for tasks like detection/matching to be performed on single channel image descriptors (e.g. 2D array of intensity values or gradient magnitudes). With the advent of advanced descriptors like HOG [Dalal and Triggs, 2005] and SIFT [Lowe, 1999] pattern matching across multi-channel signals has become the norm rather than the exception in most visual detection tasks. Most of these image descriptors can be viewed as multi-channel images/signals with multiple measurements (such the oriented edge energies) associated with each pixel location. We shall herein refer to all image descriptors as multi-channel images.

The motivation for working with multi-channel image descriptors rather than raw single channel pixel intensities stems from seminal work on the mammalian primary visual cortex (V1) [Hubel and Wiesel, 1962]. Here, local object appearance and shape can be well categorized by the distribution of local directional edges, without precise knowledge of their spatial location. Jarrett et al. [2009] showed that many V1-inspired features follow a similar pipeline of filtering an image through a large filter bank, followed by a nonlinear rectification step, and finally a blurring/histogramming step resulting in a multi-channel signal (where the number of channels was dictated by the size of the filter bank). It has been noted [Jarrett et al., 2009] that V1-inspired descriptors obtain superior photometric and geometric invariance in comparison to raw intensities giving strong motivation for their use in many modern vision applications.

The most notable approach to multi-channel detection in computer vision can be found in the seminal work of Dalal & Triggs [Dalal and Triggs, 2005] where the authors employ a HOG descriptor in conjunction with a linear SVM to learn a detector for pedestrian detection. This same multi-channel detection pipeline has gone on to be employed in a myriad of other detection tasks in vision ranging from facial landmark localization/detection [Zhu and Ramanan, 2012] to general object detection [Felzenszwalb et al., 2010].

Computational and memory efficiency, however, are issues for Dalal & Triggs style multi-channel detectors, and originate from solving a quadratic objective in the spatial domain, and the pre-loading of multi-channel descriptors of all training images. A central advantage of using a linear SVM for learning a multi-channel detector, however, is the ability to treat that detector as a multi-channel linear filter during evaluation. Instead of inefficiently moving the detector spatially across a multi-channel image, one can take advantage of the fast Fourier transform (FFT) for the efficient application of correlating a desired template/-filter with a signal.

During training, however, all learning is done in the spatial domain. This can be a slow and inefficient process. The strategy involves the extraction of positive and negative multi-channel image patches of the object/pattern of interest across large amounts of data. From a practical perspective, most algorithms employed for learning multi-channel object detectors incur a memory cost linear in the number of samples (e.g. linear SVM in Dalal and Triggs [2005]). Whilst this seems reasonable at a glance, consider a simple example of storing $200,000$ 50×50 single-channel image in double precision. In the case of raw pixels this amounts to only 3.72 GB of storage, a manageable figure on current desktop hardware. Using a multi-channel image of 40 channels (e.g. 5 scales and 8 orientations when using oriented edge energies), storage blows out to an untenable 149 GB. Strategies have been proposed to curb storage complexity, however they are largely based on heuristic subsampling of the resolution of the multi-channel image, or the number of training samples.

From a learning perspective, much of this storage can be viewed as inefficient as it often involves shifted versions of the same multi-channel image. This is a real strength of MOSSE correlation filters as the objective provides a way for

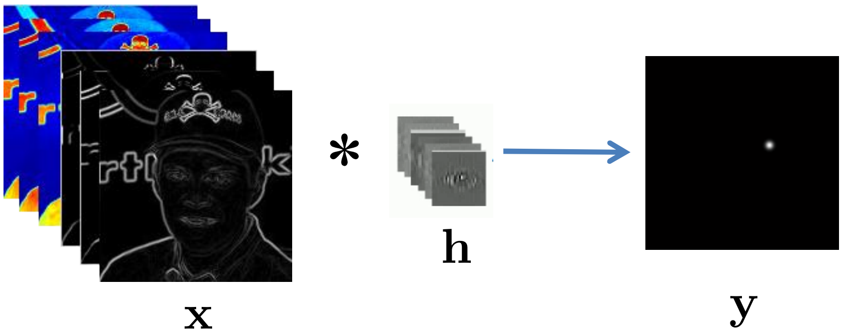


Figure 3.1: An example of multi-channel correlation/convolution where one has a multi-channel image **x** correlated/convolved with a multi-channel filter **h** to give a single-channel response **y**. By posing this objective in the frequency domain, our multi-channel correlation filter approach attempts to give a computational & memory efficient strategy for estimating **h** given **x** and **y**.

naturally modeling shifted versions of an image without the burden of explicitly storing all the shifted image patches. In addition, inspecting the objective of the MOSSE technique in the Fourier domain one can see that its learning processes only involves an FFT of training samples and an simple summation of the samples in the Fourier domain, referred as to auto- cross- correlation energies, with a memory footprint of O(D), where D is the length of the signal/image. This great memory efficiency of the MOSSE technique gives one the chance to use a huge amount of positive/negative samples during the learning process with no memory concern. A nice property which can give the technique the ability to learning numerous different patterns.

Hitherto, correlation filter theory with efficient memory and computation, to our knowledge, has been restricted to single-channel signals/filters limited to image intensities. This does not allow the canonical correlation filters to handle multi-channel descriptors for accurate detection/matching of complex patterns as many non-filter detection techniques do.

An example of multi-channel correlation can be seen in Figure 3.1 where a multi-channel image is convolved/correlated with a multi-channel filter/detector in order to obtain a *single-channel* response. The peak of the response (in white) indicating where the pattern of interest is located. Training efficient multi channel correlation filters for pattern detection/matching is at the heart of this chapter.

3.1 Correlation Filters in the Spatial Domain

Bolme et al. [2010]'s MOSSE correlation filter can be expressed in the spatial domain as,

$$E(\mathbf{h}) = \frac{1}{2}\sum_{i=1}^{N} ||\mathbf{y}_i - \mathbf{h} \star \mathbf{x}_i||_2^2 + \frac{\lambda}{2}||\mathbf{h}||_2^2 \tag{3.1}$$

where $\mathbf{y}_i \in \mathbb{R}^D$ is the desired response for the i-th observation $\mathbf{x}_i \in \mathbb{R}^D$, λ is a regularization term and $\star$ indicates the spatial correlation of two signals. In order to mathematically represent the spatial correlation operator one can express Equation 3.1 as solving a ridge regression problem in the spatial domain,

$$E(\mathbf{h}) = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{D} ||\mathbf{y}_i(j) - \mathbf{h}^\top \mathbf{x}_i[\Delta\boldsymbol{\tau}_j]||_2^2 + \frac{\lambda}{2}||\mathbf{h}||_2^2 \tag{3.2}$$

where $\mathbb{C} = [\Delta\boldsymbol{\tau}_1, \ldots, \Delta\boldsymbol{\tau}_D]$ represents the set of all circular shifts for a signal of length D. Bolme et al. [2010] advocated the use of a 2D Gaussian of small variance (2-3 pixels) for $\mathbf{y}_i$ centered at the location of the object (typically the centre of the image patch). The solution to this objective becomes,

$$\mathbf{h} = \mathbf{H}^{-1}\sum_{i=1}^{N}\sum_{j=1}^{D} \mathbf{y}_i(j)\mathbf{x}_i[\Delta\boldsymbol{\tau}_j] \tag{3.3}$$

where,

$$\mathbf{H} = \lambda\mathbf{I} + \sum_{i=1}^{N}\sum_{j=1}^{D} \mathbf{x}_i[\Delta\boldsymbol{\tau}_j]\mathbf{x}_i[\Delta\boldsymbol{\tau}_j]^\top \ . \tag{3.4}$$

Solving a correlation filter in the spatial domain quickly becomes intractable as a function of the signal length D, as the cost of solving Equation (3.3) becomes $\mathcal{O}(D^3 + ND^2)$.

3.2 Correlation Filters in the Fourier Domain

It is well understood in signal processing that circular convolution in the spatial domain can be expressed as a Hadamard product in the frequency domain [Kumar, 2005]. This allows one to express the objective in Equation 3.2 more succinctly and equivalently as,

$$\begin{aligned} E(\hat{\mathbf{h}}) &= \frac{1}{2}\sum_{i=1}^{N} ||\hat{\mathbf{y}}_i - \hat{\mathbf{x}}_i \circ \mathrm{conj}(\hat{\mathbf{h}})||_2^2 + \frac{\lambda}{2}||\hat{\mathbf{h}}||_2^2 \qquad (3.5) \\ &= \frac{1}{2}\sum_{i=1}^{N} ||\hat{\mathbf{y}}_i - \mathrm{diag}(\hat{\mathbf{x}}_i)^\top \hat{\mathbf{h}}||_2^2 + \frac{\lambda}{2}||\hat{\mathbf{h}}||_2^2 \ . \end{aligned}$$

where $\hat{\mathbf{h}}, \hat{\mathbf{x}}, \hat{\mathbf{y}}$ are the Fourier transforms of $\mathbf{h}, \mathbf{x}, \mathbf{y}$. The complex conjugate of $\hat{\mathbf{h}}$ is employed to ensure the operation is correlation not convolution. The equivalence between Equations 3.2 and 3.5 also borrows heavily upon another well known property from signal processing namely, Parseval's theorem which states that

$$\mathbf{x}_i^\top \mathbf{x}_j = D^{-1}\hat{\mathbf{x}}_i^\top \hat{\mathbf{x}}_j \quad \forall i, j, \quad \text{where } \mathbf{x} \in \mathbb{R}^D \ . \qquad (3.6)$$

The solution to Equation 3.5 becomes

$$\begin{aligned} \hat{\mathbf{h}} &= [\mathrm{diag}(\hat{\mathbf{s}}_{xx}) + \lambda \mathbf{I}]^{-1} \sum_{i=1}^{N} \mathrm{diag}(\hat{\mathbf{x}}_i)\hat{\mathbf{y}}_i \qquad (3.7) \\ &= \hat{\mathbf{s}}_{xy} \circ^{-1} (\hat{\mathbf{s}}_{xx} + \lambda \mathbf{1}) \end{aligned}$$

where $\circ^{-1}$ denotes element-wise division, and

$$\hat{\mathbf{s}}_{xx} = \sum_{i=1}^{N} \hat{\mathbf{x}}_i \circ \mathrm{conj}(\hat{\mathbf{x}}_i) \quad \& \quad \hat{\mathbf{s}}_{xy} = \sum_{i=1}^{N} \hat{\mathbf{y}}_i \circ \mathrm{conj}(\hat{\mathbf{x}}_i) \qquad (3.8)$$

are the average auto-spectral and cross-spectral energies respectively of the training observations. The solution for $\hat{\mathbf{h}}$ in Equations 3.2 and 3.5 are identical other than that one is posed in the spatial domain, and the other is in the frequency domain. The power of this method lies in its computational efficiency. In the frequency domain a solution to $\hat{\mathbf{h}}$ can be found with a cost of $\mathcal{O}(ND \log D)$. The primary cost is associated with the DFT on the ensemble of training signals $\{\mathbf{x}_i\}_{i=1}^{N}$ and desired responses $\{\mathbf{y}_i\}_{i=1}^{N}$.

Inspecting Equation 3.8 one can see an additional advantage of correlation filters when posed in the frequency domain, specifically, memory efficiency. One does not need to store the training examples in memory before learning. As Equation 3.8 suggests one needs to simply store a summation of the auto-spectral $\hat{\mathbf{s}}_{xx}$ and cross-spectral $\hat{\mathbf{s}}_{xy}$ energies. This is a powerful result not often discussed in correlation filter literature as unlike other spatial strategies for learning detectors (e.g. linear SVM) whose memory usage grows as a function of the number of training examples $\mathcal{O}(ND)$, correlation filters have fixed memory overheads $\mathcal{O}(D)$ irrespective of the number of training examples.

3.3 Proposed Multi-Channel Framework

Inspired by single-channel correlation filters we proposed a multi-channel strategy for learning a correlation filter [Kiani et al., 2013]. Our proposed multi-channel objective in the spatial domain is,

$$E(\mathbf{h}) = \frac{1}{2} \sum_{i=1}^{N} ||\mathbf{y}_i - \sum_{k=1}^{K} \mathbf{h}^{(k)} \star \mathbf{x}_i^{(k)}||_2^2 + \frac{\lambda}{2} \sum_{k=1}^{K} ||\mathbf{h}^{(k)}||_2^2 \tag{3.9}$$

where $\star$ indicates the spatial correlation of two signals in Figure 3.2(a). As shown for the single channel correlation filters, the above equation can be expressed as solving a multi-channel ridge regression problem in the spatial domain as:

$$E(\mathbf{h}) = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{D} ||\mathbf{y}_i(j) - \sum_{k=1}^{K} \mathbf{h}^{(k)\top} \mathbf{x}_i^{(k)}[\Delta \boldsymbol{\tau}_j]||_2^2 + \frac{\lambda}{2} \sum_{k=1}^{K} ||\mathbf{h}^{(k)}||_2^2 \tag{3.10}$$

where $\mathbf{x}^{(k)}$ and $\mathbf{h}^{(k)}$ refers to the kth channel of the vectorized image and filter respectively where K represents the number of filters. As with a canonical filter the desired response is single-channel $\mathbf{y} = [\mathbf{y}(1), \ldots, \mathbf{y}(D)]^T$ even though both the filter and the signal are multi-channel. Solving this multi-channel form in the spatial domain is even more intractable than the single-channel form with a cost of $\mathcal{O}(D^3K^3 + ND^2K^2)$ since we now have to solve a $KD \times KD$ linear system.

Inspired by the efficiencies of posing single-channel correlation filters in the Fourier domain we can express Equation 3.10 equivalently and more succinctly as,

$$E(\hat{\mathbf{h}}) = \frac{1}{2}\sum_{i=1}^{N} ||\hat{\mathbf{y}}_i - \sum_{k=1}^{K} \operatorname{diag}(\hat{\mathbf{x}}_i^{(k)})^\top \hat{\mathbf{h}}^{(k)}||_2^2 + \frac{\lambda}{2}\sum_{k=1}^{K} ||\hat{\mathbf{h}}^{(k)}||_2^2 \tag{3.11}$$

where $\hat{\mathbf{h}} = [\hat{\mathbf{h}}^{(1)\top}, \ldots, \hat{\mathbf{h}}^{(K)\top}]^\top$ is a KD dimensional supervector of the Fourier transforms of each channel. This can be simplified further,

$$E(\hat{\mathbf{h}}) \;=\; \frac{1}{2}\sum_{i=1}^{N} ||\hat{\mathbf{y}}_i - \hat{\mathbf{X}}_i\hat{\mathbf{h}}||_2^2 + \frac{\lambda}{2}||\hat{\mathbf{h}}||_2^2 \;. \tag{3.12}$$

where $\hat{\mathbf{X}}_i = [\operatorname{diag}(\hat{\mathbf{x}}_i^{(1)})^\top, \ldots, \operatorname{diag}(\hat{\mathbf{x}}_i^{(K)})^\top]$, Figure 3.2(b). At first glance the cost of solving this linear system looks no different to the spatial domain as one still has to solve a $KD \times KD$ linear system with an intractable cost of $\mathcal{O}(D^3K^3 + ND^2K^2)$:

$$\hat{\mathbf{h}}^* \;=\; (\lambda\mathbf{I} + \sum_{i=1}^{N} \hat{\mathbf{X}}_i^\top \hat{\mathbf{X}}_i)^{-1} \sum_{i=1}^{N} \hat{\mathbf{X}}_i^\top \hat{\mathbf{y}}_i \tag{3.13}$$

3.3.1 A Variable Re-ordering Approach for Efficient Optimization

According to Figure 3.2 (b), $\hat{\mathbf{X}}$ is sparse banded and inspecting Equation 3.12 one can see that the jth element of each correlation response $\hat{\mathbf{y}}_i(j)$ is dependent only on the K values of $\mathcal{V}(\hat{\mathbf{h}}(j))$ and $\mathcal{V}(\hat{\mathbf{x}}(j))$, where $\mathcal{V}$ is a concatenation operator that returns a $K \times 1$ vector when applied on the jth element of a K-channel vector $\{\mathbf{a}^{(k)}\}_{k=1}^{K}$, i.e. $\mathcal{V}(\mathbf{a}(j)) = [\text{conj}(\mathbf{a}^{(1)}(j)), ..., \text{conj}(\mathbf{a}^{(K)}(j))]^\top$. Therefore, we proposed to equivalently express Equation 3.12 through a simple variable re-ordering as, Figure 3.2(c):

$$\begin{aligned} E(\mathcal{V}(\hat{\mathbf{h}}(j))) \quad &= \quad \frac{1}{2}\sum_{i=1}^{N} ||\hat{\mathbf{y}}_i(j) - \mathcal{V}(\hat{\mathbf{x}}_i(j))^\top \mathcal{V}(\hat{\mathbf{h}}(j))||_2^2 + \frac{\lambda}{2}||\mathcal{V}(\hat{\mathbf{h}}(j))||_2^2, \\ & \quad\quad for \qquad j = 1, ..., D. \end{aligned} \tag{3.14}$$

Therefore, an efficient solution of Equation 3.12 can be found by solving D independent $K \times K$ linear systems using Equation 3.14 as:

$$\mathcal{V}(\hat{\mathbf{h}}(j))^* = \hat{\mathbf{H}}^{-1} \sum_{i=1}^{N} \mathcal{V}(\hat{\mathbf{x}}_i(j))\hat{\mathbf{y}}_i(j) \tag{3.15}$$

where,

$$\hat{\mathbf{H}} = \lambda \mathbf{I} + \sum_{i=1}^{N} \mathcal{V}(\hat{\mathbf{x}}_i(j))\mathcal{V}(\hat{\mathbf{x}}_i(j))^\top \tag{3.16}$$

This results in a substantially smaller computational cost of $\mathcal{O}(DK^3 + NDK^2)$, which is required to solve a set of D independent $K \times K$ linear systems, compared to optimizing this objective using Equations 3.12 and 3.9 by solving a $KD \times KD$ linear system of cost of $\mathcal{O}(D^3K^3 + ND^2K^2)$, where in most cases $D \gg K$ and

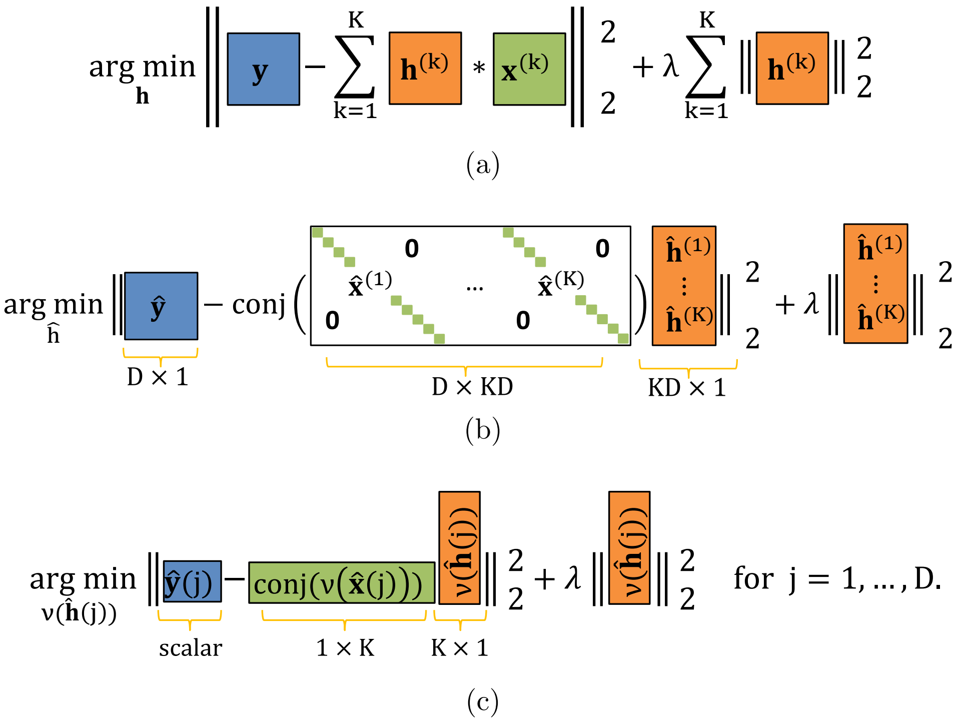


Figure 3.2: Visualizing (a) MCCF objective in the spatial domain, (b) MCCF objective in the Fourier domain, and (c) Variable re-ordering for computationally efficient optimization. For direct optimization of (a) and (b) in the spatial and frequency domain one needs to solve a $KD \times KD$ linear system with a computational cost of $\mathcal{O}(D^3K^3 + ND^2K^2)$ and memory usage of $\mathcal{O}(D^2K^2)$. Using the proposed variable re-ordering approach, the objective in (b) can be expressed by D independent $K \times K$ linear system in a substantially smaller computational cost of $\mathcal{O}(DK^3 + NDK^2)$ and required memory of $\mathcal{O}(K^2D)$ independent of the number of training samples.

D is very large (the dimension of a training example) [Kiani et al., 2013].

3.3.2 Memory Efficiency

As outlined in Section 3.2, an additional strength of single-channel correlation filters is their memory efficiency. Specifically, one does not need to hold all the training examples in memory, as SVM does. Instead, they need to just compute the auto-spectral $\hat{\mathbf{s}}_{xx}$ and cross-spectral $\hat{\mathbf{s}}_{xy}$ energies respectively of the training observations (see Equation 3.8). The memory saving become sizable as the number of training examples increases, the memory overhead remains constant at $\mathcal{O}(D)$ instead of $\mathcal{O}(ND)$ if one was to employ a spatial objective.

A similar strategy can be exploited in our multi-channel correlation form. For multi-channel correlation filters this saving becomes even more dramatic as the memory overhead remains constant $\mathcal{O}(K^2D)$ as opposed to $\mathcal{O}(NDK)$ in the non-filter learning techniques (e.g. linear SVM). This property stems from the sparse banded structure of multi-channel correlation filters such that the problem can be posed as D independent $K \times K$ linear systems [Kiani et al., 2013].

3.4 Experiments

To demonstrate the efficiency of the proposed technique, we evaluate it across a number of challenging localization and detection tasks using several publicly available datasets including the UIUC Cars dataset [Agarwal and Roth, 2002], INRIA Horses dataset [Ferrari et al., 2010] and Daimler pedestrian dataset [Munder and Gavrila, 2006]. All these datasets consist of images with large intra-class variations, cluttered background, partial occlusion, scale and huge out-door lighting changes.

Object Detection/Localization. The goal of object detection/localization is to predict the *location* and the *bounding box* of the target object within an image along with its confidence score. Given a test image and a trained multi-channel correlation filter of an specific object, the object detection/localization is performed by correlating each channel correlation filter over its corresponding feature channel and then summing up all the correlation outputs to get a single confidence map (correlation output $\mathbf{y}$, Figure 3.1). A local maximum whose confidence score is more than a threshold indicates the presence of the target object within test images. The bounding box of the detected object is approximated by the 2D size of the correlation filter/detector whose maximum response indicates the presence of the target object.

Feature Extraction and Processing. Across all the experiments, we used the same multi-channel image representation, specifically HOG [Dalal and Triggs, 2005] characterized by nine orientation bins over unsigned $[0, \pi]$ degrees normalized by cell and block sizes of 5×5. All the images are power normalized to have zero-mean and unit variance to increase the robustness of the detector against large lighting changes. A Cosine-window is applied on all the feature channels (HOGs) in the spatial domain to reduce the frequency effects caused by high frequencies belonging to the opposite borders of the images in the Fourier domain.

Desired Correlation Outputs. For all experiments, we defined the desired correlation response for the positive training samples using a 2D Gaussian function with a spatial variance of 2-3 pixels whose peak is centered at the location of the target of interest (car, horse, pedestrian, etc.) and near to zeros values elsewhere. The desired correlation filters for negative/background training samples are defined using a 2D matrix of zero values.

Evaluation Metrics. The detection performance is measured by the number of true positive (TP - the number of desired foreground objects detected by the detector) and false positive (FP - the number of background patches wrongly detected as the object of interest) detections. A detection is considered as false positive if the predicted bounding box with a correlation response higher than the detection threshold does not contain the target object. A detection is true positive if the overlap ratio a_0 between the predicted bounding box B_p and a ground truth bounding box B_{gt} is more than a decision threshold θ,

$$a_0 = \frac{area(B_p \bigcap B_{gt})}{area(B_p \bigcup B_{gt})} \geqslant \theta \tag{3.17}$$

The trade-off between TP and FP can be effectively measured using detection rate versus false positive rate, Precision-Recall and Recall-FPPI (false positive per image) by varying a threshold and computing the recall and precision/FPPI for each threshold value, where

$$Recall = \frac{TP}{nP} \tag{3.18}$$

$$Precision = \frac{TP}{TP + FP} \tag{3.19}$$

$$FPPI = \frac{FP}{N} \tag{3.20}$$

where TP, nP, FP and N respectively indicate the true positive detections, the total number of desired objects, the number of false detections and the number of test images.

3.4.1 Daimler Pedestrian Dataset: Comparison with Linear SVM

We evaluated our method for pedestrian detection using the Daimler pedestrian dataset [Munder and Gavrila, 2006] containing five disjoint images sets, three for training and two for testing. Each set consists of 4800 pedestrian and 5000 non-pedestrian (background) images of size 36 × 18. In total, there are (4500 + 5000) × 5 images in this dataset. The HOGs were computed over 5 orientation bins with cell and block sizes of 3 × 3. A multi-channel correlation filter (MCCF) for pedestrian detection was trained using all the negative and positive training samples with their corresponding desired responses. Given a testing image, we first correlate its feature channels with the trained MCCF and then measure the Peak-to-Sidelobe Ratio (PSR) of the correlation output plane with a threshold to detect pedestrian. This threshold was chosen through a cross-validation process. Peak-to-Sidelobe Ratio (PSR) is a common metric used in correlation filter literature for detection/verification tasks. It is the ratio of the peak response to the local surrounding response, more details on this measure can be found in [Kumar, 2005]. In this experiment we chose to compare our MCCF directly with a spatial detector learned using a linear SVM (as originally performed by Dalal and Triggs [Dalal and Triggs, 2005]). The linear SVM was trained in almost exactly the same fashion as our MCCF so as to keep the comparison as fair as possible.

