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. Additional, 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, obtains 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 $\mathbf{h}$ that minimizes the Average Similarity Measure (ASM),

$$\text{ASM} = \frac{1}{N}\sum_{i=1}^{N}\sum_{\boldsymbol{\tau}} ||\mathbf{h}^T\mathbf{x}_i(\boldsymbol{\tau})||^2 \tag{1}$$

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 contraints/objectives in Section 2.1. Where $\mathbf{x}_i$ is the ith vectorized training video example where $\boldsymbol{\tau} = [\Delta x, \Delta y, \Delta t]^T$ is the circular operator in space and time applied to the training examples. A drawback with this space-time representation of ASM is that is requires the explicit computation of all possible circular shifts $\boldsymbol{\tau}$ across all N training examples. Instead, correlation filters can attempt to solve for $\mathbf{h}$ with respect to the ASM in a more computational

1051-4651/10 $26.00 © 2010 IEEE
DOI 10.1109/ICPR.2010.639

tractable manner,ASM $= \hat{\mathbf{h}}^T \mathbf{D} \hat{\mathbf{h}}$ where $\hat{\mathbf{h}}$ and $\hat{\mathbf{x}}_i$[1] are the vectorized complex 3D discrete Fourier transforms (DFT) [2] of the vectorized space-time volumes $\mathbf{h}$ and $\mathbf{x}_i(\mathbf{0})$ respectively. Where $\mathbf{D}$ commonly referred to as the average spectral variance is defined as[2], $\mathbf{D} = \frac{1}{N}\sum_{i=1}^{N} \text{diag}(\hat{\mathbf{x}}_i)^T \text{diag}(\hat{\mathbf{x}}_i)$. Learning $\mathbf{h}$ in the frequency domain (i.e. $\hat{\mathbf{h}}$) 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 $\mathbf{h}$ while still trying to minimize the ASM (i.e., minimize the average correlation energy for $\mathbf{h}$ across all circular shifts). For example, Maximum Average Correlation Height (MACH) filters attempt to maximize the Average Correlation Height (ACH),

$$\text{ACH} = \frac{1}{N}\sum_{n=1}^{N} ||\mathbf{h}^T \mathbf{x}_i^T(\mathbf{0})||^2 \tag{2}$$

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 $\mathbf{x}_i(\mathbf{0})$ where as the ASM is a measure of correlation energy across all circular shifts $\boldsymbol{\tau}$ (denoted by $\mathbf{x}_i(\boldsymbol{\tau})$).

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 $\mathbf{C}$, and the intra-class spectral variance, denoted by $\mathbf{S}$, to $\mathbf{D}$. Where, $\mathbf{S} = \frac{1}{N}\sum_{i=1}^{N} \text{diag}(\hat{\mathbf{x}}_i - \hat{\mathbf{m}})^T \text{diag}(\hat{\mathbf{x}}_i - \hat{\mathbf{m}})$, and $\mathbf{C} = \mathbf{I}$. Including these additional terms results in finding the filter $\hat{\mathbf{h}}$ that minimizes the following objective function, $J(\mathbf{h}) = \frac{\hat{\mathbf{h}}^T \hat{\mathbf{m}}\hat{\mathbf{m}}^T \hat{\mathbf{h}}}{\hat{\mathbf{h}}^T(\alpha\mathbf{S}+\beta\mathbf{D}+\gamma\mathbf{C})\hat{\mathbf{h}}}$, which is referred to in correlation filter literature [1] as the *optimal tradeoff MACH filter*. The weights α, β and γ 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 $\mathbf{D}$ has a very minimal effect on action classification performance. For the purposes of this paper, the final optimal tradefoff MACH filter objective function can be found by simplified to minimizing,

$$J(\mathbf{h}) = \frac{\hat{\mathbf{h}}^T \hat{\mathbf{m}}\hat{\mathbf{m}}^T \hat{\mathbf{h}}}{\hat{\mathbf{h}}^T((1-\alpha)\mathbf{S} + \alpha\mathbf{C})\hat{\mathbf{h}}} \tag{3}$$

[1]Please note that throughout this paper that the notation ˆ applied to any vector denotes the 3D-DFT of a vectorized 3D video volume such that $\hat{\mathbf{x}} \leftarrow \mathbf{F}\mathbf{x}$, where $\mathbf{F}$ is the $N \times N$ matrix of complex basis vectors for mapping to the 3D Fourier domain for any N dimensional vectorized video volume.

