Are Correlation Filters Useful for Human Action Recognition?

Saad Ali  
Carnegie Mellon University  
saada@cs.cmu.edu

Simon Lucey  
Carnegie Mellon University  
slucey@cs.cmu.edu

Abstract

It has been argued in recent work that correlation filters are attractive for human action recognition from videos. Motivation for their employment in this classification task lies in their ability to: (i) specify where the filter should peak in contrast to all other shifts in space and time, (ii) have some degree of tolerance to noise and intra-class variation (allowing learning from multiple examples), and (iii) can be computed deterministically with low computational overhead. Specifically, Maximum Average Correlation Height (MACH) filters have exhibited encouraging results [4] on a variety of human action datasets. Here, we challenge the utility of correlation filters, like the MACH filter, in these circumstances. First, we demonstrate empirically that identical performance can be attained to the MACH filter by simply taking the average of the same action specific training examples. Second, we characterize theoretically and empirically under what circumstances a MACH filter would become equivalent to the average of the action specific training examples. Based on this characterization, we offer an alternative type of filter, based on a discriminative paradigm, that circumvent the inherent limitations of correlation filters for action recognition and demonstrate improved action recognition performance.

1. Introduction

Action recognition from videos is an important area of research in computer vision. In general, these approaches can be categorized on the basis of the representation used for actions. Some leading representations are learned geometrical models of human body parts ([5]), space-time pattern templates, appearance or region features, shape or form features [6, 7], interest-point-based representations [8, 9], volumetric features [7, 10], and motion/optical flow patterns [11, 12].

In recent years, the application of correlation filters for action recognition has yielded some promising results [4]. The core idea is to learn a correlation filter in the 3D frequency domain so that a peak occurs at origin of the action with respect to shifts in both space and time. Additionally, invariance to noise and intra-class variations can be included into the objective function from which the filter is estimated. Many variants to correlation filters, however, have been proposed in literature [1], for a variety of applications (e.g., automatic target recognition etc.), based on this central idea. Specifically, in [4], optimal tradeoff Maximum Average Correlation Height (ot-MACH) filters were used, which are variants of MACH filters, as a way of learning a filter from multiple examples of an action. MACH filters exhibit a number of characteristics that make them of interest to the action recognition community. First, they can be easily extended to learn from multiple examples of the same action. Previous to this work, many methods for action recognition [3, 7] were learned from a single example which subsequently did not generalize well to unseen testing conditions. Second, learning and evaluation is done in the 3D frequency domain where efficient 3D convolutions can be performed in space and time. Finally, the MACH filter has a closed form solution which makes the learning process computationally very efficient.

However, as we shall elucidate upon in this paper, correlation filters in most practical circumstances have fundamental limitations making them unsuitable for action recognition. Specifically, we demonstrate empirically that MACH filters, which are learned from examples for a specific action, obtain identical performance to the average filter (i.e., taking the arithmetic mean of those same training examples). To understand this empirical result, we characterize under what circumstances a MACH filter would be equivalent to the average filter. Based on this characterization we propose that the primary learning goal of correlation filters is at odds with datasets, such as those found in the action recognition community, that are poorly aligned in space and time. Most actions in these datasets are either out of phase with each other, or occur at varying rates making them unsuitable for modeling using correlation filters. These misaligned training examples violate the central goal of correlation filters (i.e., minimize the correlation energy across all possible shifts except at the origin) which assumes that the training data should be partially aligned. In this paper we argue that action filters should instead concentrate upon leveraging discriminative class information during learning, rather than enforcing where the filter should peak in contrast to all other shifts in space and time.

2. Correlation Filters

All correlation filters can be loosely defined as attempting to find the filter h that minimizes the Average Similarity Measure (ASM),

\[
\text{ASM} = \frac{1}{N} \sum_{i=1}^{N} \sum_{\tau} ||h^T x_i(\tau)||^2
\]

across all possible circular shifts of the training data subject to some constraint or objective at the origin of the training data (more on these constraints/objectives in Section 2.1). Where \( x_i \) is the \( i \)th vectorized training video example where \( \tau = [\Delta \tau, \Delta \gamma, \Delta t]^T \) is the circular operator in space and time applied to the training examples. A drawback with this space-time representation of ASM is that it requires the explicit computation of all possible circular shifts \( \tau \) across all \( N \) training examples. Instead, correlation filters can attempt to solve for \( h \) with respect to the ASM in a more computational
tractable manner, \( \text{ASM} = \hat{h}^T D \hat{h} \) where \( \hat{h} \) and \( \hat{x}_i \) are the vectorized complex 3D discrete Fourier transforms (DFT) \(^2\) of the vectorized space-time volumes \( h \) and \( x_i(0) \) respectively. Where \( D \) commonly referred to as the average spectral variance is defined as\(^\text{3}\), \( D = \frac{1}{N} \sum_{i=1}^{N} \text{diag}(\hat{x}_i) \). Learning \( h \) in the frequency domain (i.e. \( \hat{h} \)) allows for the explicit estimation of the ASM without the computationally costly step of manually performing circular shifts for all possible \( x, y \) and \( ts \) in the training data.