Detection Performance. We evaluated the detection performance of our method and SVM using detection rate versus false positive rate (FPR). The detection rate refers to the percentage of the pedestrian images which are correctly classified as pedestrians. False positive rate, on the other hand, indicates how often non-pedestrian images are wrongly detected as pedestrian.

Inspecting Figure 3.3 (a) one can see the detection performance of our method at FPR = 0.10 as a function of the size of the training set. It is interesting to note that our MCCF objective can achieve good detection performance with substantially smaller amounts of training data when compared to a linear SVM. The ROC curves of 250 and 1000 training images are shown in Figure 3.3 (b-c) for more details, where our method outperformed SVM for all false positive rates. This superior performance for less training images can be attributed to how correlation filters implicitly use synthetic circular shifted versions of images within the learning process without having to explicitly create the images. As a result our MCCF objective can do "more with less" by achieving good detection performance with substantially less training data.

Inspecting Figure 3.3 (d) one can see our MCCF obtains almost similar detection results to the linear SVM in terms of detection performance as a function of false positive rate (FPR). This result is not that surprising as the linear SVM objective is quite similar to the MCCF objective (which can be interpreted as a ridge regression when posed in the spatial domain). It is well understood that the linear SVM objective enjoys better tolerance to outliers than ridge regression, but in practice we have found that advantage to be only marginal when learning multi-channel detectors.

Some examples of true positive, false positive and false negative cases detected by our method are shown in Figure 3.5. The pedestrian patches (second row) are classified as non-pedestrian due to their low quality and low contrast that make gradient computation for HOGs inefficient. The main reason for classifying non-pedestrian patches as pedestrian (third row) is their vertical pattern (e.g. trees, bars, fences) which is visually very similar to the human body.

Computation and Memory Efficiency. Figure 3.4 depicts one of the major advantages of MCCF, and that is its superior scalability with respect to training set size. One can see how training time starts to increase dramatically for linear

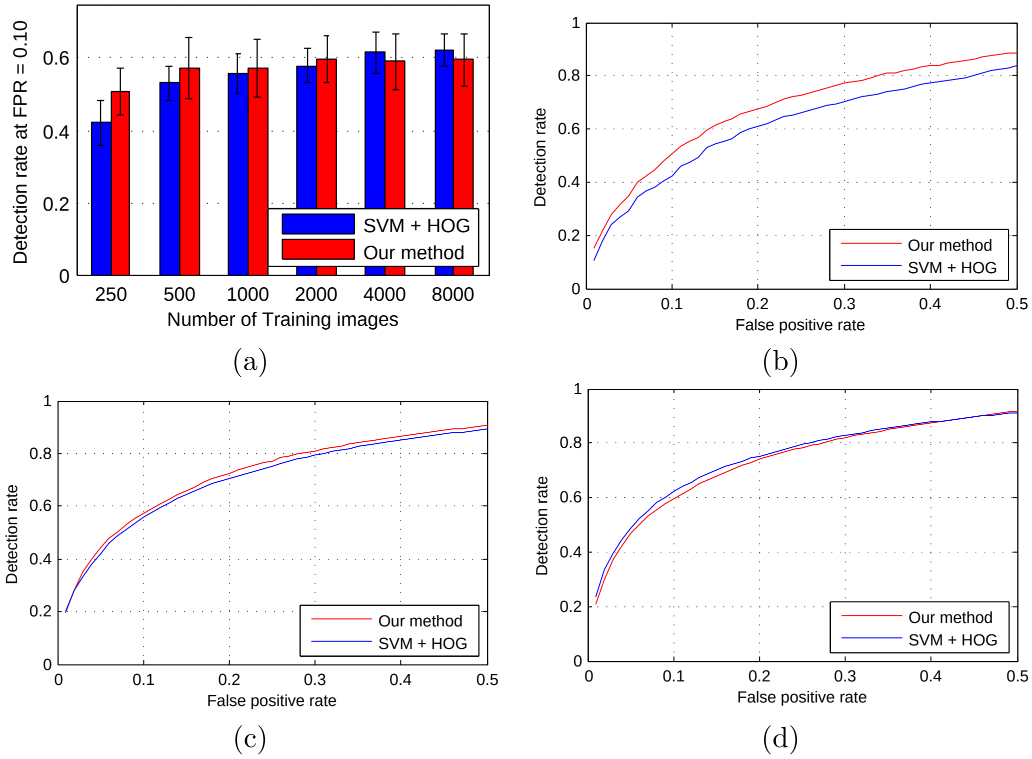


Figure 3.3: **Daimler Pedestrian Dataset.** Comparing our method with SVM + HOG (a) pedestrian detection rate at FPR = 0.10 versus the number of training images, and (b-d) ROC curves of detection rate as a function of false positive rate for (b) 250, (c) 1000 and (d) 8000 training images.

SVM[1] where as our training time only increases modestly as a function of training set size. The central advantage of our proposed approach here is that the solving of the multi-channel linear system in the frequency domain is independent of the number of images. Therefore the only component of MCCF that is dependent on training set size is the actual FFT on the training images which should only have the moderate computational cost $\mathcal{O}(ND\log D)$ as the training set size N increases. Finally, inspecting Table 3.1 one can see the superior nature of our MCCF approach in comparison to linear SVM with respect to memory usage. As discussed in Section 3.3 our proposed MCCF approach has a modest fixed memory requirement independent of the training set size, whereas the amount of memory used by the linear SVM approach is a linear function of the number of training examples employed.

[1] We employed the efficient and widely used LibLinear linear SVM package `http://www.csie.ntu.edu.tw/~cjlin/liblinear` in all our experiments.

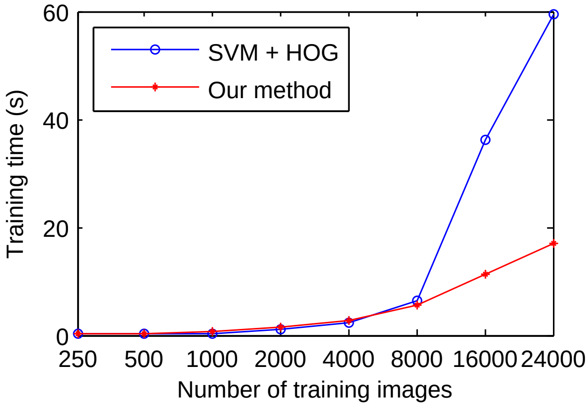


Figure 3.4: Comparing the training time of our method with SVM as a function of training set sizes. MCCF's training time linearly and slowly grows up by increasing the number of training images, and it is the time required to compute the FFT of training images. Computing the auto- and cross- correlated energies in the Frequency domain and filter training are independent of the number of training images. For SVM, however, the training time nonlinearly increases when the size of training samples increases. The training images are randomly selected from three training sets containing 29400 pedestrian and non-pedestrian images.

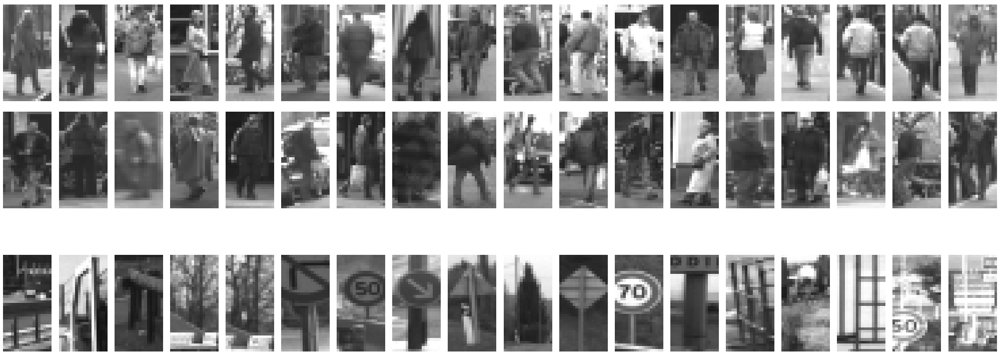


Figure 3.5: Some samples of (top) true detection of pedestrian (true positive), (middle) false detection of non-pedestrian (false negative), and (bottom) false detection of pedestrian (false positive).

	250	500	1000	2000	4000	8000	16000	24000
MCCF	0.02	0.02	0.02	0.02	0.02	0.02	0.02	0.02
SVM	6.17	12.35	24.68	49.36	98.87	197.44	395.88	592.32

Table 3.1: Comparing minimum required memory (MB) of our method with SVM as a function of number of training images. The memory footprint of our method is independent of the number of training images, because all memory it requires is for auto- and cross- correlation energies with the same size of a training image. The required memory for SVM is linear to the size of training size, since this technique requires to load all training samples in the memory for classifier learning.

3.4.2 UIUC Cars Dataset

The UIUC cars dataset [Agarwal and Roth, 2002] contains gray level images of side views of cars with very large intra-class variations, partial occlusion, different resolution, cluttered background, extreme lighting and scale changes. The training samples contain 550 cars and 550 non-car images of size 100 × 40 pixels. The dataset consist of two standard test sets. The first contains 200 single-scale cars in 170 street scene images. The cars in this set are all the same size of 100×40, as in the training images. The other test set contains 139 cars in 108 images in sizes between 89×36 and 212×85 pixels. We used the Equal Error Rate (EER) of the Precision-Recall curve to evaluate the detection performance, which is the value where the precision and recall are the same. The overlap ratio for true/false positive detections is set to $a_0 \geqslant 0.50$ ($\theta = 0.50$ in Equation 3.17).

Multi-Scale Detection and Bounding Box Approximation. For multi-scale detection, we use the pyramid approach proposed in [Felzenszwalb et al., 2010] as illustrated in Figure 3.6. First, an image pyramid is formed by scaling the test image from 0.4 to 1.4 times the image size with a scale-step of 1.2. Then, the correlation output for each level of the pyramid is obtained by correlating the multi-channel correlation filter over its corresponding scaled image. The final correlation output (confidence map) is the maximum value over all the pyramid correlation outputs $y = \max\{y_0, ..., y_{N-1}\}$. The bounding box of a detected object is approximated by dividing the 2D size of the filter over the scale factor of the pyramid level whose the maximum value belongs to.

Evaluation. We compare our method against other state-of-the-art approaches on the UIUC cars dataset as reported in Table 3.2. The evaluation is performed using the detection rate at the Equal Error Rates (EER) of the Precision-Recall curve. The results show the overall superiority of our method compared to the others both in terms of detection rate at EER and detection time. The best detection rate on the single scale set belongs to Mutch et al. [Mutch and Lowe, 2006] followed by our proposed method. This method, however, suffers from very low detection speed (0.03 fps), compared to our method with a detection

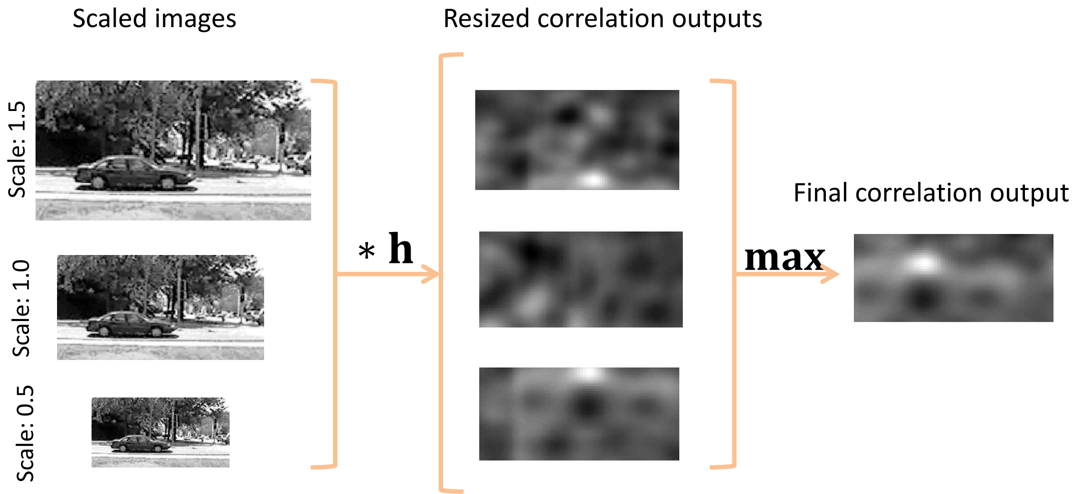


Figure 3.6: A pyramid approach for multi-scale object detection. The scale pyramid is formed by scaling up/down the input image by different factors, the first column (1.5, 1.0 and 0.5, here). The second column shows the correlation outputs corresponded to the pyramid levels (scales). The final correlation output (confidence map) is computed by max-pooling of the resized correlation outputs over all pyramid levels, the third column. The final correlation output is used to predict the location of the target objects, where the confidence score is more than a pre-defined threshold. The bounding box of the predicted target is approximated using the size of the correlation filter divided by the scale factor of the pyramid level whose the confidence score belongs to.

speed of 70 fps. For multi scale detection, our method obtained the third best detection rate lower than Lampert et al. [Lampert et al., 2008] and Gall et al. [Gall and Lempitsky, 2009]. These methods, however, require a huge amount of memory and computations to train an efficient sub-window searching technique. Figure 3.8 visualizes the detection rate (%) and speed (fps) of our method versus the state of the arts. The average detection speed of our method with slightly higher accuracy is almost 9 times higher than the fastest previous approach with same detection rate on the single scale images. Note that we can easily speed up our system by parallelizing the correlations across different feature channels and image scales (up to 400 fps) . Figure 3.7 shows some qualitative car detection results.

3.4.3 INRIA Horses Dataset

The INRIA horse dataset [Ferrari et al., 2010] consists of 170 images with one or more side-views of horses as positive samples and 170 background images

Method	Single Scale Images		Multi Scale Images	
	accuracy	time (sec.)	accuracy	time (sec.)
SVM + HOG	95.0%	0.02	77.9%	0.2
Mutch and Lowe [2006]	99.9%	30	90.6%	-
Leibe et al. [2008]	97.5%	3	95.0%	3
Kuo and Nevatia [2009]	98.5%	0.5	95.0%	2.8
Lampert et al. [2008]	98.5%	0.125	98.6%	0.40
Gall and Lempitsky [2009]	98.5%	1.5	98.6%	6
Our method	**99.5%**	**0.015**	**97.84%**	**0.12**

Table 3.2: **UIUC car dataset.** Detection rate at EER and detection time (sec) of our method compared to state-of-the-art approaches. Our method achieves very competitive results with much faster detection speed.

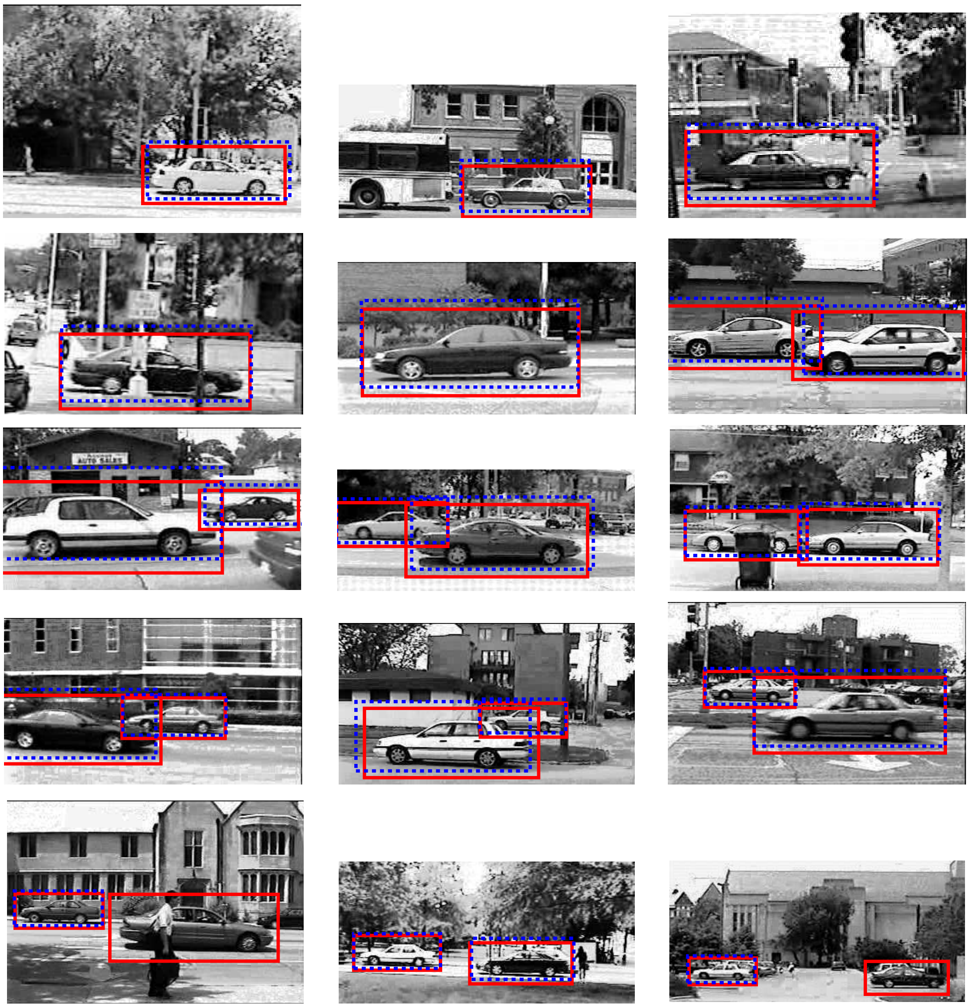


Figure 3.7: Detection results of our method over the UIUC cars dataset. The proposed method is able to find the cars in street images captured under uncontrolled circumstances with challenging intra-class variations, very textured background, extreme lighting and scale changes. The ground truth and the detected cars are respectively shown by the red and dashed blue boxes.

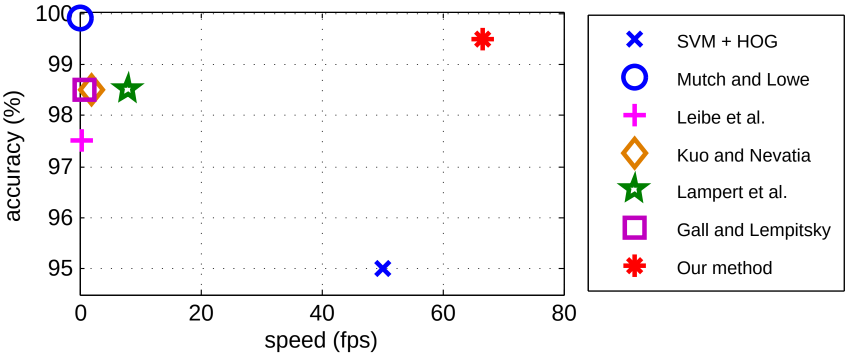


Figure 3.8: Visualizing accuracy and detection speed of our method compared to the state of the arts. In a very comparable detection accuracy, our method is almost 30 times faster than the state of the art for single scale car detection in the UIUC dataset.

without any horses as negative samples. The horses appear at a wide range of scales against cluttered backgrounds. For training, as suggested in [Ferrari et al., 2008] we use the first 50 positive and negative images for training and the rest for testing. Since the horse patches are not at a same size, we resize all the ground truth bounding box to the median aspect of all horses in the dataset (1.3) and then resize them again into positive patches of size 96 × 128 pixels. The negative training patches (of size 96 × 128) are randomly cropped from the negative samples (20 negative patches per sample, 1000 in total). The Recall/FPPI curve of $a_0 \geqslant 0.20$ are used for evaluation. Similar to the previous experiment, we used a pyramid approach to deal with the multi-scale object detection and bounding box approximation with the difference that the image pyramid is formed by scaling the test image from 0.3 to 1.6 times the image size.

Results. The results in Figure 3.9 and Table 3.3 show that our method outperforms the state-of-the-art approaches proposed by Monroy et al. [Monroy et al., 2011], Toshev et. al [Toshev et al., 2010], Villamizar et al. [Villamizar et al., 2012] and SVM+HOG in terms of detection time and performance. The detection rate achieved by our method at 1.0 FPPI (95.35 %) is slightly higher than those obtained by Monroy et al. and Toshev et al. (92.40% and 94.50%, respectively). In the case of complexity, our method respectively enjoys 12 and 100 times faster detection speed compared to Monroy et al. and Toshev et al. respectively. The reason is that Monroy's work integrates HOG features

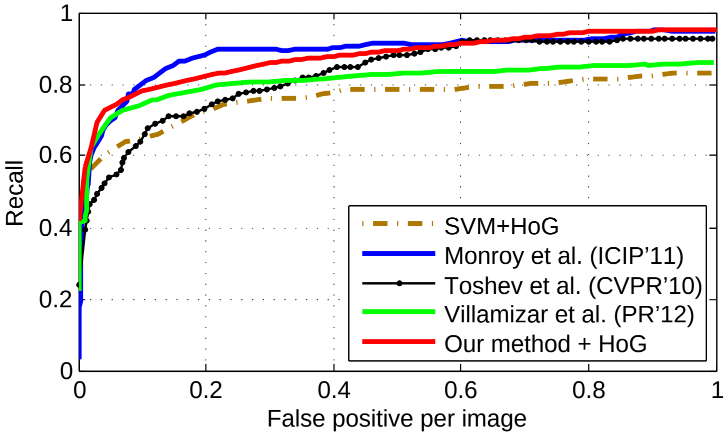


Figure 3.9: Recall versus FPPI ROC curve of our method compared to the state-of-the-art approaches on the INRIA horses dataset.

with curvature information to improve the detection performance. Moreover, the computational cost of this method is heavily affected by additional image processing procedures such as sophisticated edge extraction, connected element extraction for segment detection and distance accumulation computation. In Toshevs work, a boundary structure segmentation model is proposed for simultaneously object detection and segmentation. It's huge computational expense comes from superpixel mining, initial segmentation based on superpixels and an optimization problem that is solved using semidefinite programming relaxation. Our method, on the other hand, takes advantage of the mathematical simplicity and efficiency of correlation/convolution in the Fourier domain for super fast pattern detection. All that required by our method for detection is the FFT of feature channels, a set of element-wise multiplications in the Fourier domain and an IFFT on the correlation output over scale space. Some detection examples illustrated in Figure 3.10 show the robustness of our method against cluttered background, partial occlusion, extreme scale and illumination variations.

Method	Recall % (0.10 FPPI)	Recall % (1.0 FPPI)	detection time/image (sec.)
SVM + HOG	64.10	83.09	0.3
Monroy et al. [2011]	80.00	94.50	4
Toshev et al. [2010]	64.27	92.40	30
Villamizar et al. [2012]	74.77	86.000	1
Our method	**78.19**	**95.35**	**0.25**

Table 3.3: The detection performance (recall at 0.10 and 1.0 FPPI)and search time of the competing approaches versus the proposed technique on the INRIA horses dataset. Our method outperforms the others both in terms of test computation and detection rate.

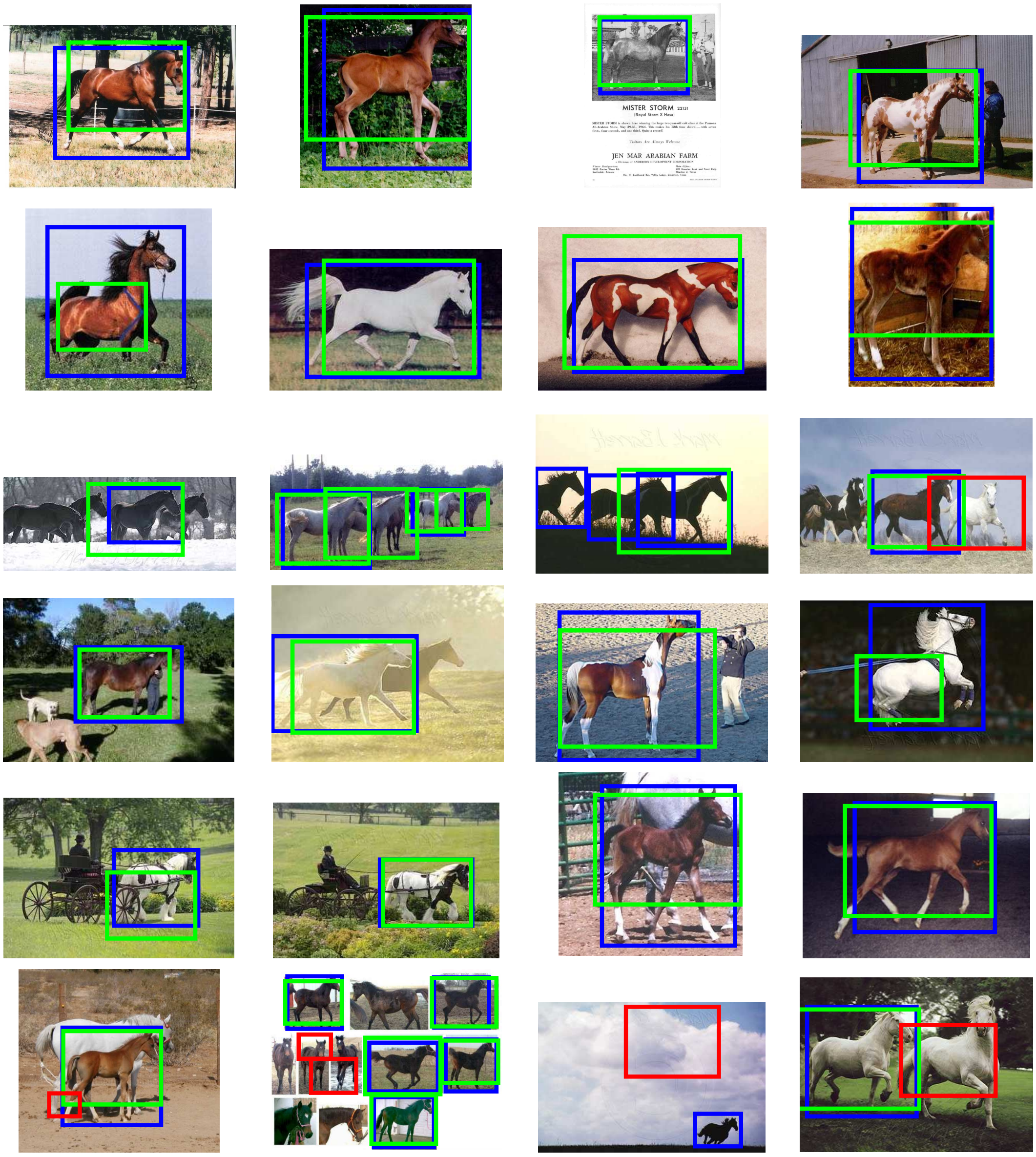


Figure 3.10: Detection results of our method over the INRIA horses dataset. Our proposed method is stable against cluttered background, illumination and scale changes. The ground truth, true positive and false positive detections are respectively shown by the blue, green and red boxes.