[2]In Equation 2 we are taking advantage of the fact that $\text{diag}(\hat{\mathbf{x}})\mathbf{h} = \hat{\mathbf{x}} \cdot \hat{\mathbf{h}}$ where diag() 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 ^{T} on a complex vector or matrix in this paper additionally takes the complex conjugate in a similar fashion to the Hermitian adjoint [2].

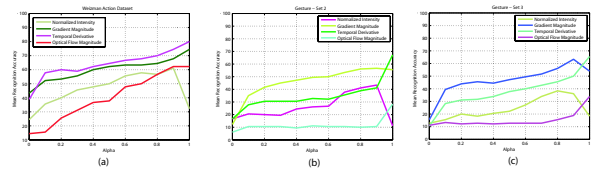

Figure 1. Graphs summarizing performance on the task of action classificaiton as a function of α.

whose solution is the dominant eigenvectors of $((1-\alpha)\mathbf{S} + \alpha\mathbf{C})^{-1}\hat{\mathbf{m}}\hat{\mathbf{m}}^T$ and $\alpha \in \{0, 1\}$.

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:$PSR = \frac{peak-\mu}{\sigma}$, where μ and σ are the mean and standard deviation of the correlation values in some neighborhood of the peak. time.

3. Varying α

In this section, we study the effect of varying α 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 α was varied from 0 to 1. As α increased the contribution from the spectral variance term, $\mathbf{S}$, becomes less and less important. For each value of α, 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 videos are preprocessed in the following manner for the purpose of learning a filter. A bounding box of size 80×80 pixel is initialized around the actor by approximately keeping him/her at the center. Thirty frames are extracted for each action resulting in a $80 \times 80 \times 30$ volume. This volume is then used to compute various features including edge magnitude, temporal derivative, and optical flow. For testing, we applied the MACH filter learned for each class on a test video, and label corresponding to the MACH filter that returns maximum PSR value is assigned.

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 α. 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 α results in better performance in terms of recognition. The best results are obtained when $\alpha = 1$ 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 α is main tained across different types of features.

3.2. Average Filter

From Eq. (3), it is easy to show that when $\alpha = 1$ thi is equivalent to learning an average filter since we are nov finding the eigenvectors of $(\mathbf{C})^{-1}\hat{\mathbf{m}}\hat{\mathbf{m}}^T$ where $\mathbf{C} = \mathbf{I}$. Th results in the previous section indicate that in nearly all cir cumstances the average filter (i.e., $\alpha = 1$ outperformed al other versions of the MACH filter). This result can be under stood by realizing that the spectral variance $\mathbf{S}$ is at the heart of learning a distortion tolerant MACH filter. The frequencies where the magnitude of $\mathbf{S}$ 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 $\mathbf{S}$ is used as it attenuates fre quencies where samples of an action vary most in space an 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, fee 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 various values of α 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 α is increased.

4. Discriminative Filters

We offer an alternative type of filter, based on a discriminative paradigm, that circumvent the inherent limitations of correlation 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 classification tasks [13]. Given a set of training example pairs $(\mathbf{x}_i, y_i)$, $i = 1, \ldots, l$, $\mathbf{x}_i \in \Re^N$, $y_i \in \{+1, -1\}$, a linear SVM attempts to find the solution to the following unconstrained optimization problem, $\min_{\mathbf{w}} \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{l} \max(1 - y_i\mathbf{w}^T\mathbf{x}_i, 0)^2$, where C is a penalty parameter and the bias b is accounted for in $\mathbf{w} \leftarrow [\mathbf{w}^T, b]$ by $\mathbf{x} \leftarrow [\mathbf{x}^T, 1]$. The parameters of the hyper-plane are estimated by maximizing the *margin term* (i.e., minimizing $\mathbf{w}^T\mathbf{w}$) and

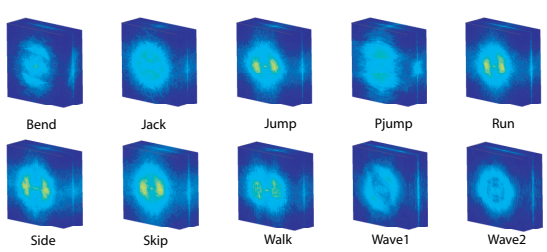

Figure 2. Log magnitude of spectral variance volumes of 10 action classes of the Weizman action data set.