2.1. MACH filter

It is outside the scope of this paper to give a full review on the topic of correlation filters. Interested readers are encouraged to inspect [1] for a full treatment on the topic. Most techniques, however, can be viewed as varying in terms of constraining or maximizing an objective function at the origin response of the filter \( h \) while still trying to minimize the ASM (i.e., minimize the average correlation energy for \( h \) across all circular shifts). For example, Maximum Average Correlation Height (MACH) filters attempt to maximize the Average Correlation Height (ACH),

\[
\text{ACH} = \frac{1}{N} \sum_{n=1}^{N} ||\hat{h}^T x_i^T(0)||^2
\]

while minimizing the ASM. Note that the ACH, in Equation 2, is quite similar to the ASM, in Equation 1, with the exception that the ACH is a measure of correlation energy at the origin of the training data (denoted by \( x_i(0) \)) while the ASM is a measure of correlation energy across all circular shifts \( \tau \) (denoted by \( x_i(\tau) \)).

2.2. Optimal tradeoff MACH filter

In practice, maximizing the ACH while minimizing the ASM alone can cause overfitting so it has become common practice in correlation filter literature [1] to add a noise/regularization term, denoted by \( C \), and the intraclass spectral variance, denoted by \( S \), to \( D \). Where, \( S = \frac{1}{N} \sum_{i=1}^{N} \text{diag}(\hat{x}_i - \bar{m})^2 \text{diag}(\hat{x}_i - \bar{m}) \), and \( C = I \). Including these additional terms results in finding the filter \( h \) that minimizes the following objective function, \( J(h) = \frac{\text{h}^T \text{mm}^T \text{h}}{\text{h}^T ((1 - \alpha)S + \alpha C) \text{h}} \), which is referred to in correlation filter literature [1] as the optimal tradeoff MACH filter. The weights \( \alpha, \beta \) and \( \gamma \) can be tuned through cross-validation to optimize performance. Fortunately, however, it has been empirically demonstrated in previous work of [4] (and the accompanying code on their web-site) that \( D \) has a very minimal effect on action classification performance. For the purposes of this paper, the final optimal tradeoff MACH filter objective function can be found by simplified to minimizing,

\[
J(h) = \frac{\text{h}^T \text{mm}^T \text{h}}{\text{h}^T ((1 - \alpha)S + \alpha C) \text{h}}
\]

\(^\text{1}\)Please note that throughout this paper that the notation ‘\( \cdot \)’ applied to any vector denotes the 3D-DFT of a vectorized 3D video volume such that \( \hat{x} \leftarrow \text{F}x \), where \( \text{F} \) is the \( N \times N \) matrix of complex basis vectors for mapping to the 3D Fourier domain for any \( N \) dimensional vectorized video volume.

\(^\text{2}\)In Equation 2 we are taking advantage of the fact that \( \text{diag}(\hat{x}) \) is an operator that transforms a \( N \) dimensional vector into a \( N \times N \) dimensional diagonal matrix. We should also note that any transpose operator \(^\text{3}\) on a complex vector or matrix in this paper additionally takes the complex conjugate in a similar fashion to the Hermitian adjoint [3].

\(^\text{3}\)The match-score of an input video with respect to a learned filter is found through the peak to side lobe ratio (PSR) estimated from the correlation surface. This approach is different from the one adopted by [4] where maximum of the correlation surface is used as a match-score. Specifically, the PSR is defined as:

\[
\text{PSR} = \frac{\text{Peak} - \mu}{\sigma},
\]

where \( \mu \) and \( \sigma \) are the mean and standard deviation of the correlation values in some neighborhood of the peak time.

3. Varying \( \alpha \)

In this section, we study the effect of varying \( \alpha \) with respect to action recognition performance. We used two publicly available data sets. We picked these data sets as they have minimum amount of clutter, and therefore allows us to investigate issues arising from misalignment of action in space and time.

3.1. Experiments

We conducted an experiment where the value of \( \alpha \) varied from 0 to 1. As \( \alpha \) increased the contribution from the spectral variance term, \( S \), becomes less and less important. For each value of \( \alpha \), one MACH filter per action class is learned. At testing stage, MACH filters of each action class are correlated with the test video, and label corresponding to the MACH filter that returns the maximum PSR value is assigned to the test video. In case of Weizman action dataset, the testing is performed in a leave-one-out setting where the MACH filter of each action class is learned using all the examples of that class except the example on which the testing needs to be performed.