3.5 Chapter Summary

In this chapter we proposed a novel extension to correlation filter theory for learning multi-channel correlation filters. We introduced the objective of multi-channel correlation filters both in the spatial and frequency domain with identical closed-form solutions. We illustrated that the sparse-banded form of the objective in the Fourier domain allows one to find the optimal solution with very efficient memory usage and computational complexity. Specifically, we demonstrated that the required memory to learn a multi-channel correlation filter is independent of the number of training images allowing one to employ a large amount of training data during filter learning. We evaluated the proposed approach on classification (pedestrian) and detection (car and horse) in images. The experiments demonstrated the superior computation and efficient memory usage of our proposed approach compared to the linear classifier SVM and leading spatial detectors with near to state of the art detection/classification performance. The experiments and the key findings are summarized as follows.

Pedestrian Classification

- **dataset:** Daimler Pedestrian dataset including three sets of training and two sets of testing images. Each individual set contains 4800 positive and 5000 negative examples of size 36×18.
- **compared to:** linear SVM + HOG
- **challenges:** large pedestrian pose and clothing variations, poor image quality, low contrast to background
- **constraints:** single scale cropped patches
- **key findings:** (1) MCCF's training time linearly grows by increasing the number of training images. For SVM, the training time nonlinearly increases when the training size increases. (2) Unlike SVM, the required memory for MCCF training is constant and independent of the number

of training images. (3) MCCF achieved better detection rate compared to linear SVM with substantially less training images.

Car Detetion

- **dataset:** UIUC Cars dataset including 550 cars and 550 non-car training images of size 100 × 40.
- **compared to:** linear SVM + HOG, Mutch and Lowe [2006]; Leibe et al. [2008]; Kuo and Nevatia [2009]; Lampert et al. [2008]; Gall and Lempitsky [2009]
- **challenges:** large scale variations, cluttered background, large intra-class variations, extreme lighting, partial occlusion and varying image quality
- **constraints:** single side-view cars
- **key findings:** (1) MCCF is robust against scale variations through a simple pyramid scaling technique. (2) The detection performance achieved by MCCF is slightly lower than the state of the art for both single-scale and multi-scales testing images. (3) The detection speed of MCCF on the UIUC testing set is around 70 fps. The detection speed for the fastest previous technique with almost same detection rate is around 8 fps.

Horse Detection

- **dataset:** The INRIA horse dataset consists of 170 images with one or more side-views of horses as positive samples and 170 background images without any horses as negative samples.
- **compared to:** linear SVM + HOG, Monroy et al. [2011]; Toshev et al. [2010]; Villamizar et al. [2012]
- **constraints:** single side-view horses
- **challenges:** large scale variations, cluttered background, extreme outdoor

lighting, partial occlusion and different image quality

- **key findings:** Same as the car detection experiment, this experiment shows that the MCCF is able to detect multi-scale horses at a superior detection speed of around 4 fps. Compared to the state of the art with the same detection performance, MCCF is almost 16 times faster.

A question may that arise is why we evaluated our method on the pedestrians, cars and horses objects. We selected these object classes because of (1) their small intra-class variations (in a single view), and (2) the drawback of current correlation filters learning large intra-class variations. Note that this is not particularly a disadvantage of correlation filters, as many modern complicated classifiers/detectors are not robust against large intra-class variation and do not generalize well on unseen data. In fact, the major goal of these experiments is showing how the new approach can address some classical limitations of current correlation filters (e.g. using multi-channel features, here) by preserving their memory and computation efficiencies. In addition, it is worth to state that the proposed approach is not limited to single view car and horse detection, as was done in the experiments. One can easily perform multi-view horse and car detection by learning a set of filters that corresponded to different object views (e.g side, front, back views of cars) for view-invariant object detection.

Chapter 4

Multi-Channel Correlation Filters: From Image to Video

Human action recognition in video is a challenging problem in computer vision which has received substantial attention over the past years [Wang and Mori, 2011; Laptev et al., 2008; Fernandez and Kumar, 2013; Huang et al., 2011; de Campos et al., 2011]. Action recognition has been considered an important step for various video understanding problems such as video surveillance, video retrieval, gesture recognition, human-computer interaction and event analysis.

4.1 Spatial-Temporal Descriptors

From a vision perspective, a video data V with T frames can be represented as an spatial-temporal volume $V = \{v^t\}_{t=1}^{T}$, where T is the dimension of the video in the time domain (number of frames) and each v^t is a 2D image/frame (typically intensity) in the space domain, such that $t_1 < t_2 < ... < t_T$. Based on this representation, recent developments in feature description in static images (spatial domain) have been successfully extended to video data (spatial-temporal domain) to represent the video motion(temporal) and appearance(spatial) characteristics.

Various spatial-temporal representations for video have been evaluated by Dollar et al. [Dollar et al., 2005] for action classification including normalized pixel values, windowed optical flow and gradient magnitudes. The evaluation showed the superiority of the magnitude descriptor. Discarding orientation information and primarily using magnitudes for representation, however, made this approach very sensitive to illumination variations. This drawback was addressed by Kluser et al.'s [Kluser et al., 2008] 3D HOG descriptors, where the popular HOG descriptor [Dalal and Triggs, 2005] was extended to video sequences. The spatial-temporal HOG descriptors yield promising results and have been used in several recent works [Wang et al., 2009]. The spatio-temporal extensions of SIFT and SURF descriptors were defined in a similar manner to 3D HOG and can be found in [Paul Scovanner and Shah, 2007] and [Willems et al., 2008]. The major problem of these descriptors was their heavy memory usage and very high computations caused by learning a non-linear SVM over these multi-channel spatio-temporal features.

Apart from the above spatial-temporal features, some other leading representations are learned geometrical models of human body parts [Wang and Mori, 2011], space-time pattern templates, shape or form features [Blank et al., 2005] [Rodriguez et al., 2008], interest-point-based representations [Laptev et al., 2008], and motion/optical flow patterns [Efros et al., 2003] which are not in the scope of this thesis.

4.2 Correlation Filters for Action Recognition: Pros and Cons

The application of correlation filters has recently been investigated for human action detection/recognition and has yielded promising results [Ali and Lucey, 2010] [Rodriguez et al., 2008] [Fernandez and Kumar, 2013]. The main idea of these approaches is to represent actions using spatial-temporal volumes and learn a correlation filter in the 3D frequency domain that produces a peak at the origin of the action in both spatial and temporal domain.

More specifically, Rodriguez et al. [Rodriguez et al., 2008] has extended the Optimal Trade-off Maximum Average Correlation Height (OT-MACH) filter to 3D MACH and proposed action MACH to train 3D correlation filters for action recognition in video sequences. The main advantage of this approach is its closed-form solution for both scalar and vector features which makes the training process computationally very efficient. Moreover, detection can be made fast due to the efficiency of correlation in the frequency domain.

This method, however, suffers from three major limitations. First, action MACH trains a correlation filter that satisfies a set of criteria (e.g. maximizing the average correlation height) over all positive training examples. It has been shown by Ali and Lucey [Ali and Lucey, 2010] that action filters trained using action MACH are equivalent to the average of the action specific examples which suffers from poor generalization for unseen data and over-training for training examples. Second, action MACH only makes use of positive examples and ignores negative examples during learning process (according to its learning objective). This may result in training correlation filters with low discrimination power which do not perform well against large inter-class similarities (confusions among *walking*, *jogging* and *running* in [Rodriguez et al., 2008]). Finally, action MACH does not specify desired values over the entire correlation output of training examples. It was discussed in [Bolme et al., 2010] that this may increase the sensitivity to noise or produce smooth peaks which are difficult to accurately recognize/detect. A cross-correlation in the spatial domain is employed in [Rodriguez et al., 2008] to deal with smooth peaks. This cross-correlation in the spatial domain, however, resulted in a very high computational cost.

4.3 Action MCCF

We propose to apply multi-channel correlation filters, see Chapter 3, for human action recognition in video [Kiani et al., 2014b, 2013]. The core idea is that each video sequence with N time-ordered frames can be considered as a multi-channel signal with N channels (scalar features such as intensity, gradient magnitude and temporal derivatives). Given a set of training examples (including both positive and negative videos) and their corresponding correlation outputs, the goal is to learn a multi-channel action filter in the frequency domain that produces the desired correlation outputs when correlated with corresponding training examples.

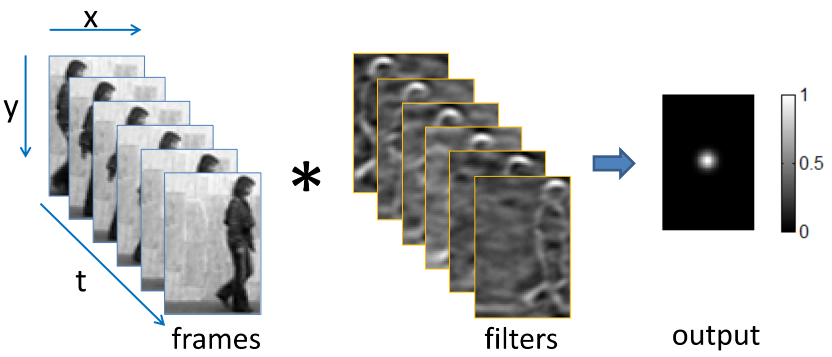
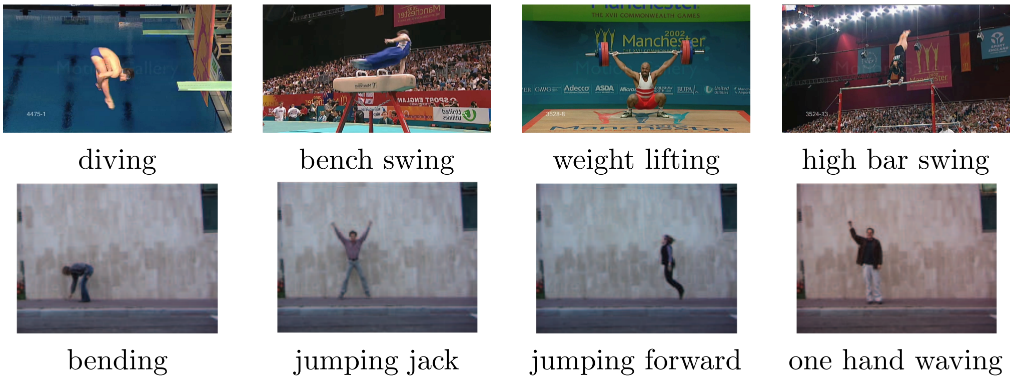


Figure 4.1: An example of learning a multi-channel correlation filter for *walking* action cycle. The action cycle is represented by N intensity frames and the goal is to learn an N-channel correlation filter that returns a desired correlation output (a 2D Gaussian) when correlated with training action examples.

An example of learning a multi-channel correlation filter for the *walking* action is shown in Figure 4.1. For training examples with N frames of scalar feature (e.g. intensity), the temporal dimension (number of channels) of the trained MCCF is equal to N. Fortunately, the MCCF can be easily extended to vector features such as HOG and SIFT. In the case of vectored features, an action MCCF of a video with N frames in a feature space of M channels (e.g. M = 5 for a 5-bin HOG), the number of the filter channels is $M \times N$.

The advantages of MCCF for action recognition are as follows [Kiani et al., 2014b]. First, the ridge regression form of MCCF in the spatial domain allows one to specify the desired values for the entire correlation output (e.g. using the 2D Gaussian function). This significantly reduces the sensitivity to noise and practically produces sharp peaks for more accurate detection/recognition with no need of any post-processing as done in action MACH [Rodriguez et al., 2008]. Second, the MCCF is able to use both positive and negative training examples. This allows us to train discriminative action filters which are more robust against large inter-class similarities, e.g the similarities among *jogging*, *running* and *walking* actions. Finally, real-time action recognition can be performed due to the low computation of MCCF in the frequency domain.



diving bench swing weight lifting high bar swing

bending jumping jack jumping forward one hand waving

Figure 4.2: Examples of actions from (top) the UCF sport, and (bottom) the Weizmann datasets.

4.4 Experiments

4.4.1 Databases

We used two publicly available action datasets for evaluation: Weizmann dataset [Blank et al., 2005], and UCF sport dataset [Rodriguez et al., 2008].

Weizmann Dataset: The Weizmann dataset contains 10 actions (bending, running, walking, skipping, jumping jack, jumping forward, jumping in place, jumping sideway, waving two hands and waving one hand) performed by nine different subjects over a static background with slight changes in view point, scale and illumination. There all 90 sequences in this dataset, each contains about 40 - 120 frames of size 189 × 144.

UCF Sport Datset: The UCF sport dataset is more challenging and contains 10 different types of human actions including diving, kicking, weight lifting, horse riding, running, walking, skateboarding, golf swinging, swinging at the high bar and swinging on the pommel horse (and floor) performed by a different number of subjects. This dataset consists of 150 video clips filmed under challenging situations with cluttered background, large lightning/scaling changes, and significant intra-class variations.

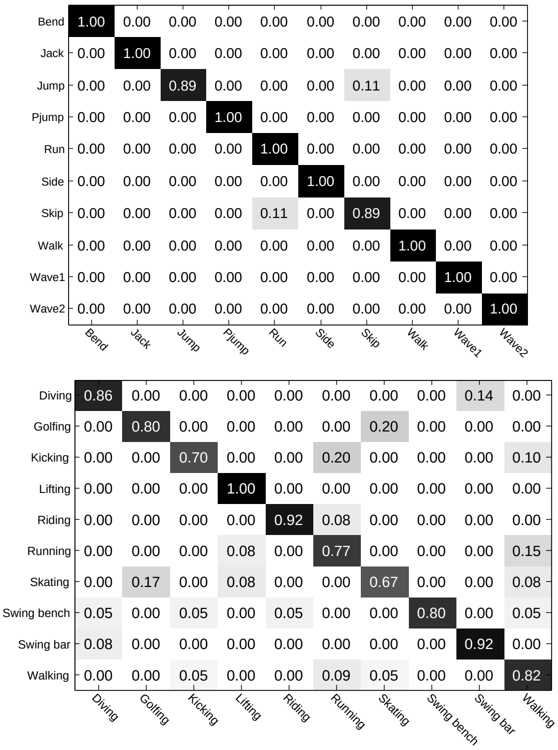
4.4.2 Feature Extraction and Desired Correlation Outputs

We evaluated our method using different types of features including normalized intensity, edge magnitude, temporal derivative and multi-channel HOG descriptors computed over 5 orientation bins normalized by cell and block sizes of 5×5 and 3×3, respectively. We used HOG for comparing our technique with the state of the arts. To compensate for the large illumination variation, all frames were power-normalized.

For positive examples, a 2D Gaussian with spatial variance of 2 was employed to define the desired correlation outputs whose peak was centered at the center of the actor at last frame. A 2D matrix of zero values formed the desired correlation outputs of negative examples.

4.4.3 Filter Training and Testing

The spatial-temporal annotations from [Tian et al., 2013] were used to extract training action cycles of both datasets. The cycles of each action class ($\sim$10 - 60 frames per cycle) were carefully aligned to be consistent in both the spatial and temporal domains. Given a set of positive and negative training examples and their corresponding desired correlation outputs, the action specific filter was trained using Equation 3.13. For testing, we applied the MCCF filter trained for each class on a test video, and the label of the filter with maximum Peak-to-Sidelobe Ratio (PSR) [Kumar, 2005] is assigned. To deal with scaling in the UCF sport dataset, a simple pyramid approach was employed to scan testing videos across different scales (from 0.4 to 1.5 of scaling-step 1.5) and the correlation output with maximum PSR across the pyramid was selected for each video. For actions with whole-body translation (e.g. *walking* and *skipping*) we trained two action filters (left-to-right and right-to-left) by vertically flipping the training examples. We performed leave-one-subject-out cross-validation for the Weizmann dataset (8 subjects for training and the other one for testing) [Blank et al., 2005] and leave-one-sample-out for UCF sport dataset (one video sample



	Bend	Jack	Jump	Pjump	Run	Side	Skip	Walk	Wave1	Wave2
Bend	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Jack	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Jump	0.00	0.00	0.89	0.00	0.00	0.00	0.11	0.00	0.00	0.00
Pjump	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00
Run	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00
Side	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00
Skip	0.00	0.00	0.00	0.00	0.11	0.00	0.89	0.00	0.00	0.00
Walk	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00
Wave1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00
Wave2	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00

	Diving	Golfing	Kicking	Lifting	Riding	Running	Skating	Swing bench	Swing bar	Walking
Diving	0.86	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.14	0.00
Golfing	0.00	0.80	0.00	0.00	0.00	0.00	0.20	0.00	0.00	0.00
Kicking	0.00	0.00	0.70	0.00	0.00	0.20	0.00	0.00	0.00	0.10
Lifting	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	0.00
Riding	0.00	0.00	0.00	0.00	0.92	0.08	0.00	0.00	0.00	0.00
Running	0.00	0.00	0.00	0.08	0.00	0.77	0.00	0.00	0.00	0.15
Skating	0.00	0.17	0.00	0.08	0.00	0.00	0.67	0.00	0.00	0.08
Swing bench	0.05	0.00	0.05	0.00	0.05	0.00	0.00	0.80	0.00	0.05
Swing bar	0.08	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.92	0.00
Walking	0.00	0.00	0.05	0.00	0.00	0.09	0.05	0.00	0.00	0.82

Figure 4.3: The confusion matrix of our method for (top) the Weizmann, and (bottom) the UCF dataset (best viewed in pdf).

for testing and the rest for training) Rodriguez et al. [2008].

4.4.4 Quantitative Results

The confusion matrix of our approach on the Weizmann dataset is illustrated in Figure 4.3 (top), showing 100% accuracy of our method for all action classes except *jump* and *skip*. Two confusions occurred between *jump* versus *skip* and *skip* versus *run* actions caused by their significant motion/appearance similarities. Figure 4.3 (bottom) shows our confusion matrix on the UCF sport dataset. The proposed method achieved promising results for most of the actions. There are more errors in *skating*, *running* and *kicking*. This might be caused by the disadvantage of correlation filters (and most of the linear classifiers) to deal with very large intra-class variations and inter-class similarities. In addition, HOG features are not able to capture motion features in the temporal domain which has been shown to be more robust to large motion similarities [Wang et al., 2009]. We can increase the robustness of MCCF against large inter-class motion similar-

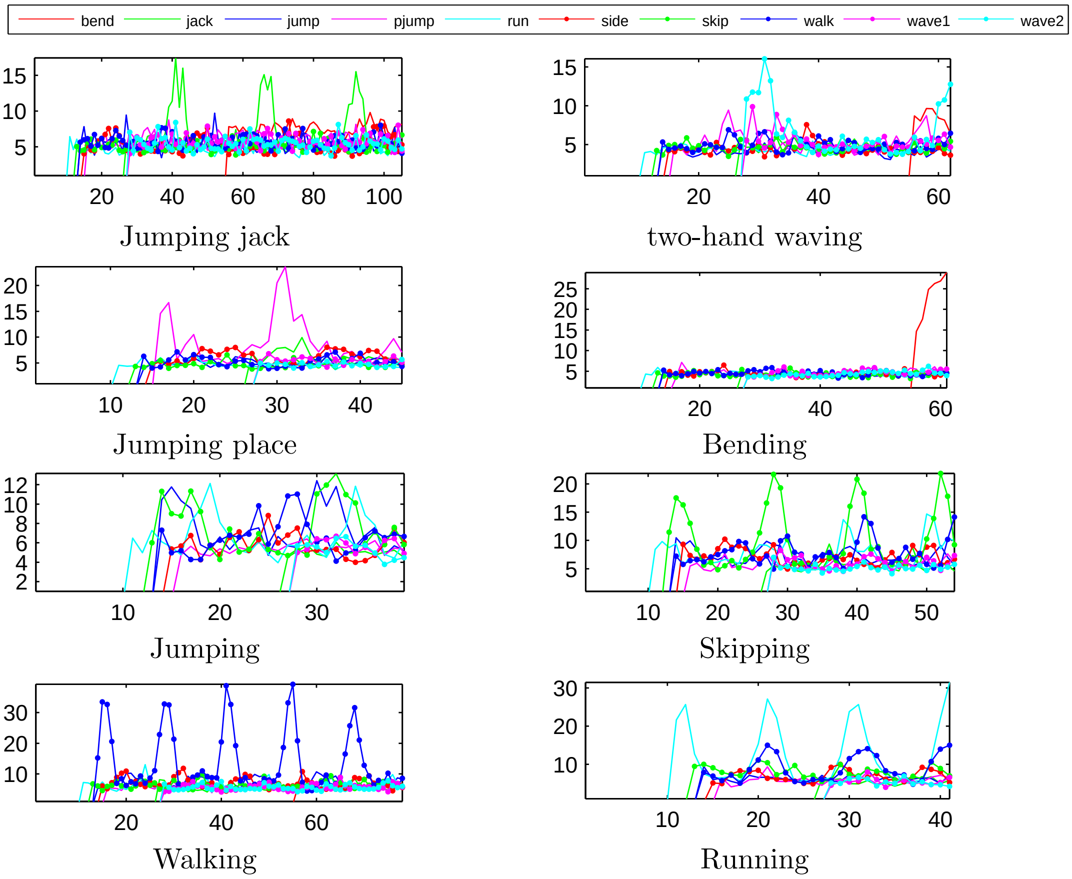


Figure 4.4: Qualitative results. PSRs versus frame number obtained by applying 10 trained action MCCFs on some selected testing videos. The *jumping* sequence is a misclassified case since the PSR produced by the *skipping* filter is slightly higher than *jumping* filter. For the successful cases, the high peaks (e.g. there are 5 high peaks in *walking*) correspond to action cycles (best viewed in color).

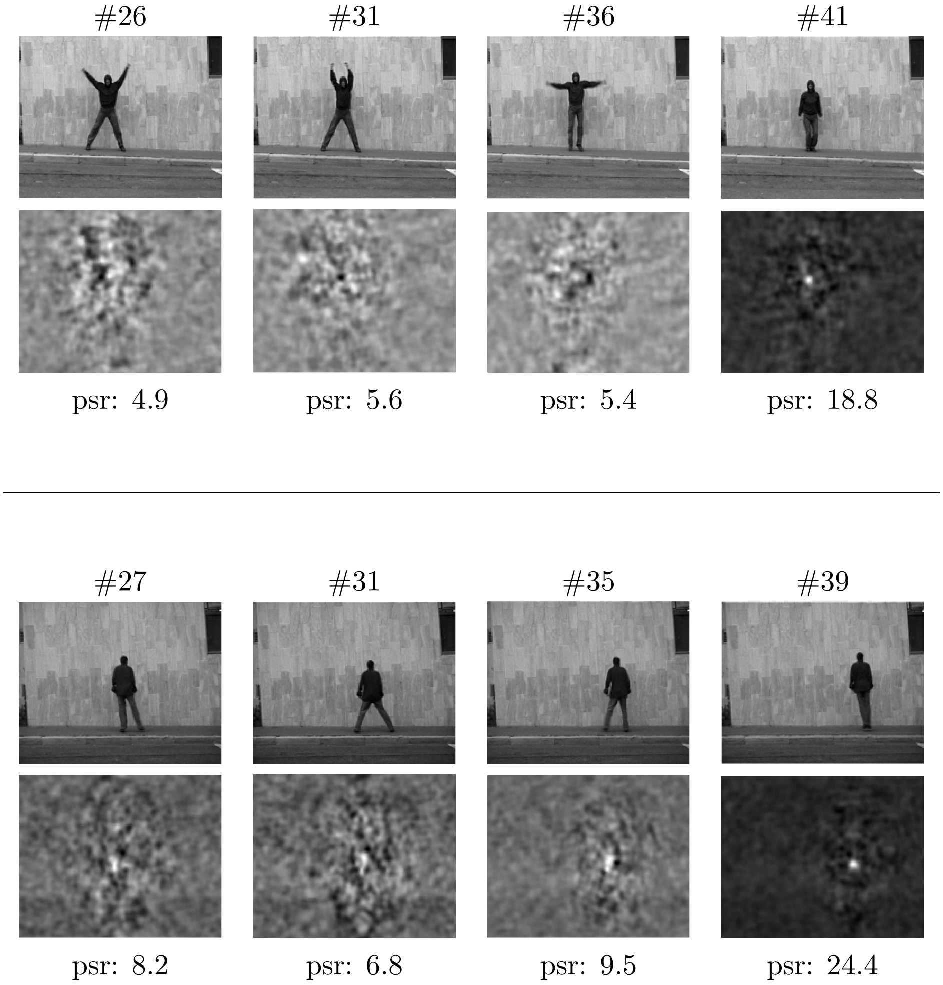


Figure 4.5: Qualitative results. Selected frames of two actions with corresponding correlation outputs. Please refer to **"Qualitative Results"** for more explanation.

ities using more discriminative spatio-temporal features such as 3DHOG [Kluser et al., 2008] and HOG/HOF descriptors [Laptev et al., 2008].

Table 4.1 provides a comparison of our method with those previously reported in the literature on the Weizmann and UCF sport datasets. For the Weizmann dataset, the highest mean recognition rate (100%) achieved by Huang et al. [Huang et al., 2011]. This method, however, was evaluated on 9 actions (the *skip* action was discarded) using a rich combination of optical flow and color histogram features (the accuracy of our method would be 100% if we discard the *skip* action too). In addition, feature extraction, tracking and stabilization made Huang's method very slow. Our accuracy (97.8%) is slightly lower than this method on more action classes with real-time recognition speed, and higher than those reported by [de Campos et al., 2011] and [Wang et al., 2009]. For the UCF sport dataset, our method achieved competitive accuracy compared to the stat-of-the-art. The best performance obtained by Cai et al. [Cai et al., 2013] using the dynamic structure preserving map (DSPM) technique. This technique, however, suffers from heavy computations. For both datasets, the action MACH [Rodriguez et al., 2008] obtained the lowest performance of 86.6% (Weizmann) and 69.2% (UCF sport) due to its sensitivity to inter-class similarities and poor generalization.

Moreover, the result reported in Table 4.1 shows that the accuracy of all methods on the Weizmann dataset is much higher than the UCF sport dataset. The reason is that unlike the Weizzmann dataset which was filmed under controlled situations (e.g. plain background, stationary camera, single viewpoint and slight lighting changes), the UCF sprot dataset contains realistic actions captured under real-world circumstances including stationary and moving camera, diversity of cluttered backgrounds, different viewpoints, illumination changes, scaling and low resolution. This implies that the action MCCF performs quite well and very competitive on action videos captured under fairly controlled situations. Whereas, similar to the other approaches its accuracy is degraded when action videos contain real-world situations such as background clutter and large action variations.

We also evaluated our method with various types of image representations including normalized intensity, edge magnitude, temporal derivative and 5-bin HOG descriptors. The results reported in Table 4.2 show that our method is fairly robust against different feature types and performs quite well using simple features such as image intensity and edge magnitude. Action recognition using these simple features can significantly improve the training and testing efficiencies in terms of feature extraction time, recognition speed and memory usage. As expected, the best performance belongs to the HOG descriptor.

4.4.5 Qualitative Results

Figure 4.4 shows the PSRs obtained by applying the trained action filters on some testing examples such as the *Jumping jack*, *Running*, *Jumping side* and *Jumping* (from the Weizmann dataset) versus frame numbers. Obviously, the PSRs produced by same-class filter are significantly higher than those produced by different-class filters. The *Jumping* sequence is a failed case, where the PSRs produced by *Skip* filter over the test video are slightly higher than those by *Jump* filter. Interestingly, our method is able to produce high PSR for each action cycle through the test sequences. For example, the *Jumping side* video contains two ground truth cycles of *Jumping side* action which correspond to the (two) high PSRs. The high PSRs can be further used to accurately *detect* the action occurrences across the test videos. Figure 4.5 illustrates some selected frames of *Jumping jack* (left) and *Jumping side* (right) action cycles with their corresponding correlation outputs produced by *Jumping jack* and *Jumping side* filters, respectively. For each frame, the frame number and PSR value are shown. The maximum peak almost occurs at the last frame of each action cycle (temporal domain) upon the location of the actor (spatial domain).

4.4.6 Runtime Complexity

The average runtime of MCCF (HOG descriptors for each frame) for a $144 \times 180 \times 200$ Weizmann testing sequence was 8.15 seconds (real-time) on a Core i7,

Method	Weizmann	UCF sport
Huang et al. [Huang et al., 2011]	100%	-
Cai et al. [Cai et al., 2013]	98.7%	90.6%
Wang et al. [Wang et al., 2009]	97.8 %	77.4%
Campos et al. [de Campos et al., 2011]	96.7 %	80.0%
Rodriguez et al. [Rodriguez et al., 2008]	86.6%	69.2%
Yeffet & Wolf [Yeffet and Wolf, 2009]	-	79.3%
Our method	97.8%	82.6%

Table 4.1: Mean accuracy of our method compared to the state-of-the-art on the Weizmann and UCF sport datasets.

Normalized intensity	Edge magnitude	Temporal derivative	HoG (5 bins)
89.4%	91.2%	92.3%	97.8%

Table 4.2: Mean accuracy of different features (the Weizmann dataset)

3.40 GHz. While, action MACH and [Blank et al., 2005] reported a runtime of 18.65 seconds (without spatial normalized correlation, with normalization it takes about 11 minutes for the same video) and 30 minutes for a same video of this dataset [Blank et al., 2005], respectively. Moreover, most of the other methods in Table 4.1 employed the non-linear SVM over a sliding window (in the spatial and temporal domains) for classification which is shown to be much slower than the MCCF [Kiani et al., 2013].

4.5 Chapter Summary

In this Chapter, we proposed to apply multi-channel correlation filters for human action recognition in videos. Towards this purpose, we present a video using spatio-temporal feature channels, and then use MCCF to learn a spatio-temporal correlation filter for each action class. We discussed that the MCCF can efficiently address the drawbacks of current correlation-based action recognition techniques. The experiments showed the competitive performance of our approach against the state of the arts with superior computational efficiency and real-time recognition speed.

Chapter 5

Correlation Filters with Limited Boundaries

The traditional correlation filter objective described in Equation (3.2) produces a filter that is particularly sensitive to misalignment in translation. A highly undesirable property when attempting to detect or track an object in terms of translation. This sensitivity is obtained due to the circular shift operator $\mathbf{x}[\Delta\boldsymbol{\tau}]$, where $\Delta\boldsymbol{\tau} = [\Delta x, \Delta y]^\top$ denotes the 2D circular shift in x and y. It has been well noted in correlation filter literature [Kumar, 2005] that this circular-shift alone tends to produce filters that do not generalize well to other types of appearance variation (e.g. illumination, viewpoint, scale, rotation, etc.). This generalization issue can be somewhat mitigated through the judicious choice of non-zero regularization parameter λ, and/or through the use of an ensemble $N > 1$ of training observations that are representative of the type of appearance variation one is likely to encounter.

A deeper problem with the objective in Equation 3.2, however, is that the shifted image patches $\mathbf{x}[\Delta\boldsymbol{\tau}]$ at all values of $\Delta\boldsymbol{\tau} \in \mathbb{C}$, except where $\Delta\boldsymbol{\tau} = \mathbf{0}$, are not representative of image patches one would encounter in a normal correlation operation (Figure 5.1(c)). In signal-processing, one often refers to this as the *boundary effect.*

In computer vision, the boundary effect causes learning correlation filters from an unbalanced set of “real-world” and “synthetic” examples. These synthetic examples are created through the application of a circular shift on the real-world examples, and are supposed to be representative of those examples at different translational shifts. We use the term synthetic, as all these shifted examples are plagued by circular boundary effects and are not truly representative of the shifted example (see Figure 5.1(c)). As a result the training set used for learning the template is extremely unbalanced with one real-world example for every $D-1$ synthetic examples (where D is the dimensionality of the examples) [Kiani et al., 2014a].

These boundary effects can dramatically affect the resulting performance of the estimated template. Fortunately, these effects can be largely removed if the correlation filter objective is slightly augmented, but has to be now solved in the spatial rather than frequency domains. Unfortunately, this shift to the spatial domain destroys the computational efficiency that make correlation filters so attractive.

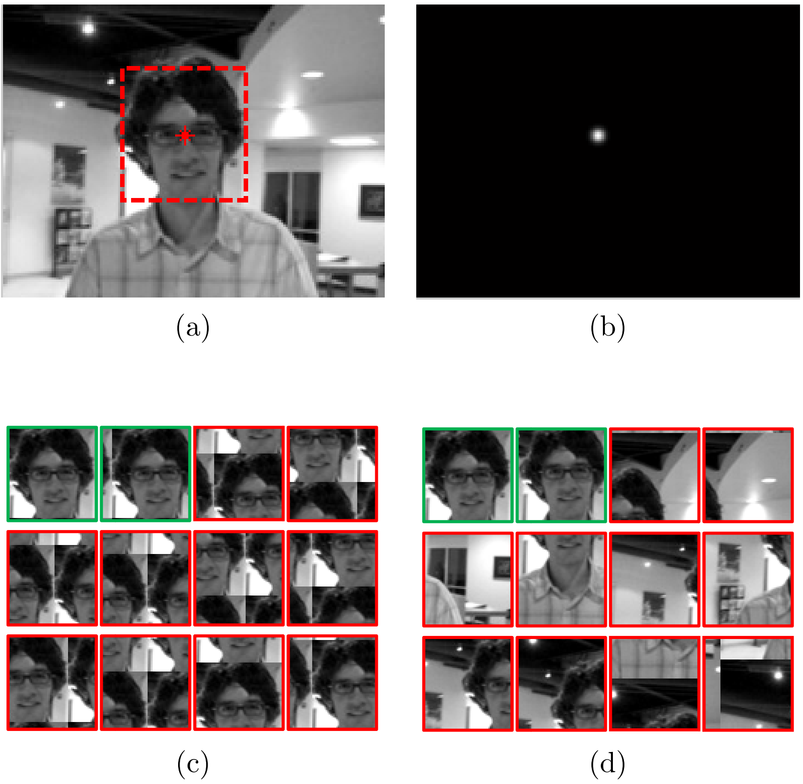
5.1 Proposed Approach

One simple way to circumvent this problem spatially is to allow the training signal $\mathbf{x} \in \mathbb{R}^T$ to be a larger size than the filter $\mathbf{h} \in \mathbb{R}^D$ such that $T > D$ [Kiani et al., 2014a]. Through the use of a $D \times T$ masking matrix $\mathbf{P}$ one can reformulate Equation (3.2) as,

$$E(\mathbf{h}) = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{T} ||\mathbf{y}_i(j) - \mathbf{h}^\top \mathbf{P} \mathbf{x}_i[\Delta \boldsymbol{\tau}_j]||_2^2 + \frac{\lambda}{2} ||\mathbf{h}||_2^2 \ . \tag{5.1}$$

The masking matrix $\mathbf{P}$ of ones and zeros encapsulates what part of the signal should be active/inactive. The central benefit of this augmentation in Equation 5.1 is the dramatic increase in the proportion of examples unaffected by boundary effects ($\frac{T-D+1}{T}$ instead of $\frac{1}{D}$). From this insight it becomes clear that if one chooses $T >> D$ then boundary effects become greatly diminished (Figure 5.1(d)). The computational cost $\mathcal{O}(D^3 + NTD)$ of solving this objective is only slightly larger than the cost of Equation 3.2, as the role of $\mathbf{P}$ in practice can be accomplished efficiently through a lookup table.

As mentioned earlier, posing the objective in the Fourier domain has been a



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

Figure 5.1: (a) Defines the example of fixed spatial support within the image from which the peak correlation output should occur. (b) The desired output response, based on (a), of the correlation filter when applied to the entire image. (c) A subset of patch examples used in a canonical correlation filter where green denotes a non-zero correlation output, and red denotes a zero correlation output in direct accordance with (b). (d) A subset of patch examples used in our proposed correlation filter. Note that our proposed approach uses patches stemming from different parts of the image, whereas the canonical correlation filter simply employs circular shifted versions of the same single patch. The central dilemma in this proposal is how to perform (d) efficiently in the Fourier domain. The two last patches of (d) show that $\frac{D}{T}$ patches near the image border are affected by circular shift in our method which can be greatly diminished by choosing $D << T$.

standard solution to deal with the inefficient complexity of learning canonical correlation filters in the spatial domain. A problem that arises, however, when one attempts to apply the same Fourier insight to the augmented spatial objective in Equation 5.1 is its non capability to the Fourier domain ,

$$E(\mathbf{h}) = \frac{1}{2}\sum_{i=1}^{N} ||\hat{\mathbf{y}}_i - \text{diag}(\hat{\mathbf{x}}_i)^\top \sqrt{D}\mathbf{F}\mathbf{P}^\top \mathbf{h}||_2^2 + \frac{\lambda}{2}||\mathbf{h}||_2^2 \ . \tag{5.2}$$

Unfortunately, since we are enforcing a spatial constraint the efficiency of this objective balloons to $\mathcal{O}(D^3 + ND^2)$ as $\mathbf{h}$ *must* be solved in the spatial domain.

5.1.1 Augmented Lagrangian

Our proposed approach for solving Equation (5.2) involves the introduction of an auxiliary variable $\hat{\mathbf{g}}$ [Kiani et al., 2014a],

$$\begin{aligned} E(\mathbf{h}, \hat{\mathbf{g}}) &= \frac{1}{2}\sum_{i=1}^{N} ||\hat{\mathbf{y}}_i - \text{diag}(\hat{\mathbf{x}}_i)^\top \hat{\mathbf{g}}||_2^2 + \frac{\lambda}{2}||\mathbf{h}||_2^2 \\ &\text{s.t.} \quad \hat{\mathbf{g}} = \sqrt{D}\mathbf{F}\mathbf{P}^\top \mathbf{h} \ . \end{aligned} \tag{5.3}$$

We propose to handle the introduced equality constraints through an Augmented Lagrangian Method (ALM) [Boyd, 2010]. The augmented Lagrangian of our proposed objective can be formed as,

$$\begin{aligned} \mathcal{L}(\hat{\mathbf{g}}, \mathbf{h}, \hat{\boldsymbol{\zeta}}) &= \frac{1}{2}\sum_{i=1}^{N} ||\hat{\mathbf{y}}_i - \text{diag}(\hat{\mathbf{x}}_i)^\top \hat{\mathbf{g}}||_2^2 + \frac{\lambda}{2}||\mathbf{h}||_2^2 \\ &+ \quad \hat{\boldsymbol{\zeta}}^\top (\hat{\mathbf{g}} - \sqrt{D}\mathbf{F}\mathbf{P}^\top \mathbf{h}) \\ &+ \quad \frac{\mu}{2}||\hat{\mathbf{g}} - \sqrt{D}\mathbf{F}\mathbf{P}^\top \mathbf{h}||_2^2 \end{aligned} \tag{5.4}$$

where μ is the penalty factor that controls the rate of convergence of the ALM, and $\hat{\boldsymbol{\zeta}}$ is the Fourier transform of the Lagrangian vector needed to enforce the newly introduced equality constraint in Equation 5.3.

5.1.2 Optimization by Alternating Direction Method of Multipliers

ALMs are not new to learning and computer vision, and have recently been used to great effect in a number of applications [Boyd, 2010; Del Bue et al., 2011]. Specifically, the Alternating Direction Method of Multipliers (ADMMs) has provided a simple but powerful algorithm that is well suited to distributed convex optimization for large learning and vision problems. A full description of AD-

MMs is outside the scope of this work (readers are encouraged to inspect [Boyd, 2010] for a full treatment and review), but they can be loosely interpreted as applying a Gauss-Seidel optimization strategy to the augmented Lagrangian objective. Such a strategy is advantageous as it often leads to extremely efficient subproblem decompositions. A full description of our proposed algorithm can be seen in Algorithm 1. We detail each of the subproblems as follows:

Subproblem g:

$$\begin{aligned} \hat{\mathbf{g}}^* &= \arg\min \mathcal{L}(\hat{\mathbf{g}}; \hat{\mathbf{h}}, \hat{\boldsymbol{\zeta}}) \\ &= (\hat{\mathbf{s}}_{xy} + \mu\hat{\mathbf{h}} - \hat{\boldsymbol{\zeta}}) \circ^{-1} (\hat{\mathbf{s}}_{xx} + \mu\mathbf{1}) \end{aligned} \tag{5.5}$$

where $\hat{\mathbf{h}} = \sqrt{D}\mathbf{F}\mathbf{P}^\top\mathbf{h}$. In practice $\hat{\mathbf{h}}$ can be estimated extremely efficiently by applying a FFT to $\mathbf{h}$ padded with zeros implied by the $\mathbf{P}^\top$ masking matrix. $\hat{\mathbf{s}}_{xx}$ and $\hat{\mathbf{s}}_{xy}$ are respectively the average auto-spectral and cross-spectral energies of the training images in the Frequency domain as defined in Equation 2.26.

Subproblem h:

$$\begin{aligned} \mathbf{h}^* &= \arg\min \mathcal{L}(\mathbf{h}; \mathbf{g}, \boldsymbol{\zeta}) \\ &= (\mu + \frac{\lambda}{\sqrt{D}})^{-1}(\mu\mathbf{g} + \boldsymbol{\zeta}) \end{aligned} \tag{5.6}$$

where $\mathbf{g} = \frac{1}{\sqrt{D}}\mathbf{P}\mathbf{F}^\top\hat{\mathbf{g}}$ and $\boldsymbol{\zeta} = \frac{1}{\sqrt{D}}\mathbf{P}\mathbf{F}^\top\hat{\boldsymbol{\zeta}}$. In practice both $\mathbf{g}$ and $\boldsymbol{\zeta}$ can be estimated extremely efficiently by applying an inverse FFT and then applying the lookup table implied by the masking matrix $\mathbf{P}$.

5.1.3 Lagrange Multiplier Update

$$\hat{\boldsymbol{\zeta}}^{(i+1)} \leftarrow \hat{\boldsymbol{\zeta}}^{(i)} + \mu(\hat{\mathbf{g}}^{(i+1)} - \hat{\mathbf{h}}^{(i+1)}) \tag{5.7}$$

where $\hat{\mathbf{g}}^{(i+1)}$ and $\hat{\mathbf{h}}^{(i+1)}$ are the current solutions to the above subproblems at iteration $i + 1$ within the iterative ADMM.

5.1.4 Choice of Convergence Rate $\boldsymbol{\mu}$

A simple and common [Boyd, 2010] scheme for selecting μ is the following

$$\mu^{(i+1)} = \min(\mu_{\max}, \beta\mu^{(i+1)}) \ . \tag{5.8}$$

We found experimentally $\mu^{(0)} = 10^{-2}$, $\beta = 1.1$ and $\mu_{\max} = 20$ to perform well.

5.1.5 Computational Analysis

Inspecting Algorithm 1 the dominant cost per iteration of the ADMM optimization process is $\mathcal{O}(T \log T)$ for FFT. There is a pre-computation cost (before the iterative component, steps 4 and 5) in the algorithm for estimating the auto- and cross-spectral energy vectors $\hat{\mathbf{s}}_{xx}$ and $\hat{\mathbf{s}}_{xy}$ respectively. This cost is $\mathcal{O}(NT \log T)$ where N refers to the number of training signals. Given that K represents the number of the ADMM iterations the overall cost of the algorithm is therefore $\mathcal{O}([N + K]T \log T)$ [Kiani et al., 2014a].

Algorithm 1 Our approach using ADMMs

```
1: Intialize $\mathbf{h}^{(0)}$, $\boldsymbol{\zeta}^{(0)}$.
2: Pad with zeros and apply FFT: $\sqrt{D}\mathbf{F}\mathbf{P}^\top\mathbf{h}^{(0)} \rightarrow \hat{\mathbf{h}}^{(0)}$.
3: Apply FFT: $\sqrt{D}\mathbf{F}\boldsymbol{\zeta}^{(0)} \rightarrow \hat{\boldsymbol{\zeta}}^{(0)}$.
4: Estimate auto-spectral energy $\hat{\mathbf{s}}_{xx}$ using Eqn. (3.8).
5: Estimate cross-spectral energy $\hat{\mathbf{s}}_{xy}$ using Eqn. (3.8).
6: $i = 0$
7: repeat
8:     Solve for $\hat{\mathbf{g}}^{(i+1)}$ using Eqn. (5.5), $\hat{\mathbf{h}}^{(i)}$ & $\hat{\boldsymbol{\zeta}}^{(i)}$.
9:     Inverse FFT then crop: $\frac{1}{\sqrt{D}}\mathbf{P}\mathbf{F}^\top\hat{\mathbf{g}}^{(i+1)} \rightarrow \mathbf{g}^{(i+1)}$.
10:    Inverse FFT then crop: $\frac{1}{\sqrt{D}}\mathbf{P}\mathbf{F}^\top\hat{\boldsymbol{\zeta}}^{(i+1)} \rightarrow \boldsymbol{\zeta}^{(i+1)}$.
11:    Solve for $\mathbf{h}^{(i+1)}$ using Eqn. (5.6), $\mathbf{g}^{(i+1)}$ & $\boldsymbol{\zeta}^{(i)}$.
12:    Pad and apply FFT: $\sqrt{D}\mathbf{F}\mathbf{P}^\top\mathbf{h}^{(i+1)} \rightarrow \hat{\mathbf{h}}^{(i+1)}$.
13:    Update Lagrange multiplier vector Eqn. (5.7).
14:    Update penalty factor Eqn. (5.8).
15:    $i = i + 1$
16: until $\hat{\mathbf{g}}, \mathbf{h}, \hat{\boldsymbol{\zeta}}$ has converged
```

5.2 Extensions to MCCF

Combining the objective of correlation filters with limited boundaries, Equation 5.1, with that of multi-channel correlation filters, Equation 3.10, we propose a general framework in the spatial domain to learn multi-channel correlation filter with limited boundaries $\{\mathbf{h}_k\}_{k=1}^{K}$ by minimizing the following objective function:

$$\begin{aligned} E(\mathbf{h}) \quad &= \quad \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{T}||\mathbf{y}_i(j) - \sum_{k=1}^{K}\mathbf{h}^{(k)\top}\mathbf{P}\mathbf{x}_i^{(k)}[\Delta\boldsymbol{\tau}_j]||_2^2 + \\ & \quad\quad \frac{\lambda}{2}\sum_{k=1}^{K}||\mathbf{h}^{(k)}||_2^2 \end{aligned} \tag{5.9}$$

Similar to the single-channel case, we employ an auxiliary variable $\hat{\mathbf{g}}$ to (i) express the circular shift operator $[\Delta\boldsymbol{\tau}_j]$ in the spatial domain equivalently by a Hadamard product in the Fourier domain, and (ii) apply the masking matrix $\mathbf{P}$ in the spatial domain to force small spatial supports on $\mathbf{h}$. The new objective using the auxiliary variable $\hat{\mathbf{g}}$ is defined as:

$$\begin{aligned} E(\mathbf{h}, \hat{\mathbf{g}}) \quad &= \quad \frac{1}{2}\sum_{i=1}^{N}||\hat{\mathbf{y}}_i - \hat{\mathbf{X}}_i\hat{\mathbf{g}}||_2^2 + \frac{\lambda}{2}||\mathbf{h}||_2^2 \\ & \quad\quad \text{s.t.} \quad \hat{\mathbf{g}} = \sqrt{D}(\mathbf{F}\mathbf{P}^\top \otimes \mathbf{I}_K)\mathbf{h} \quad . \end{aligned} \tag{5.10}$$

where $\mathbf{h} = [\mathbf{h}^{(1)\top}, \ldots, \mathbf{h}^{(K)\top}]^\top$ and $\hat{\mathbf{g}} = [\hat{\mathbf{g}}^{(1)\top}, \ldots, \hat{\mathbf{g}}^{(K)\top}]^\top$ respectively show the $KD \times 1$ over-complete representations of $\mathbf{h}$ and $\hat{\mathbf{g}}$ by concatenating their K channels. Note that the length of the vectorized $\mathbf{h}^{(k)}$ and $\hat{\mathbf{g}}^{(k)}$ are D and T, respectively, where $D \ll T$ and the sparse banded matrix $\hat{\mathbf{X}}_i$ is defined as $\hat{\mathbf{X}}_i = [\text{diag}(\hat{\mathbf{x}}_i^{(1)})^\top, \ldots, \text{diag}(\hat{\mathbf{x}}_i^{(K)})^\top]$ of size $T \times KT$.

In a same manner, we handle the equality constraint caused by the new auxiliary variable using an Augmented Lagrangian Method. The augmented lagrangian of the above equation is formulated as below:

$$\begin{aligned}\mathcal{L}(\hat{\mathbf{g}},\mathbf{h},\hat{\boldsymbol{\zeta}}) &= \frac{1}{2}\sum_{i=1}^{N}||\hat{\mathbf{y}}_i - \hat{\mathbf{X}}_i\hat{\mathbf{g}}||_2^2 + \frac{\lambda}{2}||\mathbf{h}||_2^2 \\ &+ \hat{\boldsymbol{\zeta}}^\top(\hat{\mathbf{g}} - \sqrt{D}(\mathbf{F}\mathbf{P}^\top \otimes \mathbf{I}_K)\mathbf{h}) \\ &+ \frac{\mu}{2}||\hat{\mathbf{g}} - \sqrt{D}(\mathbf{F}\mathbf{P}^\top \otimes \mathbf{I}_K)\mathbf{h}||_2^2 \end{aligned} \qquad (5.11)$$

where μ is the penalty factor and $\hat{\boldsymbol{\zeta}} = [\hat{\boldsymbol{\zeta}}^{(1)\top}, \ldots, \hat{\boldsymbol{\zeta}}^{(K)\top}]^\top$ is the $KT \times 1$ Lagrangian vector in the Fourier domain. Equation 5.11 can be solved using ADMM by iteratively solving two subproblems g and h with closed-from solutions described as below.