The results of these experiments are summarized in graphs shown in Figure 1. On the \( y \)-axis we have the mean recognition accuracy while on the \( x \)-axis we have increasing values of \( \alpha \). Experiments were performed using various types of features and each line in the graphs correspond to one of these features. It is evident from the graph that increasing the value of \( \alpha \) results in better performance in terms of recognition. The best results are obtained when \( \alpha = 0.9 \) except in case of normalized intensity feature where the best results are obtained for \( \alpha = 0.9 \). However, the interesting aspect of the result is
that the trend of better performance with increasing $\alpha$ is main
tained across different types of features.

3.2. Average Filter

From Eq. (3), it is easy to show that when $\alpha = 1$ this
is equivalent to learning an average filter since we are now
finding the eigenvectors of $(C)^{-1}\min^2$ where $C = I$. The
results in the previous section indicate that in nearly all cir-
cumstances the average filter (i.e., $\alpha = 1$) outperformed all
other versions of the MACH filter. This result can be under-
stood by realizing that the spectral variance $S$ is at the heart
of learning a distortion tolerant MACH filter. The frequen-
cies where the magnitude of $S$ are high are unattractive for
in-class distortion tolerance since these are the frequencies at
which the target action class exhibits a large variance. There
fore, in Eq. (3), the inverse of $S$ is used as it attenuates fre-
quencies where samples of an action vary most in space and
time. However, when samples of an action are misaligned
high spectral variance is observed for frequencies that corre-
spond to the locations of moving limbs such as hands, feet
and torso in the spatial domain. These frequencies in turn ge-
suppressed in the learned filter thus reducing the distortion tol-
erance of the filter as moving limbs are the basic ingredient
which discriminate one action against the other. A visualiza-
tion of these spectral variance volumes for various actions i
presented in Figure 2. Visualization of MACH filters for vari-
ous values of $\alpha$ is provided in Figures 3-4. These are learned
from temporal derivatives of videos of each action class. It
can be observed from the figure that filters become smoother
as the value of $\alpha$ is increased.

4. Discriminative Filters

We offer an alternative type of filter, based on a discrimina-
tive paradigm, that circumvent the inherent limitations of cor-
relation filters for action recognition. These filters are based
on reinterpretation of few important concepts related to SVM.

4.1. Linear SVMs in the Fourier Domain

SVMs have been demonstrated to be useful for many clas-
sification tasks [13]. Given a set of training example pairs
$(x_i, y_i), i = 1, \ldots, l, x_i \in \mathbb{R}^N, y_i \in \{+1, -1\}$, a
linear SVM attempts to find the solution to the follow-
ing unconstrained optimization problem, $\min_w \frac{1}{2}w^T w +
C\sum_{i=1}^{l} \max(1 - y_i w^T x_i, 0)^2$, where $C$ is a penalty pa-
rameter and the bias $b$ is accounted for in $w \leftarrow [w^T, b]$ by $x \leftarrow
[x^T, 1]$. The parameters of the hyper-plane are estimated by
maximizing the margin term (i.e., minimizing $w^T w$) and
minimizing an upper bound (i.e., the hinge error function)
for the misclassification rate on the training examples. Above
equation is often referred to as solving the primal form of an
SVM. One may instead solve the dual problem,

$$\max_{0 \leq \alpha \leq C} \quad -\frac{1}{2} \sum_{i=1}^{l} \sum_{j=1}^{l} \alpha_i \alpha_j y_i y_j x_i^T x_j + \sum_{i=1}^{l} \alpha_i$$ (4)

subject to $\alpha^T y = 0$ (5)

As pointed out by [15] it is easy to see that the dual of Equa-
tion 4.1 is rotation invariant. For example if all $x_i$ were re-
placed by $A x_i$, where $A \in \mathbb{R}^N \times N, A^T A = I$, then the
solution remains the same. Interestingly, one can view the
application of a 3D-DFT as multiplication by a complex or-
thonormal basis $\tilde{x}_i = F x_i$ where $F^T F = I^3$. In signal pro-
cessing this effect is commonly referred to as Parseval’s rela-
tion which states that, $x_i^T x_j = \tilde{x}_i^T \tilde{x}_j, \forall i, j$, given that we
assume our complex 3D-DFT basis $F$ is orthonormal. Based
on this formulation learning a linear SVM in the spatial or
Fourier domain should be identical. A linear SVM clas-
sification decision is made for an unlabeled test observation $x^*$ by,
$w^T x^* \leq b$, where $w$ is the vector normal to the separating hy-
perplane and $b$ is the bias. Both $w$ and $b$ are estimated so that
they minimize the structural risk of a train-set, thus avoiding