Subproblem g

$$\begin{aligned}\hat{\mathbf{g}}^* &= \arg\min \mathcal{L}(\hat{\mathbf{g}}; \hat{\mathbf{h}}, \hat{\boldsymbol{\zeta}}) \qquad (5.12) \\ &= (\sum_{i=1}^{N}\hat{\mathbf{X}}_i^\top\hat{\mathbf{X}}_i + \mu\mathbf{I})^{-1}(\sum_{i=1}^{N}\hat{\mathbf{X}}_i^\top\hat{\mathbf{y}}_i + \mu\hat{\mathbf{h}} - \hat{\boldsymbol{\zeta}})\end{aligned}$$

where $\hat{\mathbf{h}} = \sqrt{D}(\mathbf{F}\mathbf{P}^\top \otimes \mathbf{I}_K)\mathbf{h}$. Fortunately, the kronecker product with the identity matrix can be broken into K independent IFFT computations of $\hat{\mathbf{h}}^{(k)} = \sqrt{D}\mathbf{F}\mathbf{P}^\top\mathbf{h}^{(k)}$. In practice, each $\hat{\mathbf{h}}^{(k)}$ can be estimated extremely efficiently by applying a FFT to each $\mathbf{h}^{(k)}$ padded with zeros implied by the $\mathbf{P}^\top$ masking matrix. The over-complete vector $\mathbf{h}$ can be easily obtained by concatenating $\{\mathbf{h}^{(k)}\}_{k=1}^{K}$.

Subproblem h

$$\begin{aligned}\mathbf{h}^* &= \arg\min \mathcal{L}(\mathbf{h}; \mathbf{g}, \boldsymbol{\zeta}) \qquad (5.13) \\ &= (\mu + \frac{\lambda}{\sqrt{D}})^{-1}(\mu\mathbf{g} + \boldsymbol{\zeta})\end{aligned}$$

where $\mathbf{g} = \frac{1}{\sqrt{D}}(\mathbf{P}\mathbf{F}^\top \otimes \mathbf{I}_K)\hat{\mathbf{g}}$ and $\boldsymbol{\zeta} = \frac{1}{\sqrt{D}}(\mathbf{P}\mathbf{F}^\top \otimes \mathbf{I}_K)\hat{\boldsymbol{\zeta}}$. Similarly, due to separability of the kronecker product with the identity matrix $\mathbf{I}_(K)$ both $\mathbf{g}$ and $\boldsymbol{\zeta}$ can be estimated extremely efficiently by applying an inverse FFT on each $\hat{\mathbf{g}}^{(k)}$ and $\hat{\boldsymbol{\zeta}}^{(k)}$ and then applying the lookup table implied by the masking matrix $\mathbf{P}$.

Lagrange Multiplier Update

$$\hat{\boldsymbol{\zeta}}^{(i+1)} \leftarrow \hat{\boldsymbol{\zeta}}^{(i)} + \mu(\hat{\mathbf{g}}^{(i+1)} - \hat{\mathbf{h}}^{(i+1)}) \tag{5.14}$$

where $\hat{\mathbf{g}}^{(i+1)}$ and $\hat{\mathbf{h}}^{(i+1)}$ are the current solutions to the above subproblems at iteration $i+1$ within the iterative ADMM.

5.2.1 Computational and Memory Analysis

Inspecting Equations 5.12, 5.13 and 5.14 one realizes that the dominant cost over the ADMM optimization is $\mathcal{O}(C(T^3K^3 + NT^2K^2))$, where C is the number of iterations and $\mathcal{O}(T^3K^3 + NT^2K^2)$ is the amount of computations required to solve a $KT \times KT$ linear system in subproblem g. As explained earlier, $\hat{\mathbf{X}}$ is sparse banded and through the variable reordering proposed in Equation 3.14 we can efficiently compute $\hat{\mathbf{g}}$ in a substantially smaller computational expense of $\mathcal{O}(C(TK^3 + NTK^2))$ by solving T independent $K \times K$ linear systems per iteration. In this case, the memory usage is $\mathcal{O}(K^2T)$ which is constant and independent of the number of training images and iterations. In addition, an overall computational cost of $\mathcal{O}((N + C)T \log T)$ is needed to compute the FFTs/IFFTs of (i) N vectorized training images of length T before the iterations, and (ii) the $\boldsymbol{\zeta}$, $\mathbf{g}$ and $\mathbf{h}$ variables during C iterations.

5.3 Experiments

5.3.1 Localization Performance

In the first experiment, we evaluated our method on the problem of eye localization, comparing with prior correlation filters (without limited boundaries), e.g. OTF [Refregier, 1991], MACE [Mahalanobis et al., 1987], UMACE [Savvides and Kumar, 2003], ASEF [Bolme et al., 2009], and MOSSE [Bolme et al., 2010]. For fair comparison, we trained a limited correlation filter using normalized intensity images (single-channel). The goal of this experiment is to demonstrate the superiority of correlation filters with limited boundaries against traditional correlation filter techniques.

Dataset: The CMU Multi-PIE face database [1] was used for this experiment, containing 900 frontal faces with neutral expression and normal illumination. We randomly selected 400 of these images for training and the reminder for testing.

Image Preprocessing: All images were cropped to have a same size of 128 × 128 such that the left and right eye are respectively centered at (40,32) and (40,96). The cropped images were power normalized to have a zero-mean and unit standard deviation. Then, a 2D cosine window was employed to reduce the frequency effects caused by opposite borders of the images in the Fourier domain.

Filters Training: We trained a 64 × 64 filter of the right eye using full face images for our method ($T = 128 \times 128$ and $D = 64 \times 64$), and 64 × 64 cropped patches (centered upon the right eye) for the others. Similar to ASEF and MOSSE, we defined the desired response as a 2D Gaussian function with a spatial variance of $s = 2$ whose the peak was located upon the center of the right eye.

Localization and Evaluation: Eye localization was performed by correlating the filters over the face testing images followed by selecting the peak of the output as the predicted eye location. The eye localization was evaluated by the distance between the predicted and desired eye locations normalized by inter-ocular distance,

$$d = \frac{\|\mathbf{p}_r - \mathbf{m}_r\|_2}{\|\mathbf{m}_l - \mathbf{m}_r\|_2} \tag{5.15}$$

where $\mathbf{m}_r$ and $\mathbf{m}_l$ respectively indicate the true coordinates of the right and left eye, and $\mathbf{p}_r$ is the predicted location of the right eye. A localization with normalized distance $d < th$ was considered as a successful localization. The threshold th was set to a fraction of inter-ocular distance.

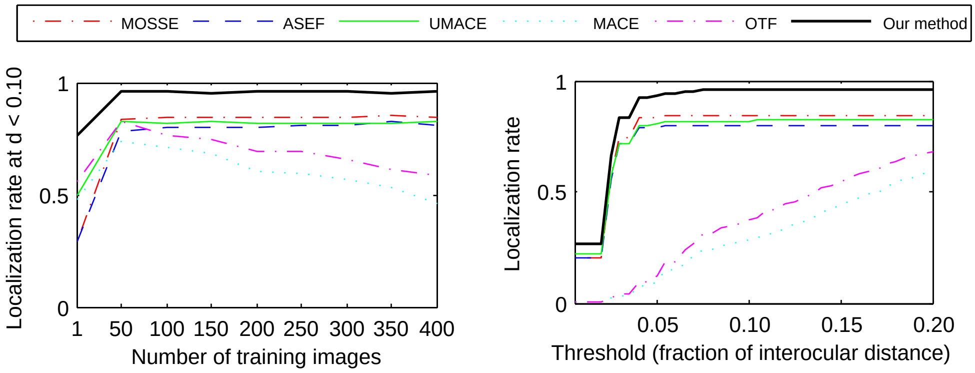
[1] http://www.multipie.org/

The average of evaluation results across 10 random runs (at each run 400 images are randomly selected for training and the others for testing) are depicted in Figure 5.2, where our method outperforms the other correlation filters across all thresholds and numbers of training examples. The accuracy of OTF and MACE declines by increasing the number of training images due to over-fitting. During the experiment, we observed that the low performance of the UMACE, ASEF and MOSSE was mainly caused by wrong localizations of the left eye and the nose. This was not the case for our method, as our filter was trained in a way that returned zero correlation values when centered upon non-target patches of the face image. A visual depiction of the filters and their outputs can be seen in Figure 5.3, illustrating examples of wrong and correct localizations. The high Peak-to-Sidelobe Ratio (PSR) [Bolme et al., 2010] value of our method (15.7) indicates that our approach produces sharper peaks compared to the other techniques (9.3 for MOSSE, for example) which makes detection easier and more reliable.

The Influence of D and T on Detection Accuracy: We examined the influence of T (the size of training images) on the performance of eye localization. For this purpose, we employed cropped patches of the right eye with varying sizes of $T = \{D, 1.5D, 2D, 2.5D, 3D, 3.5D, 4D\}$ to train filters of size $D = 32 \times 32$. The localization results are illustrated in Figure 5.4(a), showing that the lowest performance is obtained when T is equal to D (32×32) and the localization rate improved by increasing the size of the training patches with respect to the filter size. The reason is that (1) by choosing $T > D$ the portion of patches not affected by boundary effects ($\frac{T-D+1}{T}$) reduces, and (2) when $T > D$ more non-target (negative) patches are involved in the learning process, making the detector robust against wrong detection.

5.3.2 Runtime and Convergence Evaluation

This experiment demonstrates the utility of our approach to other iterative methods. Specifically, we compared our proposed approach against other methods in



(a)

(b)

Figure 5.2: Eye localization performance as a function of (a) number of training images, and (b) localization thresholds. Best viewed in color.

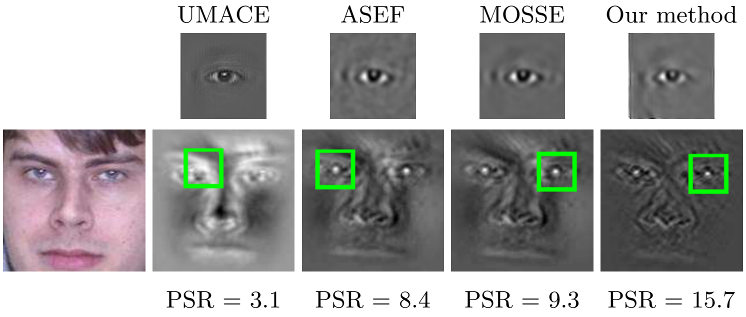


Figure 5.3: An example of eye localization is shown for an image with normal lighting. The outputs (bottom) are produced using 64 × 64 correlation filters (top). The green box represents the approximated location of the right eye (output peak). The peak strength measured by PSR shows the sharpness of the output peak.

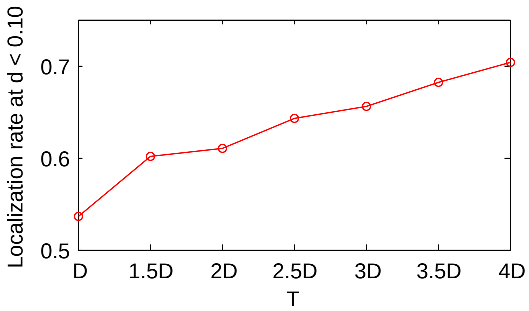
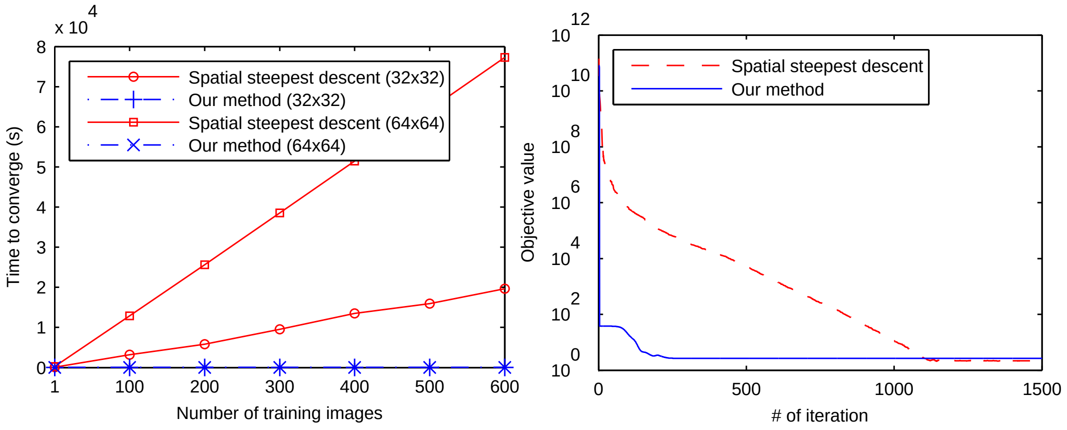


Figure 5.4: The localization rate obtained by different sizes of training images (T), the size of the trained filter is $D = 32 \times 32$.



(a) (b)

Figure 5.5: Runtime performance and convergence behavior of our method against another naive iterative method (steepest descent method) [Zeiler et al., 2010]. Our approach enjoys superior performance in terms of: (a) convergence speed to train two filters with different sizes (32 × 32 and 64 × 64) and (b) the number of iterations required to converge.

literature for learning filters efficiently using iterative methods. We compare our convergence performance with a steepest descent method [Zeiler et al., 2010] for optimizing the same objective. Results can be seen in Figure 5.5: (a) represents time to converge as a function of the number of training images, and (b) represents the number of iterations required to optimize the objective (in Equation 5.2).

In (a) one notices impressively how convergence performance is largely independent of the number of images used during training. This can largely be attributed to the pre-computation of the auto- and cross-spectral energy vectors. As a result, iterations of the ADMM do not need to re-touch the training set, allowing our proposed approach to dramatically outperform more naive iterative approaches. Similarly, in (b) one also notices how relatively few iterations are required to achieve good convergence.

5.3.3 Visual Object Tracking

We evaluated the proposed method for the task of real-time tracking on a sequence of commonly used test videos [Ross et al., 2008] shown in Table 5.1. We compared our approach with state-of-the-art trackers including MOSSE [Bolme et al., 2010], kernel-MOSSE [Henriques et al., 2012], MILTrack [Babenko et al., 2011], STRUCK [Hare et al., 2011], OAB [Grabner et al., 2006], SemiBoost [Grabner et al., 2008], FragTrack [Adam et al., 2006] and IVT [Ross et al., 2008].

All of these methods were tuned by the parameter settings proposed in their reference papers. The desired response for a $m \times n$ target was defined as a 2D Gaussian with a variance of $\mathbf{s} = \sqrt{mn}/16$ following [Henriques et al., 2012]. The regularization parameter λ was set to 10^{-2}. The number of ADMM iterations for optimization was four as a trade-off between precision and tracking speed. A track initialization process was employed for our approach and MOSSE, where for fair comparison eight random affine perturbations were used to initialize the first filter following Bolme et al. [2010].

5.3.3.1 Implicit Dense Sampling

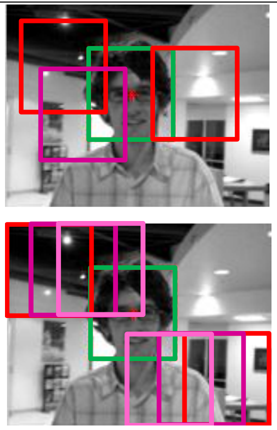
Almost all current tracking approaches perform online learning and detection using a random sampling strategy [Babenko et al., 2011], [Grabner et al., 2008],[Adam et al., 2006], [Hare et al., 2011]. This class of trackers typically select several non-target examples from the target's neighborhood for the learning algorithm to make the tracker robust against wrong detections, as shown in Figure 5.6(top).

There are three major drawbacks of random sampling trackers. First, the sparse samples are stemmed from a small and limited region around the target of interest with a large amount of overlap and pattern redundancy. This can drastically reduce the discrimination of the tracker to unseen patches and lead to over-training. Second, the spatial information of the randomly selected samples is ignored in random sampling. Finally, collecting the random samples from small neighborhood of the target does not allow the tracker to fully exploit all possible non-target (background) samples over the entire of the frame during online learning and detection. For example, a video frame of size 100 × 100 contains almost 10^4 background samples which can efficiently be employed for a robust tracker learning and adaption. Whereas, a random sampling strategy can only make use of just 10-15 of these patches at each frame to save computation and memory.

On the other hand, one may say that all of the problems associated with random sampling can be addressed by dense sampling strategy, where all possible (or at least a large amount) of samples stemmed from the entire frame can be used for online learning and tracking. This solution seems helpful at first glance, but it negates the computational complexity and memory usage of learning algorithms (SVM, Struct [Hare et al., 2011], Boost [Babenko et al., 2011] [Grabner et al., 2008]) and, as a result, drastically reduces the tractability and the speed of the tracker. For clarification, simply consider the amount of computation and memory required to learn/update a linear SVM using 10^4 samples of size 60×60 per frame.

Fortunately, the proposed approach exhibits a desired characteristic that makes it very useful for dense sampling learning and tracking. A stated earlier, our approach is capable of learning a filter with small spatial support from training examples with much larger size. Referring to Figure 5.1, suppose that the problem is tracking the 50×50 face object over frame sequences of size 200×200. For this purpose, our approach can be applied to train a small filter of size $D = 50 \times 50$ that returns a desired correlation output of size $T = 200 \times 200$ when correlated over a frame of size $T = 200 \times 200$. The global maximum over each frame can be used to localize and track the face object over video sequences.

According the Figure 5.1(d), our method implicitly exploits all possible patches stemmed from different parts of the image/frame during learning including the target (positive) and background (negative) patches. This is what exactly dense sampling strategies try to do. As explained in the Figure 5.1 caption, a very small portion of these patches are affected by circular shift (synthetic) while the rest exactly show the real world background patches which can be efficiently used to learn a robust and discriminative tracking filter. The main differences between our method and the random sampling tracker are clearly illustrated in



Memory. $\mathcal{O}(ND)$
Complexity. learning algorithms (SVM, Boost, etc)
speed. 10 - 15 *fps*

Memory. $\mathcal{O}(D)$
Complexity. FFT and ADMM iterations
speed. 50 - 100 *fps*

Figure 5.6: The main differences between regular random sampling trackers (top) and our approach with implicit dense sampling strategy (bottom). **The required memory** for random sampling trackers is linear to the number of selected training samples, N, while, for our approach is independent of the number of training samples. The **complexity** of random sampling trackers is determined by the type of the learning algorithm used for learning/detection, e.g. Boosting and SVM. Whereas, all our method needs is computing a FFT of the frame and performing few ADMM iterations. Due to the very low computational cost of our method, its **tracking speed** is remarkably faster (in terms of *fps*) compared to the random sampling approaches. D is the length of signal (size of frame)

Figure 5.6.

In general, the proposed method is very efficient for dense sampling due to the following reasons:

- Our method exploits all possible patches implicitly collected from the entire frame with no additional memory usage.
- The computation of our method for implicit dense sampling is limited to the FFT of the frame and a few iterations of ADMM optimization.
- This dense sampling is embedded in an online adaption, subsection 5.3.3.2, which make our method much more robust against wrong detection and challenging circumstances.

5.3.3.2 Online Adaption

We borrowed the online adaption from the work of Bolme et al. [2010] to adapt our filter at the i^{th} frame by updating the auto-spectral and cross-spectral ener-

Sequence	Frames	Main Challenges
Faceocc1	886	Moving camera, occlusion
Faceocc2	812	Appearance change, occlusion
Girl	502	Moving camera, scale change
Sylv	1344	Illumination and pose change
Tiger1	354	Fast motion, pose change
David	462	Moving camera, illumination change
Cliffbar	472	Scale change, motion blur
Coke Can	292	Illumination change, occlusion
Dollar	327	Similar object, appearance change
Twinings	472	Scale and pose change

Table 5.1: Video sequences used for tracking evaluation.

gies in Equation 5.6:

$$\begin{aligned}(\hat{\mathbf{s}}_{xx})^{i} &= \eta(\hat{\mathbf{x}}_i \circ \mathrm{conj}(\hat{\mathbf{x}}_i)) + (1-\eta)(\hat{\mathbf{s}}_{xx})^{i-1} \\ (\hat{\mathbf{s}}_{xy})^{i} &= \eta(\hat{\mathbf{y}}_i \circ \mathrm{conj}(\hat{\mathbf{x}}_i)) + (1-\eta)(\hat{\mathbf{s}}_{xy})^{i-1} \qquad (5.16)\end{aligned}$$

where, η is the adaption rate. We practically found that $\eta = 0.025$ is appropriate for our method to quickly be adapted against pose, scale, illumination variations, etc.

5.3.3.3 Quantitative and Qualitative Results

The tracking results are evaluated in Table 5.2, as (i) percentage of frames where the predicted position is within 20 pixels of the ground truth (precision) [Babenko et al., 2011] [Hare et al., 2011], (ii) average localization error in pixels, and (iii) tracking speed as number of frame per second (*fps*). Our method averagely achieved maximum precision and minimum localization errors, followed by STRUCK. One explanation for this is that our limited boundaries approach employs a rich set of training samples containing all possible positive (target) and negative (non-target of the whole frame) patches to train the correlation filter, dense sampling. Whilst, the non filter approaches such as STRUCK and MILTrack are limited by learning a small subset of positive and negative patches randomly stemmed from the target's neighborhoods.

Similarly, it can be explained that the accuracy of MOSSE and kernel-MOSSE are affected by synthetic negative samples which are not representative of the "real-world" examples, as illustrated in Figure 5.1(c). In addition, these techniques train an object filter/template using a cropped patch of the target subject and discard the non-target (background) part of the whole frame. This reduces the robustness of these methods against cluttered background. Moreover, our method enjoys a high stability against challenging variations in scale (*Cliffbar* and *Twinings*), illumination (*Sylv*), pose (*David*), appearance (*Girl*) and partial occlusion (*Faceocc1* and *Faceocc2*) due to the online adaption. In case of the tracking speed, MOSSE outperformed the other methods by 600 *fps*. Our method obtained lower *fps* compared to MOSSE and kernel-MOSSE, due to its iterative manner. However, it obtained a tracking speed of 50 *fps* which is appropriate for real-time tracking. Our method can run at 100 *fps* using two ADMM iterations with superior average position error and precision compared to MOSSE and kernel-MOSSE. The position error as a function of iteration number is shown in Figure5.7.

A visual depiction of tracking results for some selected videos is shown in Figure 5.8, where our method achieved higher precision over all videos except *Tiger1* and *Twinings*. Moreover, Figure 5.9 shows that our approach suffers from less drift over the test videos. In Figure 5.10, tracking results of some selected frames of the tested videos are shown, including ground truth, objects of interest, trained filter per frame, tracking result and the estimated correlation outputs.

5.3.3.4 Parameter Selection

Number of ADMM Iterations. We evaluated our method with different iterations of $\{1, 2, 4, 8, 16, 32, 64\}$, as shown in Figure 5.7, and eventually selected four iterations (as a tradeoff between precision and tracking speed) for our tracker. The tracking speed of our method using 4 iterations is around 50 *fps* with an average position error of 8 pixels as reported in Table 5.2. The

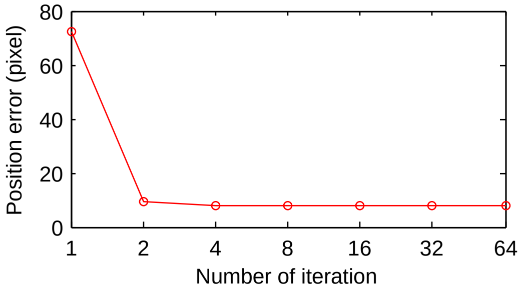


Figure 5.7: The position error of tracking versus the number of ADMM iterations. We selected 4 iterations as a tradeoff between tracking performance and computation.

average position error of 2 iterations is around 9.5 pixels with tracking speed of 100 *fps*. This allows one to choose the number of ADMM iterations respect to the desired tracking speed and performance.

Initial Filter Learning. Following [Bolme et al., 2010], we randomly generated eight affine perturbations of the first frame to initially train the target filter. For this, the first frame was randomly perturbed by rotating by up to $\pm\frac{\pi}{36}$, scaling by up to 1.0 ± 0.1 and translating by up to ± 4.0 pixels. Eight random perturbations were selected for fair comparison to the MOSSE technique.

Adaption Rate η. The adaption rate for auto- cross- correlation adaption was empirically set to $\eta = 0.025$, Equation 5.16. We evaluated different values of η form 0.001 to 0.500 and eventually selected $\eta = 0.025$ with less average tracking error over all testing videos. This rate for the MOSSE and kernel-MOOSE was selected as 0.05 and 0.075, respectively.

Regularization Parameter λ. We evaluated the proposed approach by adjusting λ form 0.0001 to 1.0. Similar to MOSSE [Bolme et al., 2010], we trained an initial filter using the first frame of each video and evaluated its PSR on the second frame. We realized that $0.001 \leqslant \lambda \leqslant 0.1$ produced high PSRs for all videos and eventually selected $\lambda = 0.01$ for tracking. We used the same value of λ for all experiments in this thesis.

	MOSSE	**KMOSSE**	**MILTrack**	**STRUCK**	**OAB(1)**	**SemiBoost**	**FragTrack**	**Our method**
FaceOcc1	**1.00** 7	**1.00** **5**	0.75 17	0.97 8	0.22 43	0.97 7	0.94 7	**1.00** 8
FaceOcc2	0.74 13	0.95 8	0.42 31	0.93 **7**	0.61 21	0.60 23	0.59 27	**0.97** **7**
Girl	0.82 14	0.44 35	0.37 29	**0.94** **10**	- -	- -	0.53 27	0.90 12
Sylv	0.87 7	**1.00** 6	0.96 8	0.95 9	0.64 25	0.69 16	0.74 25	**1.00** **4**
Tiger1	0.61 25	0.62 25	0.94 9	**0.95** **9**	0.48 35	0.44 42	0.36 39	0.79 18
David	0.56 14	0.50 16	0.54 18	0.93 9	0.16 49	0.46 39	0.28 72	**1.00** **7**
Cliffbar	0.88 8	0.97 6	0.85 12	0.44 46	0.76 -	- -	0.22 39	**1.00** **5**
Coke Can	0.96 **7**	**1.00** **7**	0.58 17	0.97 **7**	0.45 25	0.78 13	0.15 66	0.97 **7**
Dollar	**1.00** **4**	**1.00** **4**	**1.00** 7	**1.00** 13	0.67 25	0.37 67	0.40 55	**1.00** 6
Twinings	0.48 16	0.89 11	0.76 15	**0.99** **7**	0.74 -	- -	0.82 14	**0.99** 9
mean	0.80 11	0.84 12	0.72 16	0.91 12	0.53 31	0.62 29	0.51 37	**0.97** **8**
fps	**600**	100	25	11	25	25	2	50

Table 5.2: The tracking performance is shown as a tuple of {*precision within 20 pixels*, *average position error in pixels*}, where our method achieved the best performance over 8 of 10 videos. The best *fps* was obtained by MOSSE. Our method obtained a real-time tracking speed of 50 *fps* using four iterations of ADMM. The best result for each video is highlighted in bold.

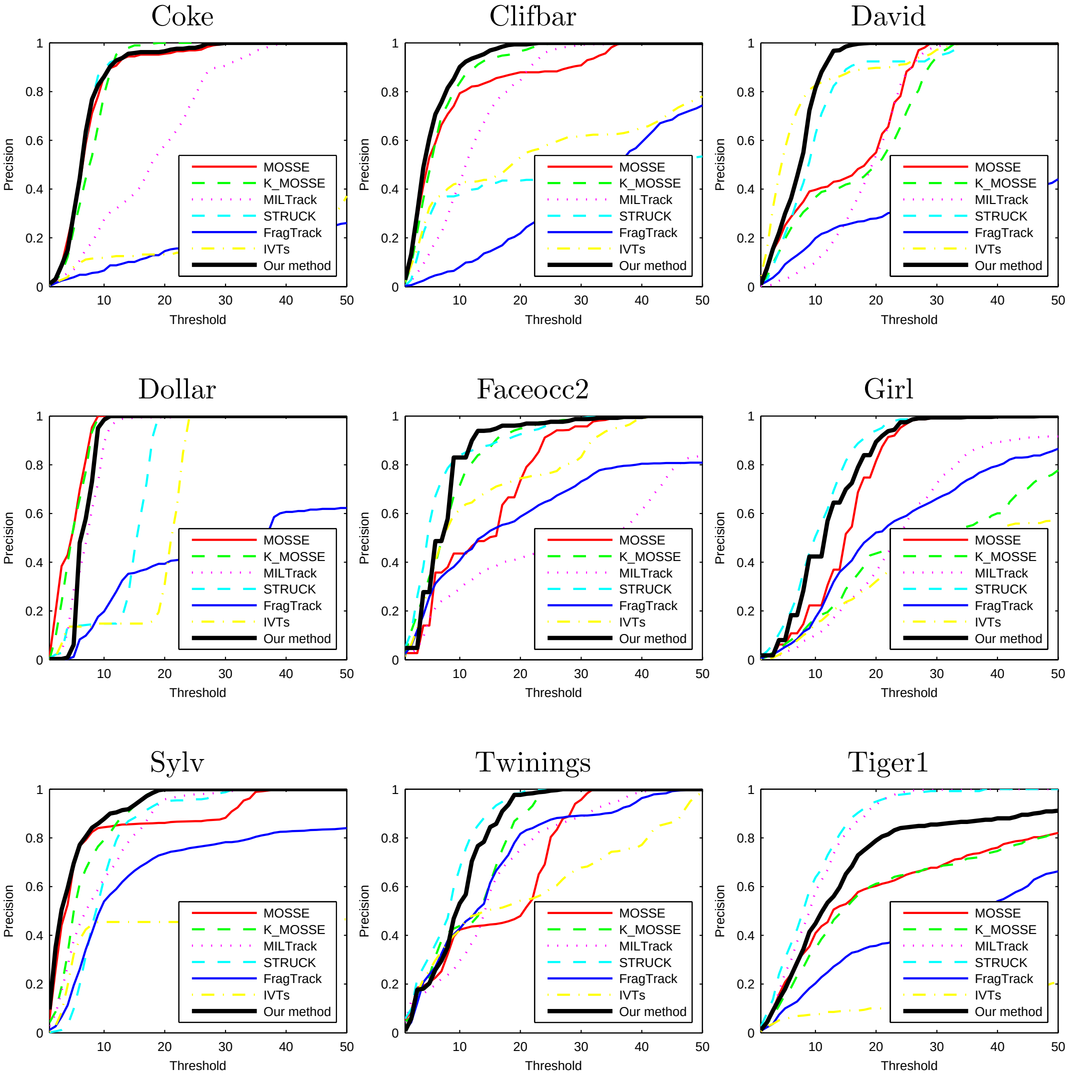


Figure 5.8: Tracking results for selected videos, precisions versus the thresholds.

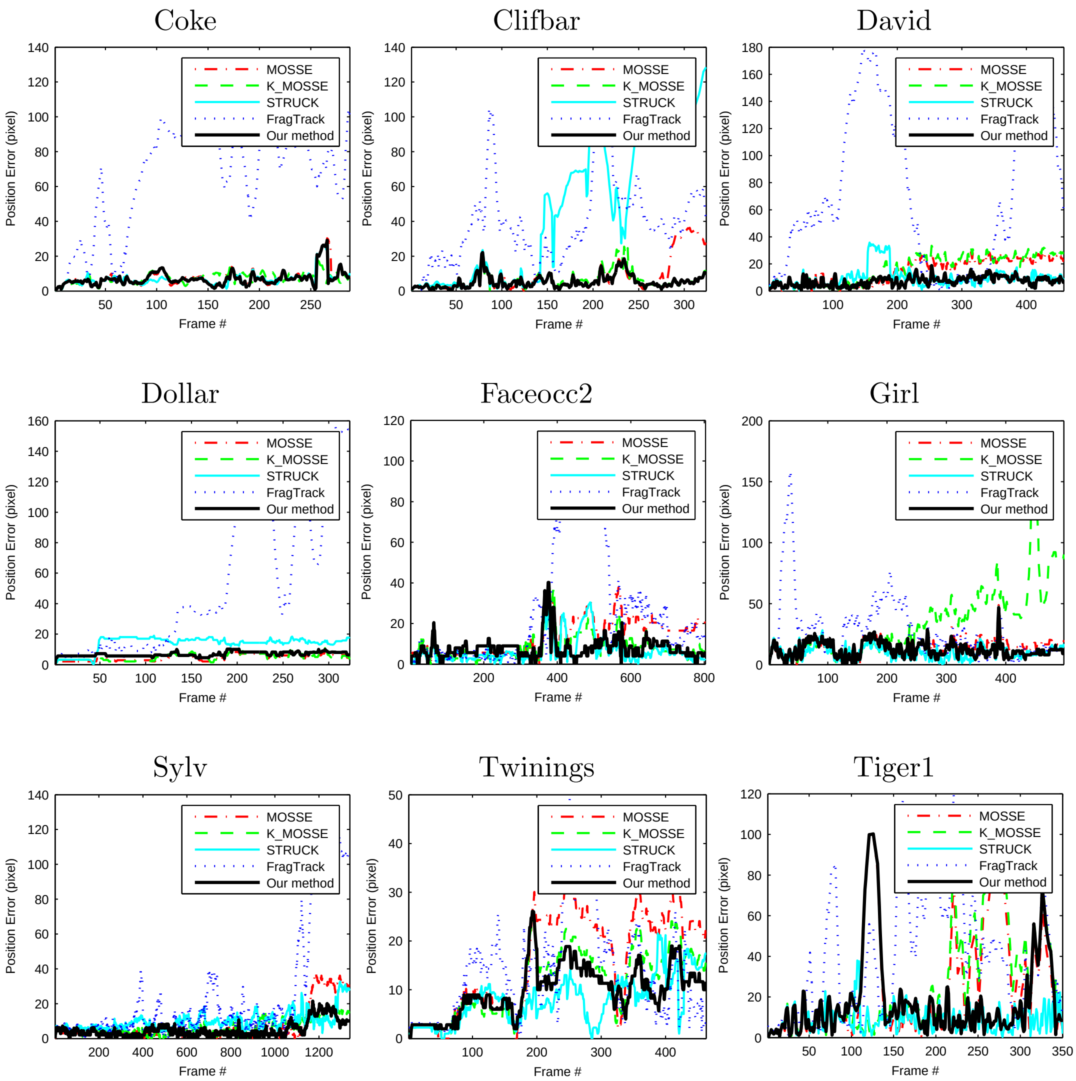


Figure 5.9: Tracking results for selected videos, position error per frame.

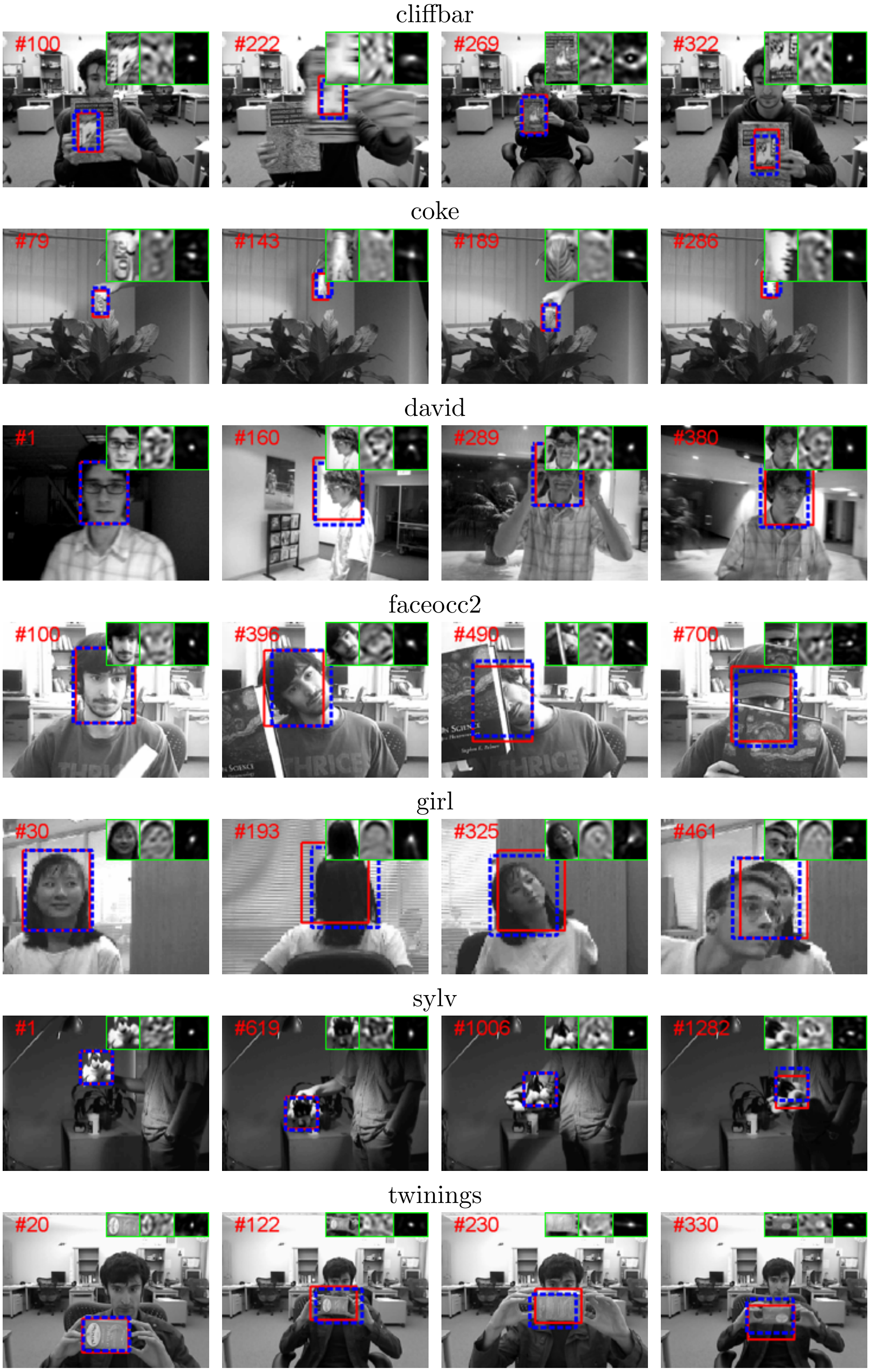


Figure 5.10: Tracking results of our method over testing videos with challenging variations of pose, scale, illumination and partial occlusion. The blue (dashed) and red boxes respectively represent the ground truth and the positions predicted by our method. For each frame, we illustrate the target, trained filter and correlation output.

5.4 Chapter Summary

In this chapter, we investigated the influence of boundary effects on correlation filter estimation. We theoretically demonstrated that current techniques learn correlation filters using an unbalanced training set of $D-1$ shifted and one non-shifted training patches, where D is the signal length. These shifted patches are implicitly created through the circular property of the correlation/convolution operator. We explained that the number of patches affected by boundary effects can be drastically reduced by learning correlation filters whose spatial support is much smaller than the dimension of training images. For this purpose, we proposed a new objective to learn correlation filters with limited spatial support. Particularly, we demonstrated how this objective can be iteratively optimized in the frequency domain through an augmented lagrangian method with very efficient memory and computation expenses. Moreover, we proposed an extension of this objective to handle multi-channel features for learning multi-channel correlation filters with limited boundaries. We demonstrated the superior performance of the proposed approach for a localization task against the state of the art correlation filters. In addition, the convergence performance of our method was compared with a spatial steepest descent method for optimizing the same objective function, where we empirically showed that our approach converges in a few iterations and its convergence time is almost independent of filter size and the number of training images. Our method outperformed the state of the art trackers for the task of visual object tracking.

Chapter 6

Cascaded Correlation Filters for Facial Landmark Detection

Facial landmark detection is very important for face analysis, and has been studied widely in recent years [Belhumeur et al., 2011; Yang and Patras, 2013a,b; Dantone et al., 2012; Valstar et al., 2010; Zhou et al., 2013; Milborrow and Nicolls, 2008]. The main challenges of this problem come from imaging conditions, when face images are captured under uncontrolled circumstances such as extreme pose, illumination, expression and partial occlusions. To deal with these challenges, many approaches have been proposed which can be generally categorized in two classes: global face shape and local features based approaches.

The Active Appearance Model [Cootes et al., 2001], Active Shape Model [Cootes et al., 1995] and their variations [Belhumeur et al., 2011; Milborrow and Nicolls, 2008; Saragih et al., 2011; Zhao et al., 2013] are typical methods which exploit face global shape and geometric information for facial point detection. Although these approaches are able to handle wrong detections caused by visually similar landmarks, they are not robust against large variations in the face pose and expression. Moreover, their optimization strategies (e.g. gradient descent) are very sensitive to the initialization.

The local feature based methods, on the other hand, basically try to learn individual local detectors for each facial landmark. In [Vukadinovic and Pantic, 2005], 20 independent GentleBoost detectors are trained to detect 20 facial landmarks in face images. Although landmark localization is performed within limited search regions, this approach suffered from wrong detections caused by the lack of global face information. The approach proposed in [Zhao et al., 2012] is another technique that integrates context constraints with local texture descriptors in the cascaded AdaBoost framework to independently detect facial landmarks. Again, because of using just local texture information this method does not show robustness to uncontrolled environments.

In recent years, several regression-based approaches have been proposed to directly map the local or global face appearance to facial landmarks [Dantone et al., 2012; Valstar et al., 2010; Yang and Patras, 2013a,b]. Valstar et al. [2010] proposed to learn regressors that map local image patches into the individual landmarks. The work in Dantone et al. [2012] tried to implicitly map the global face appearance into an average shape configuration using a conditional regression forest. But as stated in [Saragih, 2011], it is challenging to directly learn an ideal regressor to map face appearance into global face shape with complex non-linear variations.

6.1 Proposed Cascaded Framework

A cascaded correlation filters framework is proposed for facial landmark detection/localization in face images. In this framework, a set of correlation filters with different spatial support (size) are connected together in a cascade manner, where the size of the correlation filter decreases at lower levels. At the first level, a global correlation filter which trained over the whole face image is applied to predict its corresponding facial landmark. Since this correlation filter is trained over an entire face, it implicitly encodes the geometric constrains and shape information of facial landmarks. Exploiting global appearance and geometric face information can significantly reduce the risk of wrong detections caused by visually similar landmarks, e.g. the right eye center and the left eye center, and approximately *predict (not detect/localize)* the position/region of the landmark of interest, even in images with challenging pose, expression and occlusion.

The prediction in the first level, however, suffers from inaccurate detection typically with small position error, especially in those landmarks which might be affected by extreme pose and expression, e.g. outer lower and upper lip points. To deal with this problems, the cascaded framework contains smaller correlation filters with limited boundaries at the next levels to locally refine the prediction of the previous levels. To avoid wrong detections and ambiguities of the small size correlation filters, the search regions at each level are limited to a small region around the predicted location of the previous level.

A three-level cascaded correlation filter is shown in Figure 6.1. The input for all the three levels are the whole face returned by a face detector. The filter size at level 1, 2 and 3 are respectively 128 × 128, 64 × 64 and 32 × 32 pixels. Given an input face image of size 128 × 128, the location of the landmark of interest is predicted at level 1. For this, a 128 × 128 correlation filter is correlated over the entire face and the global maximum peak is selected as the initial prediction of each target landmark, the blue filled circle on the face image in level 1. The red box indicates the search region which is the entire image at level 1. At level 2, a smaller 64 × 64 correlation filter is similarly correlated on the entire face and the maximum peak of the search region is selected as the refined landmark location. Note that the search region in level 2 is limited to a small region around the initial prediction of the previous level, level 1. In a similar manner, the prediction in level 2 is used to refine the landmark position in level 3. The final location of the landmark of interest is detected by averaging all the three predictions over the cascade (the yellow filled circle).

6.2 Experimental Results

We evaluated our cascaded correlation filter technique on two publicly available datasets, BioID [Jesorsky et al., 2001] and LFPW [Belhumeur et al., 2011].

6.2.1 Datasets

LFPW (Labeled Face Parts in the Wild) dataset has been used widely in facial landmark detection experiments. All face images in this dataset are downloaded

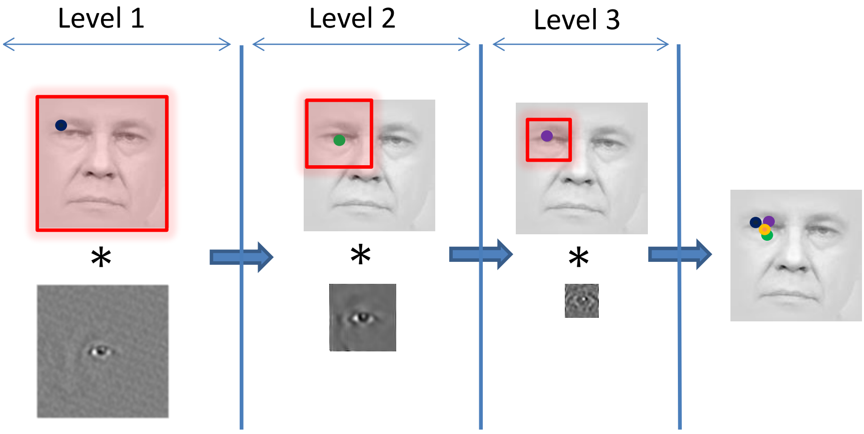


Figure 6.1: Framework of the cascaded correlation filters for facial landmark detection. At the first level an inaccurate prediction is performed using global landmark filter trained using whole face images. The prediction with small position error is refined downward to the cascade levels using correlation filters with smaller size (limited boundaries). The search area at each level is shown by the red square. The color filled circles show the predicted landmark of the interest in each level. The final detection/localization is performed by averaging all the predicted locations over the all cascade levels, the yellow filed circle in the right most face image. Best viewed in color.

from the web and represent large variations in pose, illumination, expression and occlusion. The original version of this dataset contains 1100 training and 300 testing images. Since this dataset provides only web image URLs, some of image URLs are not currently available. Therefore, we only downloaded 714 training and 214 testing images. We used this dataset for parameter tuning, validation and comparison of our method against the stat-of-the-art.

BioID dataset consists of 1521 near frontal face images of 23 subjects captured with various scales and face expressions in lab environment, and is therefore less challenging compared to LFPW. We used 20 landmarks manually annotated in the FGNET project. This dataset has been commonly used to evaluate most of the previous methods for facial landmark detection, allowing us to compare our method to them. We used the training/testing split provided by [Yang and Patras, 2013a], where 1000 images are randomly selected from the dataset for training and the rest for testing.

6.2.2 Implementation Details

We first investigate the number of cascade levels and feature channels using the LFPW dataset. Then, we compare our technique against several state-of-the-art approaches on both the BioID and LFPW datasets.

Face Bounding Box. The Viola and Jones face detector [Viola and Jones, 2001] is applied to find the face bounding box and is enlarged by 20% in order to ensure that all facial landmarks are enclosed. All the boxes are resized to 128 × 128 pixels. We assumed that there is only one face in each image and all images are in gray scale.

Desired Correlation Outputs. A 2D Gaussian function with spatial variance of 2 is employed to define the desired correlation outputs whose the peak was located upon the center of the landmark of interest. Note that the dimension of correlation outputs is the same as the size of image (128 × 128).

Landmark Detection and Evaluation: Landmark prediction at each cascade level is performed by correlating the trained landmark filters over face image feature channels and then selecting the peak over the summation of the correlation outputs as the predicted landmark location. The detection at the last cascade level is evaluated by the distance between the predicted and ground truth landmark location normalized by inter-ocular distance,

$$d = \frac{\|\mathbf{p}_i - \mathbf{m}_i\|_2}{\|\mathbf{m}_l - \mathbf{m}_r\|_2} \tag{6.1}$$

where $\mathbf{m}_r$ and $\mathbf{m}_l$ respectively indicate the true coordinates of the right and left eye, and $\mathbf{p}_i$ and $\mathbf{g}_i$ are the predicted and ground truth location of the *ith* landmark, respectively. A localization with normalized distance $d < th$ was considered as a successful localization. The threshold *th* is set to a fraction of inter-ocular distance (0.10 in our experiments).

Feature Extraction. We extracted 43 feature channels for each face image, including 40 Gabor features (eight different orientations and five scales), two Sobel features (horizontal and vertical gradient magnitudes) and one power-normalized intensity image. A Cosine-window is applied on all the feature channels to reduce the frequency effects caused by opposite borders of the images in the Fourier domain. According to Figure 6.2, Soble features obtained the highest detection error followed by image intensities, due to their sensitivity to lighting and low discrimination power to model different parts of human face. Gabor features which have been widely used for various face problems, remarkably reduced the detection error. This shows the advantage of using multi-channel features (e.g. Gabor) over scalar features (intensity) for facial point detection. Eventually, we used a combination of all these features (40 Gabors features, two Soble features and the normalized intensity image) with the lowest detection error to improve the overall detection performance as proposed in [Yang and Patras, 2013a].

Number of Cascade Levels. We investigated the performance of our technique versus the number of cascade levels. For this, we trained five different multi-channel correlation filters of size 128 × 128, 64 × 64, 32 × 32, 16 × 16 and 8×8 using 128×128 LFPW training images. Then we employed these five filters to form five cascaded correlation filters with different number of levels, namely L0 (one level just whole face - 128×128), L1 (two levels - 128×128 and 64×64), L2, L3 and L4 (five levels, including all the correlation filters).

The performance of these cascaded filters for detecting 10 landmarks of the LFPW testing images are shown in 6.3. The lowest localization rate and highest mean error belong to L0 detector with a whole-face 128 × 128 multi-channel correlation filters. The reason is that holistic correlation filter is not robust against face expression and pose, specially for those face landmarks which are more affected by pose and expression, e.g. mouth landmarks. Reducing the size of the correlation filters down to the cascade increases the detection performance. For those landmarks which are not heavily involved in face expression, e.g. the eyes and nose, the improvement is marginal.