3 It should be noted that in many practical formulations of a 3D-
DFT $F^T F = cI$, where $c$ is a constant. Typically, $c = N$ where $N$ is
the dimensionality of the feature space. In these circumstances it is trivial
to show that an SVM should still be invariant to this scalar scaling given that
the penalty term $C$ is suitably adjusted.
the possibility of over-fitting the training data. Now by expanding $w$, we can write it as: $w = \sum_{i=1}^{N} \alpha_i y_i \hat{x}_i$, and when learning is done in the Fourier domain as, $\hat{w} = \sum_{i=1}^{N} \alpha_i y_i \hat{x}_i$. On observing closely, it can be seen that $w$ can be considered as a linear filter containing the class discriminative information. Testing can be considered as the convolution of $\hat{w}$ with a test sample $\hat{x}$ and compare the response against $b$.

### 4.2. Training with Complex Vectors

One problem, however, with our proposed approach to learning a discriminative filter using SVM is that learning has to occur in the Fourier rather than the spatial domain. This means that an SVM has to be learnt using complex (real and imaginary) vectors rather than just real vectors obtained from the spatial image domain. At first glance learning an SVM with complex Fourier vectors may seem problematic and require SVM software specifically for learning in the Fourier domain as: (i) in general the inner product between two complex vectors is itself a complex number, and (ii) most existing SVM packages (e.g., LibSVM) can handle only real vectors. Fortunately, the first problem can be automatically circumvented through Parseval’s relation which guarantees that the inner product in the Fourier domain must also be identical, according to the dual of the SVM objective function.

Based on this equivalence one can replace any Fourier vector, equivalently, with a $2N$ dimensional real vector where the real and imaginary components have been concatenated into a single vector. Since the inner products are all real, then the inner product in the Fourier domain is equivalent to the inner product in the spatial domain. Since the spatial images are all real, the inner product in the Fourier domain must also be real. The second problem can also be easily circumvented through the realization that for any two Fourier complex vectors $\hat{x}_i$ and $\hat{x}_j$ derived from spatial signals/images $x_i$ and $x_j$ respectively the following equivalence holds,

$$\hat{x}_i^T \hat{x}_j = \begin{bmatrix} \text{Re}(\hat{x}_i) \\ \text{Im}(\hat{x}_i) \end{bmatrix}^T \begin{bmatrix} \text{Re}(\hat{x}_j) \\ \text{Im}(\hat{x}_j) \end{bmatrix}. \tag{6}$$

A proof of this equivalence can be found in [14]. Based on this equivalence one can replace any $N$ dimensional complex Fourier vector, equivalently, with a $2N$ dimensional real vector where the real and imaginary components have been concatenated into a single vector. Since the inner products will be identical, according to the dual of the SVM objective function, the estimated support weights should be identical. This equivalence greatly simplifies the learning of the linear SVM as we can now leverage existing software packages for learning SVMs that are only designed to handle real vectors.

### 4.3. Experiments

Discriminative filters are tested on ‘Weizman’ data set. The training and testing is done in a leave one out setting by learning a multiclass SVM using all the samples except the test sample. We performed three separate experiments using normalized pixel intensity, edge magnitude and temporal derivatives as input features. We did not use any features based on optical flow in this case as their performance was poor for MACH filter setting (see [1]). For comparison purposes number of total videos, number of frames and frame sizes were kept exactly the same as were used for testing MACH filters. Results of the experiments are summarized in Figure 5 where we are showing the mean recognition accuracy. Using normalized intensities we obtained mean recognition accuracy of 78.88% which is 17% higher than the recognition accuracy of the best performing MACH filter (with $\alpha = .9$, see Fig. 1) for this feature type. Similarly, using gradient magnitude discriminative filter based approach obtained accuracy of 79.12% which is an improvement of 6% over the MACH filter corresponding to edge magnitude. We observed the same trend for the temporal derivatives, where discriminative filters showed an improvement of 3% over the MACH filter based recognition. This improvement over the MACH filter based detection shows that comparable or better performance can be obtained by using simple discriminative filters.

### 5. Conclusion

Following are the two outcomes of the study: 1) we demonstrate empirically that identical performance can be attained to the MACH filter by simply taking the average of the same action specific training examples. Second, we characterize theoretically and empirically under what circumstances a MACH filter would become equivalent to the average of the action specific training examples. Based on this characterization, we offer an alternative type of filter, based on a discriminative paradigm, that circumvent the inherent limitations of correlation filters for action recognition and demonstrate improved action recognition performance.

### References


[3] E. Shechtman et al., Space-Time Behavior Based Correlation OR How to tell if two underlying motion fields are similar without computing them?, TPAMI, 29(11), 2007.