The correction of the initial prediction at the first level of the cascade downward to the last level is shown for several of the LFPW examples with partial

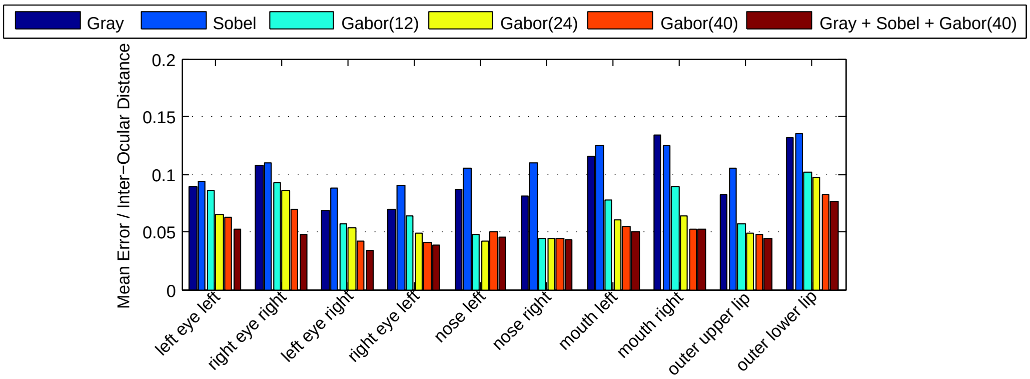


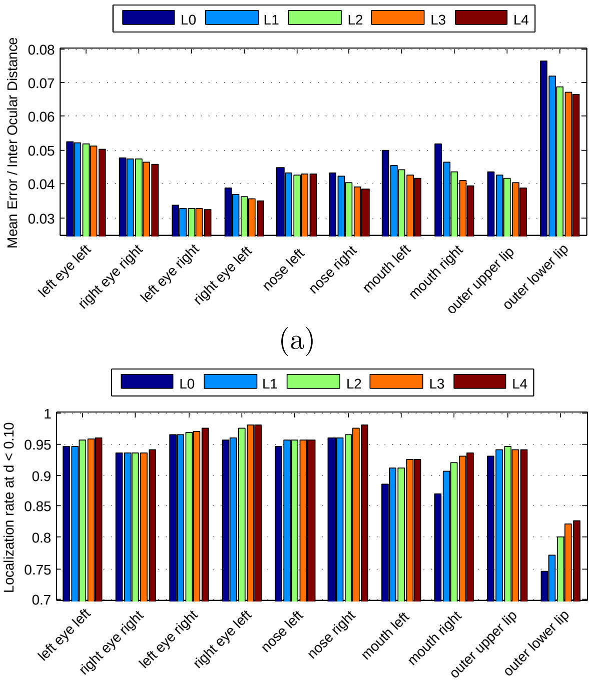
Figure 6.2: Exploring different types of feature channels for five-levels cascaded landmark detection using the LFPW dataset.

occlusions, pose variation and expression in Figure 6.4. The prediction at the first level is not very accurate, particularly for occluded and under expression landmarks, e.g. eyes and mouth corners. But, these inaccurate predications at the first level are improved through the cascade refining over the next levels. These small corrections over the cascade levels provides more accurate and robust landmark detection.

6.2.3 Comparison with The-state-of-the-art Landmark Detectors

We compared our proposed framework with the-start-of-the-art and leading landmark detectors in the literature on the LFPW and BioID datasets, including Structured-Output Regression Forests (SO-RF) [Yang and Patras, 2013a], Conditional Regression Forests (C-RF) [Dantone et al., 2012], Privileged Information-based Conditional Regression Forest (PI-CRF) [Yang and Patras, 2013b], [Belhumeur et al., 2011], Boosted Regression [Valstar et al., 2010], Exemplar-based Graph Matching [Zhou et al., 2013], Extended Active Shape Model [Milborrow and Nicolls, 2008].

Figure 6.5 illustrated the detection performance of our method compared to the others for the LFPW dataset. Our method is trained using 714 available training and tested on 214 available testing examples. For the other approaches, the accuracy is reported using 821-870 and 214 available training and testing exam-



(a)

(b)

Figure 6.3: Evaluating the proposed approach with different levels of cascade by (top) mean normalized localization error, and (bottom) localization accuracy at threshold of $d < 0.10$ on the LFPW dataset.

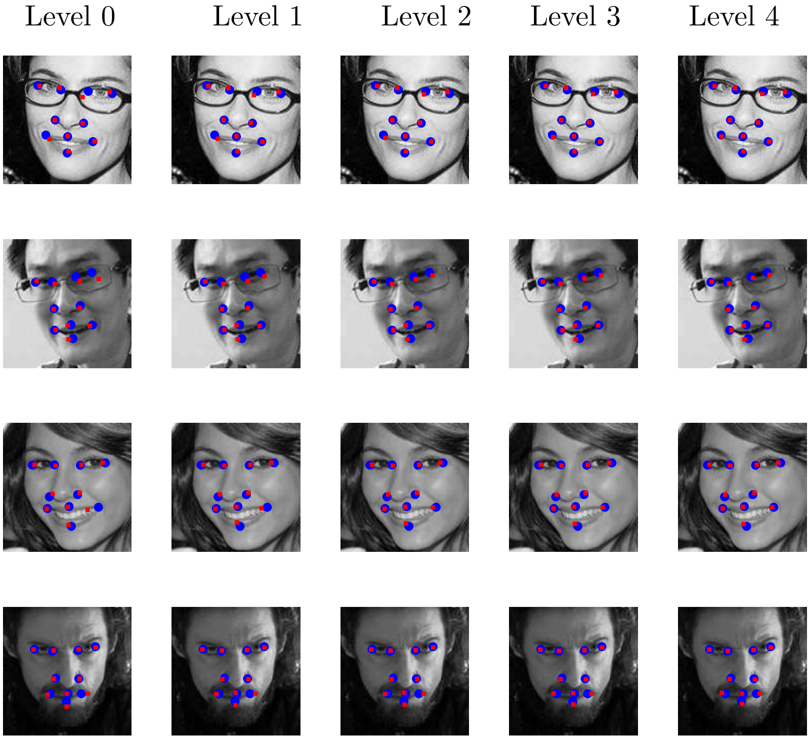
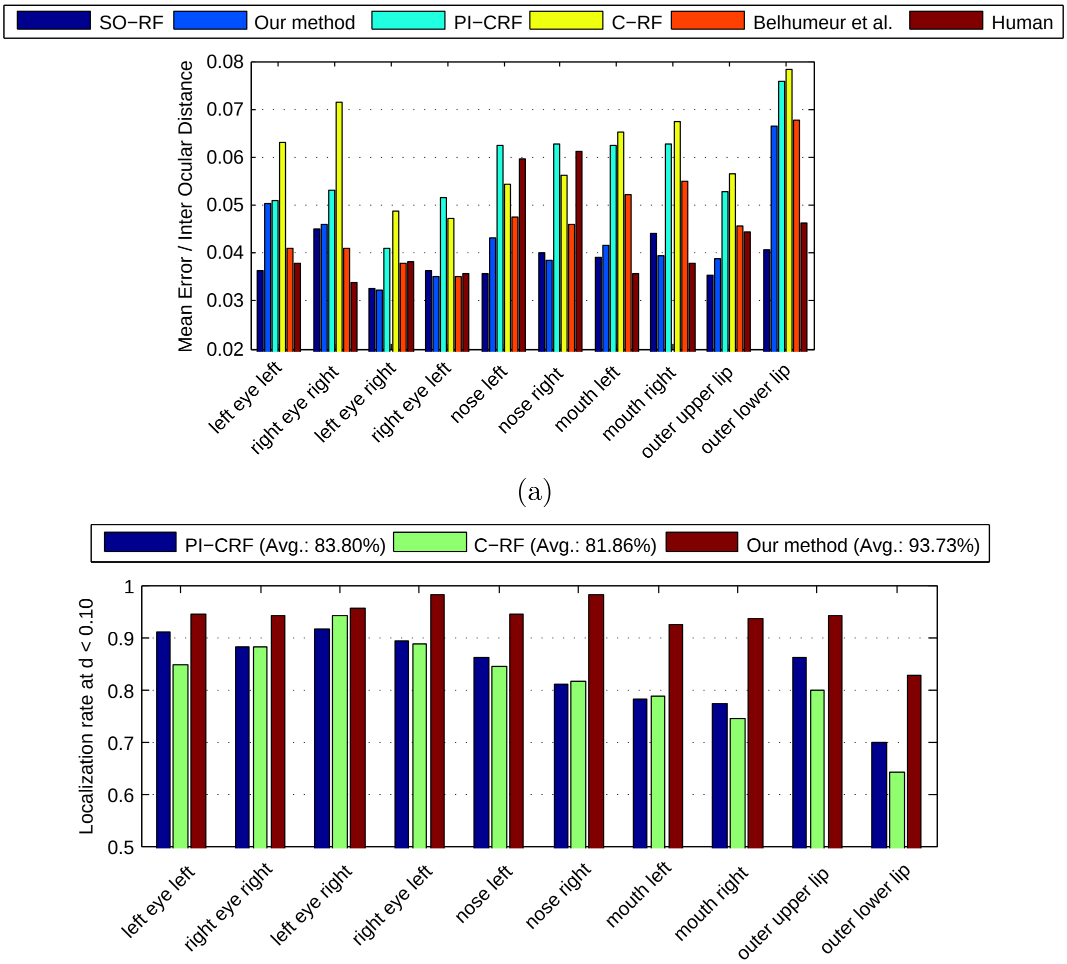


Figure 6.4: Accurate landmark detection over cascade levels. The ground truth and predicted locations are shown by blue dots and red squares, respectively. The images are selected form the LFPW testing set (best viewed in color).



(a)

(b)

Figure 6.5: Comparing the proposed approach to state of the art methods on the LFPW dataset. (top) Mean normalized localization error, and (bottom) localization accuracy at threshold of $d < 0.10$.

ples, respectively. As mentioned earlier, a part of the dataset is not available anymore. We also compared our approach to the performance of human annotators by calculating the average error of a user in comparison to the mean of the other users [Yang and Patras, 2013a].

According to Figure 6.5, the proposed approach outperformed all the leading approaches except SO-RF [Yang and Patras, 2013a]. The main reason is that SO-RF method has considered the shape constraints, while our method captures the global appearance with no shape constraints. Lacking shape constraints for landmark detection sometimes leads to wrong detections with large location errors. Even a very small number of these wrong detections will result in a very large average error. This rarely happens in approaches with shape constraints. Our method obtained superior performance compared to Conditional Regression Forests (C-RF) [Dantone et al., 2012], Privileged Information-based Conditional Regression Forest (PI-CRF) [Yang and Patras, 2013b], [Belhumeur et al., 2011] for both mean detection error and localization rate. The average localization rate of our method for all the ten landmarks is 93.73% compared to 81.86% of C-RF and 83.80% PI-CRF. This superior performance is achieved by the capability of our method for local improvement over cascade levels and its robustness against wrong detection at the first level of the cascade. The accuracy of our method is competitive to the accuracy of human annotators. Our method even performed much better for five landmarks, *left eye right*, *right eye left*, *nose left*, *nose right* and *outer upper lip*.

The results on the BioID dataset are shown in Figure 6.7 and Table 6.1, where we compared our method to state of the art techniques on this dataset. Figure 6.7(a) shows the cumulative error versus the fractions of inter-ocular distance d for m_e17 (for 17 facial landmarks of all 19 internal landmarks) of our proposed approach and the state of the arts. These results are reported by [Zhou et al., 2013], [Belhumeur et al., 2011], [Valstar et al., 2010] and [Yang and Patras, 2013a]. The results show the superiority of our method against all the other approaches for almost all fractions of inter-ocular distance. This result demonstrates that our method achieved the state of the art performance for the BioID dataset and efficiently can detect all internal facial landmarks (by internal landmark we mean the points located inside the face image not on the face border, e.g. chain) when the face images are not captured under large pose variations.

Figure 6.7(b) illustrates the mean detection error normalized by inter-ocular distance (as pixel) of our method, [Valstar et al., 2010] and [Yang and Patras, 2013a] for all 19 internal landmarks (again chain landmark is discarded) of the BioID dataset. We borrowed the parts ID from defined [Yang and Patras, 2013a]. The mean error of P9 and P14 is not reported in [Valstar et al., 2010]. The detection rates for $d = 0.1$ of our method, [Valstar et al., 2010] and [Yang and Patras, 2013a] for all 19 internal landmarks of the BioId dataset are reported in Table 6.1. The results show that our method achieved the best localization rates for 13 of 19 landmarks, shown in **bold** font. Figure 6.8 visualizes the landmarks of some BioID images localized by our method.

Aside from the competitive accuracy, our approach achieved the superior detection speed by localizing an individual landmark of 400 face images in one second (2.5 ms to detect a landmark in a 128×128 face image), which is 16 times more faster than the state-of-the-art approaches with real-time detection speed (25 face images per second) reported by [Yang and Patras, 2013a], [Dantone et al., 2012] and [Yang and Patras, 2013b]. The detection time includes the amount of time required for computing the FFT/IFFT of the feature channels, landmark's filters and correlation outputs and evaluating the correlation outputs to find the global maximum over the entire correlation outputs, excluding the time required to detect the bounding box of faces and feature extraction. The main drawback of the cascaded framework, however, is that it is not able to refine the wrong detections occur at the first level. This means that if a landmark is wrongly detected in the first level (global filter) with large position error (e.g. the left mouth corner instead of the right eye center), it is not possible to correct the error down the cascade levels, a scenario which rarely happens in complicated frameworks which employ face shape constraints and prior information (e.g. pose, occlusion and expression) during the learning and testing processes.

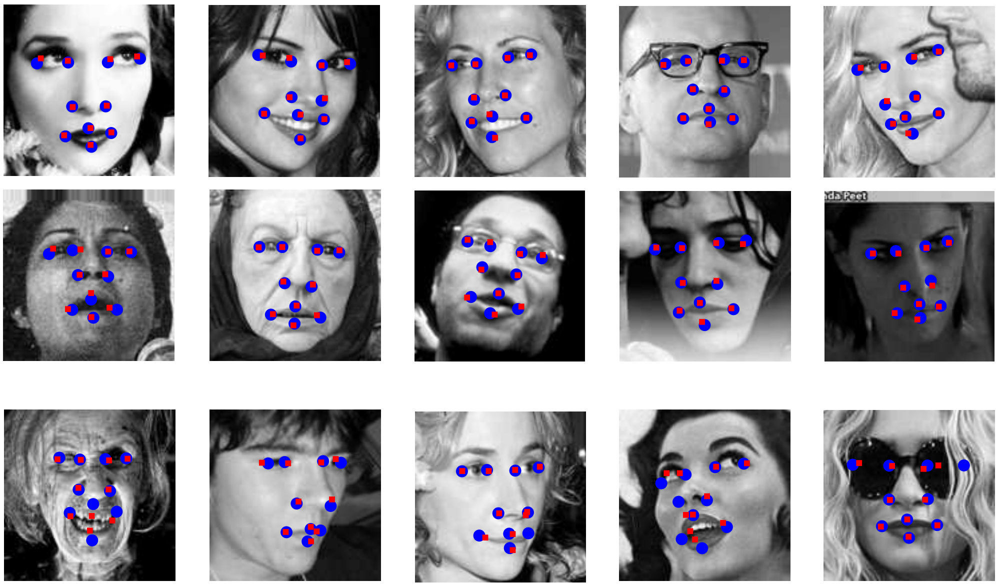
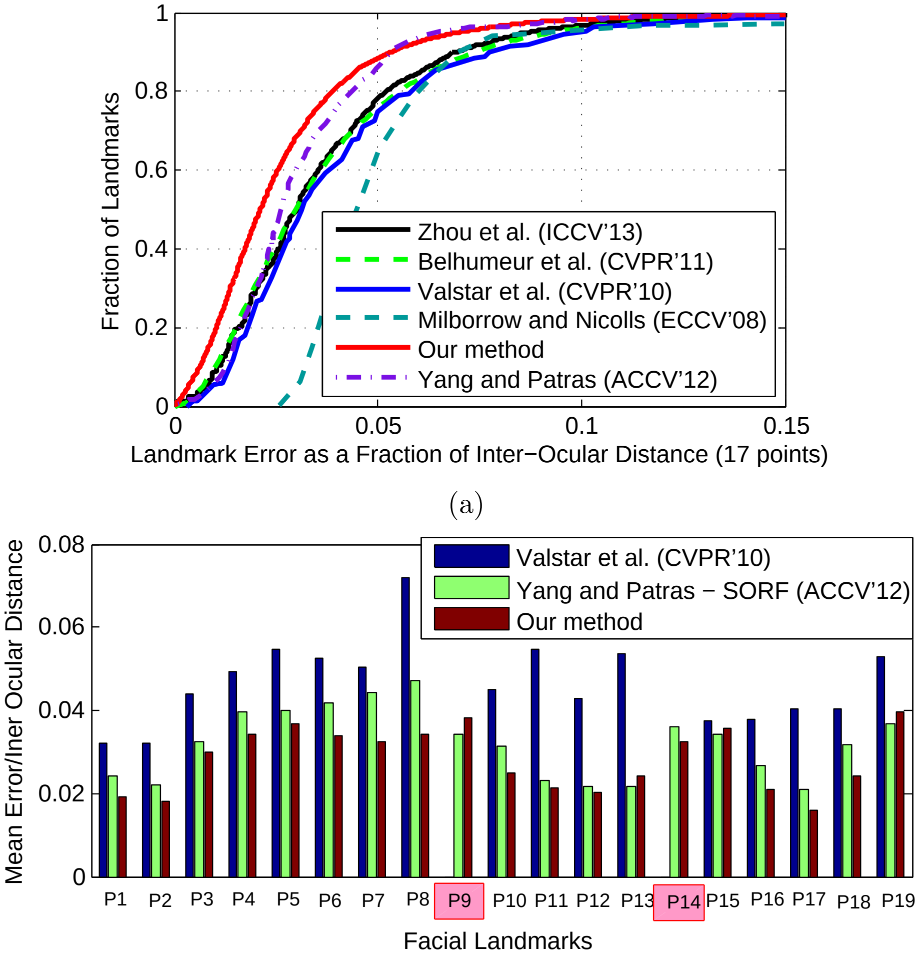


Figure 6.6: Detection examples of the LPFW dataset. The firs two rows show the successful detections under challenging circumstances of expression, occlusion, pose, lighting and poor quality. The third row shows some failed cases. The red and blue marks respectively show the detected landmark and the ground truth (best viewed in color).

Part	Valstar	SO-RF	Our method	Part	Valstar	SO-RF	Our method
P1	94.75%	98.25%	**99.35%**	P11	92.25%	100%	99.57%
P2	94.75%	98.50%	**99.14%**	P12	92.25%	100%	99.35%
P3	93.50%	98.50%	**98.71%**	P13	90.50%	99.75%	99.14%
P4	92.50%	99.00%	96.99%	**P14**	–	97.00%	**97.42%**
P5	89.00%	95.25%	**95.27%**	**P15**	96.25%	95.50%	**97.63%**
P6	90.25%	97.15%	**97.20%**	**P16**	93.50%	97.75%	**99.35%**
P7	91.25%	96.50%	**98.28%**	**P17**	93.25%	97.25%	**99.57%**
P8	81.00%	96.50%	**98.06%**	**P18**	95.00%	98.50%	**98.92%**
P9	–	97.75%	96.99%	P19	89.50%	97.25%	95.27%
10	92.25%	97.50%	**99.78%**				

Table 6.1: The detection rate of our approach, [Valstar et al., 2010] and [Yang and Patras, 2013a] for 19 individual landmarks of the BioID dataset. The **bold** parts are those landmarks our method achieved the highest detection rate for.



(a)

(b)

Figure 6.7: Comparison on the BioID dataset. (a) Average detection rate as a function of fraction of inter-ocular distance. (b) mean normalized error for each landmark. The parts (landmarks) ID are defined in [Yang and Patras, 2013a]. The mean error of P9 and P14 is not reported in [Valstar et al., 2010].

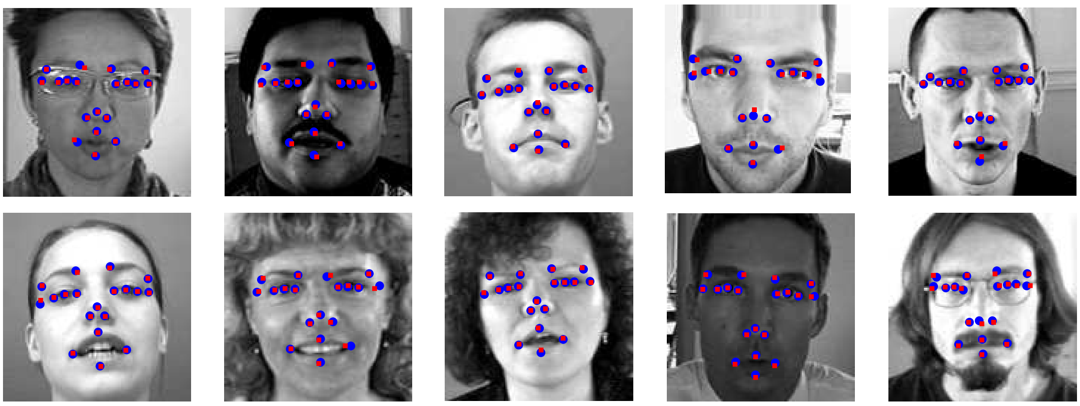


Figure 6.8: Detection examples of the BioID dataset. The red squares and blue dots represent the detected and ground truth landmarks, respectively (best viewed in color).

6.2.4 Comparison with Prior Correlation Filters

In this experiment we compare the proposed cascaded framework with the prior correlation filters in the literature and the works which are proposed in this thesis, including single-channel correlation filters (*MOSSE*) [Bolme et al., 2010], single-channel correlation filters with limited boundaries (*MOSSE w LB*, Chapter 5) [Kiani et al., 2014a], Multi-Channel Correlation filters (*MCCF*, Chapter 3) [Kiani et al., 2013], Multi-Channel Correlation Filters with Limited Boundaries (*MCCF w L*B, Chapter 5) and the cascaded framework (*Cascaded CF*) using the BioID dataset.

Similarly, 1000 of the face images are randomly selected for training and the rest for testing. We use normalized image intensities to train and test the single-channel correlation filters, and 43 feature channels (40 Gabor magnitudes, 2 Sobels and one normalized intensities) to train and test the multi-channel features. We employ 64 × 64 landmark patches cropped from the face images (centered upon the landmark of interest) to train the *MOSSE* and *MCCF* correlation filters and apply the trained filters on the 128 × 128 testing face images. For the correlation filters with limited boundaries (*MOSSE w L*B and *MCCF w L*B) we employ the entire 128 × 128 face images to train 64 × 64 landmark filters, and apply the trained filters on the 128 × 128 testing face images. The configuration, training and testing of the cascaded filters are same as the previous experiment.

The result of this experiment is shown in Figure 6.9. The lowest localization rate is obtained by MOSSE filter. During the experiment, we realized that most of the wrong detections were caused by (i) the landmarks with very similar visual appearance, e.g. left corners of the right eye and the left eye, and (ii) illumination variations. As mentioned earlier, this drawback of traditional correlation filters (e.g. MOSSE) can be improved using invariant and more discriminative multi-channel features (e.g. HOG rather than raw pixel values) and employing a large amount of negative training examples to accurately distinguish landmarks from non-landmark patches.

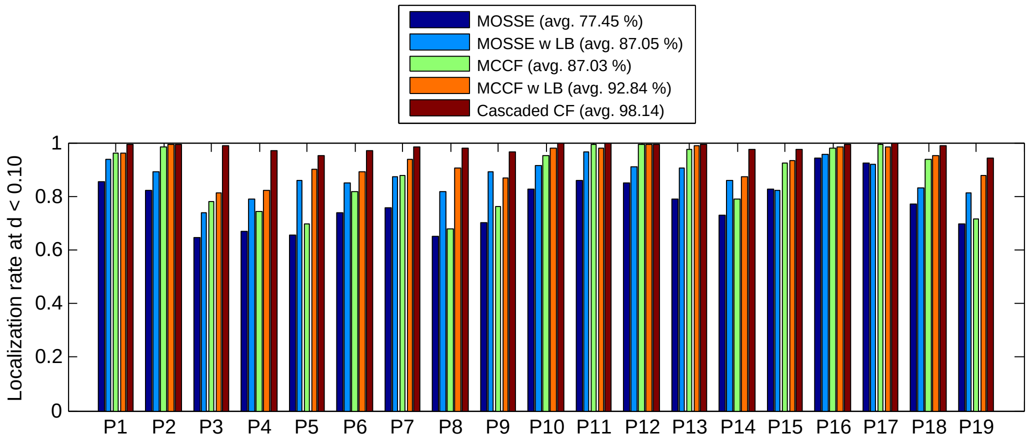


Figure 6.9: Comparison with prior correlation filters on the BioID dataset.

The localization rates obtained by the MOSSE with limited boundaries and MCCF are significantly higher (around 10 %) than the MOSSE filter, showing how using multi-channel features by MCCF and dense negative training patches by MOSSE with limited boundaries can reduce the wrong detections caused by visually similar landmarks and lighting changes, respectively. By combining the MCCF (multi-channel features) and MOSSE with limited boundaries (dense negative training patches) we increase the accuracy of these approaches around 5%.

In spite of great improvement form MOSSE (77.45 %) to MCCF with limited boundaries (92.84 %), there still remain some inevitable wrong detections even by using adequate negative training examples and discriminative features, due to the lack of face shape and geometric information. We reduce the number of these wrong detections by exploiting the geometric information of the entire face over a cascaded framework. The superior localization rate obtained by the cascaded framework (98.14%) states that the face geometric information is very useful to deal with challenging wrong detections.

6.3 Chapter Summary

In this chapter, we proposed a cascaded correlation filter framework for coarse-to-fine landmark detection in face images. In this framework, a set of correlation filters with different spatial supports are connected together in a way that the size of the correlation filter reduces downward to the lower levels. The role of the correlation filter at the first level is to predict the location of the landmark of interest. Since the spatial support of the correlation filter at the first level is large (same size of the face images), the prediction may suffer from small position error, specially for those landmarks which may be affected by face pose and expression. This position error is then corrected by filters at the lower levels of the cascade which are more robust against pose and expression. The evaluation on the LPFW and BioID datasets demonstrated the state of the art performance of our approach.

Chapter 7

Conclusions

This chapter concludes this thesis by summarizing the work presented in the previous chapters, including multi-channel correlation filters, correlation filters with limited boundaries, MCCF for human action recognition, cascaded correlation filters for facial landmark detection, evaluations and analysis. This is followed by several suggestions for possible future directions.

7.1 Summary

Pattern recognition is a challenging problem in computer vision which has received substantial attention over the years. The primary objective of this thesis was to develop techniques to improve pattern detection performance in terms of accuracy, memory usage and computational cost. Towards this purpose, we investigated the application of correlation filters to detect patterns in image/video and proposed advances to correlation filter theory to improve their performance to various pattern detection tasks.

It is well understood in computer vision that image intensity is not robust against challenging circumstances such as lighting changes, large intra-class variations and inter-class similarities. To deal with this, multi-channel features (e.g. HOG and SIFT) have been applied along with discriminative learning techniques on a wide range of vision tasks. These techniques, however, suffer from heavy computational cost and intractable memory usage. On the other hand, correlation filters show very efficient computation and memory usage since both training and testing can be performed in the Fourier domain. The main issue of traditional correlation filters, however, is that they are not able to handle multi-channel signals/features and, consequently, do not perform well under uncontrolled situations. First, we proposed multi-channel correlation filters to employ multichannel features for filter learning. We specifically demonstrated that the new objective in the Fourier domain is a sparse-banded linear system and can be efficiently optimized using a novel variable re-ordering technique. Like any learning approach, multi-channel correlation filters can be employed for a broad range of vision tasks including tracking, object detection, face recognition and video analysis. The comparison of the proposed approach with linear SVM for pedestrian classification showed the superiority of our method in terms of required memory and complexity in training and its comparable detection performance in the testing phase. The evaluation of our method for object detection (car and horse) demonstrated very comparable detection accuracy and superior detection speed of our method compared to leading spatial detectors.

Second, we proposed a different application of multi-channel correlation filters to recognize human action in video. The core idea was that each video of N frames can be considered as a $N \times M$ time-ordered feature channels, where M indicates the number of feature channels per frame (e.g. M is 1 for intensity and 5 for 5-bin HOG). This representation can be easily fed into the multi-channel correlation filter framework to learn action filters for action recognition/detection. The main advantages of MCCF over the previous correlation filters for action recognition can be summarized as its abilities to (1) employ both positive and negative training examples for filter learning, (2) specify the desired values over the entire correlation plane, and (3) recognize actions in real-time complexity. The experiments on the UCF sport and Weizmann action datasets demonstrated superior classification speed of our method compared to the state of the arts. In future work, we will show that the MCCF is not limited to *recognizing* action but similarly can be used to *detect* actions in video.

Third, we introduced a new objective to reduce the number of learning patches affected by boundary effect, called correlation filters with limited boundaries. By expressing the MOSSE correlation filter in the spatial domain, it was shown that traditional correlation filters extensively employ shifted versions of training patches for filter learning. These shifted patches which are implicitly produced by the circular property of convolution/correlation are not representative of real-world patches and may drastically reduce the discrimination of the trained filters. To deal with this, we proposed an augmented objective that learns correlation filters with limited spatial support whose size is much smaller than the size of training images. Specifically, we proposed the application of Alternating Direction Method of Multipliers (ADMM) to find the closed-form solution of the new objective with very efficient memory footprint and computations compared to spatial optimizers. An extension of correlation filters with limited boundaries was proposed to handle multi-channel features. The comparison of our method with a leading steepest descent method showed the superior performance of our method in terms of (1) the convergence time/speed as the number of training images and size of trained filters, and (2) the number of iterations required to converge. We demonstrated the superiority of our method against the state of the art correlation filters on the problem of eye localization in face images. Moreover, we evaluated our method on the problem of visual object tracking, where the proposed method outperformed the leading trackers under very challenging situations such as cluttered background, partial occlusion, lighting variations, extreme scaling and view point changes. Similar to the single-channel correlation filters with limited boundaries, the multi-channel extension can be applied on visual tracking and detection problems, provided that the size of training images are much larger than the desired trained filter.

Finally, we proposed to cascade a set of correlation filters with different spatial supports for facial landmark detection. The motivation was that correlation filters with large spatial support (at higher levels of the cascade) are able to approximate landmark positions with small position error mainly caused by face pose and expression. Whereas, the correlation filters with small spatial support (at lower cascade levels) are more robust against pose and expression, but may be affected by ambiguity of visually alike landmarks (e.g the left and right eye centers). As a result, we proposed to cascade different size of correlation filters for robust and accurate landmark detection. In this framework, the location/region of each landmark is first predicted by higher levels of the cascade. This prediction is then corrected downwards to the lower levels by smaller filters. The experiments on the LFPW and BioID benchmarks demonstrated the superiority of our method against the state of the arts.

7.2 Potential Application and Impact of Correlation Filters in Computer Vision

Hitherto, different types of Correlation filters have been developed to address many vision problems including automatic target recognition, face recognition, object detection, facial landmark localization, visual object tracking and human action recognition. Of course, there are other potential applications of correlation filters to real-world vision problems such as detection and classification problems in medical imaging (e.g. automatic mitosis detection in breast cancer tissue images and classification of human cell images) and video analysis (e.g. abnormal activity detection, surveillance, facial motion analysis etc), where real-time processing and efficient memory and computations are very important.

Furthermore, the impact of correlation filters in computer vision can be investigated from two practical perspectives. First, their efficient learning expenses that allow one to incrementally learn a pattern class from a huge amount of training examples (e.g. billions of images) with very manageable memory and computations. Second, using correlation filters in the Fourier domain can drastically reduce the detection and recognition time which most of current detector in the spatial domain suffer from (due to sliding window).

7.3 Future Work

The following research directions are suggested for future work:

- **Robust Correlation filters with L1-norm loss function.** It has been shown that L2-norm loss function is not very robust against outliers, as just a few (or even one) outliers with large squared error may drastically affect the filter training. L2-norm loss function, however, enjoys differentiability, a closed-form stable solution and efficient optimization which made it suitable to accomplish the primary objective of this thesis. On the other hand, L1-norm loss function has been known as a robust error function against outliers. It, however, is not differentiable and may suffer from multiple unstable solutions and heavy computations. All the approaches proposed in this thesis made use of L2-norm loss function for filter learning. Therefore, investigating the proposed objectives with L1-norm loss function for robust filter estimation is worth being pursued as future work.

- **L1-regularization vs. L2-regularization.** Regularization is a very important technique in machine learning to avoid over-fitting and stabilize the estimations against data collinearity (particularly matrix singularity in correlation filters). We employed the L2-regularization in our objectives since it is differentiable and computationally efficient. It, however, produces non-sparse coefficients (non-zero filter values) which may be noise-sensitive, redundant or irrelevant to estimate the desired outputs for unseen data (over-fitted to the training examples). L1-regularization, on the other hand, gives sparse (a small number of non-zero) coefficients and acts as an in-built feature selection mechanism. L1-regularization, however, does not have an analytical solution and thus suffers from high computational cost. Another possible research direction could be evaluating the proposed objectives with L1-normalization to figure out its influence on multi-channel correlation filter estimation.

- **Correlation filters for feature channel selection.** In recent years, numerous image representations are developed including shape and texture features (e.g. SIFT, HOG, SURF, LBP, GLOH, DAISY, Gabor phase and magnitude, etc.), color-based descriptions (e.g. RGB, HSV, LUV, etc.), motion-based features (e.g. temporal derivatives, optical flow, 3DHOG, 3DSIFT, etc.) with various extensions to improve the performance of a wide range of vision tasks such as image/video retrieval, object detection and recognition, biometrics, matching, tracking, registration, etc. Despite the great achievements in feature representation and learning, the question of which feature channels are more appropriate for a particular vision task still remains unanswered. The capability of MCCF to jointly handle feature channels allows one to select the best task-specific feature channels. This can intuitively done by adding a constraint on the channel cardinality to select the optimal features that minimizes the MCCF objective. Feature selection via MCCF can open a new direction of correlation filters behind pattern detection and matching.

- **Correlation filters to handle large intra-class variation.** An object class may include a wide variation of the object differing in shape, color, size, pose, etc. Handling such a large intra-class variation is the *Achilles heel* of correlation filter techniques. This is mainly caused by the linear form of the spectral energies in 3.8, where all the training images are simply averaged to represent the object class. One trivial solution could be training a set of correlation filters instead of one single filter to cover all possible variations in the target class. This may work for objects with limited variations (e.g cars and faces) but is impractical for cases with huge variations (e.g. birds and chairs). Learning robust correlation filters for object detection/recognition with large variation could be considered as the main direction for the future work on correlation filters.

7.4 List of Publications

- **Hamed Kiani**, Terence Sim and Simon Lucey, "Multi-Channel Correlation Filters", IEEE International Conference on Computer Vision (ICCV'13), 2013 *(Chapter 3).*

- **Hamed Kiani**, Terence Sim and Simon Lucey, "Multi-Channel Correlation Filters for Human Action Recognition", ICIP'14, 2014 *(Chapter 4).*

- **Hamed Kiani**, Terence Sim and Simon Lucey, "Correlation Filters with

Limited Boundaries", arXiv, 2014 *(Chapter 5).*

- **Hamed Kiani** and Terence Sim, "Face Photo Retrieval by Sketch Example", ACM Multimedia, ACM MM'12, 2012.

- **Hamed Kiani** and Terence Sim, "Inter-modality Face Sketch Recognition", IEEE International Conference on Multimedia and Expo (ICME'12), 2012.

- **Hamed Kiani** and Terence Sim, "Sketch Recognition by Local Radon Binary Pattern: LRBP", IEEE International Conference on Image Processing (ICIP"12), 2012.

- Kart Lim and **Hamed Kiani**, "Shape Classification Using Local and Global Features", Proceeding of the 4th Pacific-Rim Symposium on Image and Video Technology (PSIVT'10), 2010.

Bibliography

Adam, A., Rivlin, E., and Shimshoni, I. (2006). Robust fragments-based tracking using the integral histogram. In *CVPR*, volume 1, pages 798–805.

Agarwal, S. and Roth, D. (2002). Learning a sparse representation for object detection. In *ECCV (4)*, pages 113–130.

Ali, S. and Lucey, S. (2010). Are correlation filters useful for human action recognition? In *ICPR*, pages 2608–2611.

Alkanhal, M. and Vijaya Kumar, B. (2003). Polynomial distance classifier correlation filter for pattern recognition. *Applied optics*, 42(23):4688–4708.

Azizpour, H. and Laptev, I. (2012). Object detection using strongly-supervised deformable part models. In *Computer Vision–ECCV 2012*, pages 836–849. Springer.

Babenko, B., Yang, M.-H., and Belongie, S. (2011). Robust object tracking with online multiple instance learning. *PAMI*, 33(8):1619–1632.

Belhumeur, P. N., Jacobs, D. W., Kriegman, D., and Kumar, N. (2011). Localizing parts of faces using a consensus of exemplars. In *IEEE Conference on Computer Vision and Pattern Recognition.*

Blank, M., Gorelick, L., Shechtman, E., Irani, M., and Basri, R. (2005). Actions as space-time shapes. In *ICCV*, pages 1395–1402.

Bolme, D. S., Beveridge, J. R., Draper, B. A., and Lui, Y. M. (2010). Visual object tracking using adaptive correlation filters. In *CVPR*.

Bolme, D. S., Draper, B. A., and Beveridge, J. R. (2009). Average of synthetic exact filters. In *CVPR*.

Borgefors, G. (1988). Hierarchical chamfer matching: A parametric edge matching algorithm. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 10(6):849–865.

Boyd, S. (2010). Distributed Optimization and Statistical Learning via the Alternating

Direction Method of Multipliers. *Foundations and Trends in Machine Learning*, 3:1–122.

Bracewell, R. N. and Bracewell, R. (1986). *The Fourier transform and its applications*, volume 31999. McGraw-Hill New York.

Breiman, L. (1996). Bagging predictors. *Machine learning*, 24(2):123–140.

Cai, Q., Yin, Y., and Man, H. (2013). Dspm: Dynamic structure preserving map for action recognition. In *ICME*, pages 1–6.

Casasent, D. and Chang, W.-T. (1986). Correlation synthetic discriminant functions. *Applied Optics*, 25(14):2343–2350.

Cevikalp, H. and Triggs, B. (2012). Efficient object detection using cascades of nearest convex model classifiers. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3138–3145. IEEE.

Cooley, J. W. and Tukey, J. W. (1965). An algorithm for the machine calculation of complex fourier series. *Mathematics of computation*, 19(90):297–301.

Cootes, T. F., Edwards, G. J., Taylor, C. J., et al. (2001). Active appearance models. volume 23, pages 681–685.

Cootes, T. F., Taylor, C. J., Cooper, D. H., and Graham, J. (1995). Active shape models-their training and application. volume 61, pages 38–59. Elsevier.

Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *CVPR*, volume 1, pages 886–893. IEEE.

Dalal, N., Triggs, B., and Schmid, C. (2006). Human detection using oriented histograms of flow and appearance. In *Computer Vision–ECCV 2006*, pages 428–441. Springer.

Dantone, M., Gall, J., Fanelli, G., and Van Gool, L. (2012). Real-time facial feature detection using conditional regression forests. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2578–2585. IEEE.

de Campos, T., Barnard, M., Mikolajczyk, K., Kittler, J., Yan, F., Christmas, W. J., and Windridge, D. (2011). An evaluation of bags-of-words and spatio-temporal shapes for action recognition. In *WACV*, pages 344–351.

Del Bue, A., Xavier, J., Agapito, L., and Paladini, M. (2011). Bilinear Modelling via Augmented Lagrange Multipliers (BALM). *PAMI*, X(X):1–14.

Déniz, O., Bueno, G., Salido, J., and De la Torre, F. (2011). Face recognition using histograms of oriented gradients. *Pattern Recognition Letters*, 32(12):1598–1603.

Dollar, P., Rabaud, V., Cottrell, G., , and Belongie, S. (2005). Behavior recognition via sparse spatio-temporal features. In *VS-PETS*.

Efros, A. A., Berg, A. C., Mori, G., and Malik, J. (2003). Recognizing action at a distance. In *ICCV*, volume 1, pages 357–360.

Felzenszwalb, P. F., Girshick, R. B., McAllester, D., and Ramanan, D. (2010). Object detection with discriminatively trained part-based models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(9):1627–1645.

Fernandez, J. A. and Kumar, B. V. K. V. (2013). Space-time correlation filters for human action detection. In *SPIE*.

Ferrari, V., Fevrier, L., Jurie, F., and Schmid, C. (2008). Groups of adjacent contour segments for object detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(1):36–51.

Ferrari, V., Jurie, F., and Schmid, C. (2010). From images to shape models for object detection. *International Journal of Computer Vision*, 87(3):284–303.

Finlayson, G. D., Chatterjee, S. S., and Funt, B. V. (1996). Color angular indexing. In *Computer VisionECCV'96*, pages 16–27. Springer.

Freund, Y. and Schapire, R. E. (1995). A desicion-theoretic generalization of on-line learning and an application to boosting. In *Computational learning theory*, pages 23–37. Springer.

G. Gualdi, A. Prati, R. C. (2012). Multistage particle windows for fast and accurate object detection. *PAMI*, 34(8):1589–1604.

Gall, J. and Lempitsky, V. (2013). Class-specific hough forests for object detection. In *Decision Forests for Computer Vision and Medical Image Analysis*, pages 143–157. Springer.

Gall, J. and Lempitsky, V. S. (2009). Class-specific hough forests for object detection. In *CVPR*, pages 1022–1029.

Grabner, H., Grabner, M., and Bischof, H. (2006). Real-time tracking via on-line boosting. In *BMVC*.

Grabner, H., Leistner, C., and Bischof, H. (2008). Semi-supervised on-line boosting for robust tracking. In *ECCV*. Springer.

H. Harzallah, F. J. and Schmid, C. (2009). Combining efficient object localization and image classification. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 237–244.

Hare, S., Saffari, A., and Torr, P. H. (2011). Struck: Structured output tracking with kernels. In *ICCV*.

He, R., Zheng, W.-S., Hu, B.-G., and Kong, X.-W. (2011). A regularized correntropy framework for robust pattern recognition. *Neural Computation*, 23(8):2074–2100.

Henriques, J. F., Caseiro, R., Martines, P., and Batista, J. (2012). Exploiting the circulant structure of tracking-by-detection with kernels. In *ECCV*.

Hester, C. F. and Casasent, D. (1980). Multivariant technique for multiclass pattern recognition. *Appl. Opt.*, 19(11):1758–1761.

Huang, Z. F., Yang, W., Wang, Y., and Mori, G. (2011). Latent boosting for action recognition. In *BMVC*, pages 1–11.

Hubel, D. and Wiesel, T. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of Physiology*, 160(1):106.

Huttenlocher, D. P., Klanderman, G. A., and Rucklidge, W. J. (1993). Comparing images using the hausdorff distance. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 15(9):850–863.

Jarrett, K., Kavukcuoglu, K., Ranzato, M. A., and LeCun, Y. (2009). What is the best multi-stage architecture for object recognition? *International Conference on Computer Vision*, pages 2146–2153.

Jeong, K.-H., Liu, W., Han, S., Hasanbelliu, E., and Principe, J. C. (2009). The correntropy mace filter. *Pattern Recognition*, 42(5):871–885.

Jeong, K.-H., Pokharel, P. P., Xu, J.-W., Han, S., and Principe, J. C. (2006). Kernel based synthetic discriminant function for object recognition. In *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, volume 5, pages V–V. IEEE.

Jeong, K.-H. and Principe, J. C. (2006). The correntropy mace filter for image recognition. In *Machine Learning for Signal Processing, 2006. Proceedings of the 2006 16th IEEE Signal Processing Society Workshop on*, pages 9–14. IEEE.

Jesorsky, O., Kirchberg, K. J., and Frischholz, R. W. (2001). Robust face detection using the hausdorff distance. In *Audio-and video-based biometric person authentication*, pages 90–95. Springer.

Kiani, H., Sim, T., and Lucey, S. (2013). Multi-channel correlation filters. In *ICCV*.

Kiani, H., Sim, T., and Lucey, S. (2014a). Correlation filters with limited boundaries. In *arXiv*.

Kiani, H., Sim, T., and Lucey, S. (2014b). Multi-channel correlation filters for human action recognition. In *ICIP*.

Kluser, A., Marszalek, M., and Schmid, C. (2008). A spatio-temporal descriptor based on 3d-gradients. In *BMVC*, pages 1–10.

Kumar, B. V. (2005). *Correlation pattern recognition.* Cambridge University Press.

Kumar, B. V. K. V. (1986). Minimum-variance synthetic discriminant functions. *J. Opt. Soc. Am. A*, 3(10):1579–1584.

Kuo, C.-H. and Nevatia, R. (2009). Robust multi-view car detection using unsupervised sub-categorization. In *WACV*, pages 1–8.

Lampert, C. H., Blaschko, M. B., and Hofmann, T. (2008). Beyond sliding windows: Object localization by efficient subwindow search. In *CVPR*.

Laptev, I. (2009). Improving object detection with boosted histograms. *Image and Vision Computing*, 27(5):535–544.

Laptev, I., Marsza?ek, M., Schmid, C., and Rozenfeld, B. (2008). Learning realistic human actions from movies. In *CVPR*, pages 1–8.

Leibe, B., Leonardis, A., and Schiele, B. (2008). Robust object detection with interleaved categorization and segmentation. *International Journal of Computer Vision*, 77(1-3):259–289.

Lowe, D. (1999). Object recognition from local scale-invariant features. *International Conference on Computer Vision*, pages 1150–1157 vol.2.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. volume 60, pages 91–110. Springer.

Mahalanobis, A. and Kumar, B. V. (1997). Polynomial filters for higher order correlation and multi-input information fusion. In *Optoelectronic Information Processing: Invited*

Contributions from a Workshop Held 2-5 June 1997, Barcelona, Spain, page 1221. SPIE Press.

Mahalanobis, A., Kumar, B. V. K. V., and Casasent, D. (1987). Minimum average correlation energy filters. *Appl. Opt.*, 26(17):3633–3640.

Mahalanobis, A., Vijaya Kumar, B., Song, S., Sims, S., and Epperson, J. (1994). Unconstrained correlation filters. *Applied Optics*, 33(17):3751–3759.

Maji, S., Berg, A. C., and Malik, J. (2008). Classification using intersection kernel support vector machines is efficient. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE.

Milborrow, S. and Nicolls, F. (2008). Locating facial features with an extended active shape model. In *Computer Vision–ECCV 2008*, pages 504–513. Springer.

Monroy, A., Eigenstetter, A., and Ommer, B. (2011). Beyond straight lines - object detection using curvature. In *ICIP*, pages 3561–3564.

Munder, S. and Gavrila, D. M. (2006). An experimental study on pedestrian classification. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(11):1863–1868.

Mutch, J. and Lowe, D. G. (2006). Multiclass object recognition with sparse, localized features. In *CVPR (1)*, pages 11–18.

Nayar, S. K. and Bolle, R. M. (1996). Reflectance based object recognition. *International Journal of Computer Vision*, 17(3):219–240.

Paul Scovanner, S. A. and Shah, M. (2007). A 3-dimensional sift descriptor and its application to action recognition. In *ACM Multimedia*, volume 1, pages 357–360.

Refregier, P. (1991). Optimal trade-off filters for noise robustness, sharpness of the correlation peak, and horner efficiency. *Optics Letters*, 16(11):829–831.

Rodriguez, A., Boddeti, V., Kumar, B., and Mahalanobis, A. (2013). Maximum margin correlation filter: A new approach for localization and classification. *TIP*, 22(2):631–643.

Rodriguez, M. D., Ahmed, J., and Shah, M. (2008). Action mach a spatio-temporal maximum average correlation height filter for action recognition. In *CVPR*.

Ross, D., Lim, J., Lin, R., and Yang, M. (2008). Incremental learning for robust visual tracking. *IJCV*, 77(1):125–141.

Salas, J. and Tomasi, C. (2011). People detection using color and depth images. In *Pattern Recognition*, pages 127–135. Springer.

Saragih, J. (2011). Principal regression analysis. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 2881–2888. IEEE.

Saragih, J. M., Lucey, S., and Cohn, J. F. (2011). Deformable model fitting by regularized landmark mean-shift. volume 91, pages 200–215. Springer.

Savvides, M. and Kumar, B. V. K. V. (2003). Efficient design of advanced correlation filters for robust distortion-tolerant face recognition. In *AVSS*, pages 45–52.

Shahbaz Khan, F., Anwer, R. M., van de Weijer, J., Bagdanov, A. D., Vanrell, M., and Lopez, A. M. (2012). Color attributes for object detection. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3306–3313. IEEE.

Slater, D. and Healey, G. (1996). The illumination-invariant recognition of 3d objects using local color invariants. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 18(2):206–210.

Swain, M. J. and Ballard, D. H. (1991). Color indexing. *International journal of computer vision*, 7(1):11–32.

Thornton, J., Savvides, M., and Vijaya Kumar, B. (2004). Linear shift-invariant maximum margin svm correlation filter. In *Intelligent Sensors, Sensor Networks and Information Processing Conference, 2004. Proceedings of the 2004*, pages 183–188. IEEE.

Tian, Y., Sukthankar, R., and Shah, M. (2013). Spatiotemporal deformable part models for action detection. In *CVPR*, pages 2642–2649.

Toshev, A., Taskar, B., and Daniilidis, K. (2010). Object detection via boundary structure segmentation. In *CVPR*, pages 950–957.

Valstar, M., Martinez, B., Binefa, X., and Pantic, M. (2010). Facial point detection using boosted regression and graph models. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2729–2736. IEEE.

Vedaldi, A. and Zisserman, A. (2012). Efficient additive kernels via explicit feature maps. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(3):480–492.

Villamizar, M., Andrade-Cetto, J., Sanfeliu, A., and Moreno-Noguer, F. (2012). Bootstrapping boosted random ferns for discriminative and efficient object classification. *Pattern Recognition*, 45(9):3141–3153.

Viola, P. and Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–511. IEEE.

Vukadinovic, D. and Pantic, M. (2005). Fully automatic facial feature point detection using gabor feature based boosted classifiers. In *Systems, Man and Cybernetics, 2005 IEEE International Conference on*, volume 2, pages 1692–1698. IEEE.

Wang, H., Ullah, M. M., Kläser, A., Laptev, I., and Schmid, C. (2009). Evaluation of local spatio-temporal features for action recognition. In *BMVC*, pages 1–11.

Wang, Y. and Mori, G. (2011). Hidden part models for human action recognition: Probabilistic versus max margin. In *PAMI*, volume 33(7), pages 1310–1323.

Willems, G., Tuytelaars, T., and Gool1, L. V. (2008). An efficient dense and scale-invariant spatio-temporal interest point detector. In *ECCV*, pages 650–663.

Xie, C., Savvides, M., and VijayaKumar, B. (2005). Kernel correlation filter based redundant class-dependence feature analysis (kcfa) on frgc2. 0 data. In *Analysis and Modelling of Faces and Gestures*, pages 32–43. Springer.

Yang, H. and Patras, I. (2013a). Face parts localization using structured-output regression forests. In *Computer Vision–ACCV 2012*, pages 667–679. Springer.

Yang, H. and Patras, I. (2013b). Privileged information-based conditional regression forest for facial feature detection. In *Automatic Face and Gesture Recognition (FG), 2013 10th IEEE International Conference and Workshops on*, pages 1–6. IEEE.

Yeffet, L. and Wolf, L. (2009). Local trinary patterns for human action recognition. In *ICCV*, pages 492–497.

Zeiler, M., Krishnan, D., and Taylor, G. (2010). Deconvolutional networks. *CVPR*.

Zhao, X., Chai, X., Niu, Z., Heng, C., and Shan, S. (2012). Context modeling for facial landmark detection based on non-adjacent rectangle (nar) haar-like feature. volume 30, pages 136–146. Elsevier.

Zhao, X., Shan, S., Chai, X., and Chen, X. (2013). Locality-constrained active appearance model. In *Computer Vision–ACCV 2012*, pages 636–647. Springer.

Zhou, F., Brandt, J., and Lin, Z. (2013). Exemplar-based graph matching for robust facial landmark localization. In *ICCV*, pages 1–6. IEEE.

Zhu, X. and Ramanan, D. (2012). Face detection, pose estimation, and landmark localization in the wild. In *IEEE Conference on Computer Vision and Pattern Recognition*.