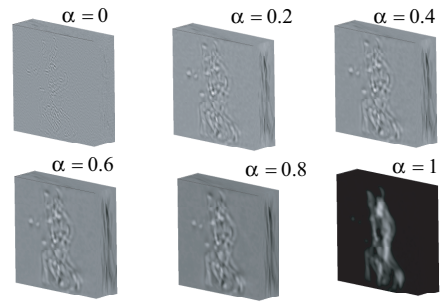

Figure 3. MACH filters learned from the temporal derivatives of the input videos for action class 'run'. The filters show effect of varying the emphasis of spectral variance on the final filter. Filter are learned by varying values of α in Eq. (3).

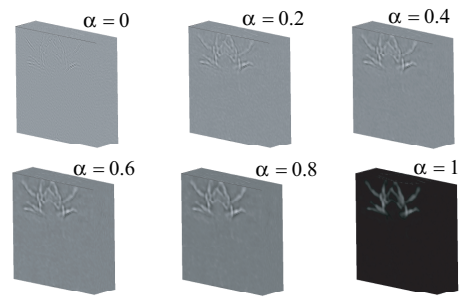

Figure 4. MACH filters learned from the temporal derivatives of the input videos for action class 'two hand wave'. The filters show effect of varying the emphasis of spectral variance on the final filter. Filter are learned by varying values of α in Eq. (3).

minimizing an upper bound (i.e., the *hinge error function*) on 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,

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

$$\text{subject to} \quad \boldsymbol{\alpha}^T\mathbf{y} = 0 \quad (5)$$

As pointed out by [15] it is easy to see that the dual of Equation 4.1 is rotation invariant. For example if all $\mathbf{x}_i$ were replaced by $\mathbf{A}\mathbf{x}_i$, where $\mathbf{A} \in \Re^{N\times N}$, $\mathbf{A}^T\mathbf{A} = \mathbf{I}$, then the solution remains the same. Interestingly, one can view the application of a 3D-DFT as multiplication by a complex orthonormal basis $\hat{\mathbf{x}}_i = \mathbf{F}\mathbf{x}_i$ where $\mathbf{F}^T\mathbf{F} = \mathbf{I}$[3]. In signal processing this effect is commonly referred to as *Parseval's relation* which states that, $\mathbf{x}_i^T\mathbf{x}_j = \hat{\mathbf{x}}_i^T\hat{\mathbf{x}}_j \forall i, j$, given that we assume our complex 3D-DFT basis $\mathbf{F}$ is orthonormal. Based on this formulation learning a linear SVM in the spatial or Fourier domain should be identical. A linear SVM classification decision is made for an unlabeled test observation x^* by, $\mathbf{w}^T\mathbf{x}^* \lessgtr b$, where $\mathbf{w}$ is the vector normal to the separating hyperplane and b is the bias. Both $\mathbf{w}$ 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 $\mathbf{F}^T\mathbf{F} = c\mathbf{I}$, 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.

Dataset	Normalized Intensity	Edge Magnitude	Temporal Derivative
Weizman	78.88%	79.12%	80%

Figure 5. Mean recognition accuracy.

the possibility of over-fitting the training data. Now by expanding $\mathbf{w}$, we can write it as: $\mathbf{w} = \sum_{i=1}^{N} \alpha_i \mathbf{y}_i \mathbf{x}_i$,, and when learning is done in the Fourier domain as, $\hat{\mathbf{w}} = \sum_{i=1}^{N} \alpha_i \mathbf{y}_i \hat{\mathbf{x}}_i$. On observing closely, it can be seen that $\mathbf{w}$ can be considered as a linear filter containing the class discriminative information. Testing can be considered as the convolution of $\hat{\mathbf{w}}$ with a test sample $\hat{\mathbf{x}}$ and compare the response gainst 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 is equivalent to the inner product in the spatial domain. Since the spatial images are all real, then 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 $\hat{\mathbf{x}}_i$ and $\hat{\mathbf{x}}_j$ derived from spatial signals/images $\mathbf{x}_i$ and $\mathbf{x}_j$ respectively the following equivalence holds,

$$\hat{\mathbf{x}}_i^T \hat{\mathbf{x}}_j = \begin{bmatrix} \mathcal{R}e\{\hat{\mathbf{x}}_i\} \\ \mathcal{I}m\{\hat{\mathbf{x}}_i\} \end{bmatrix}^T \begin{bmatrix} \mathcal{R}e\{\hat{\mathbf{x}}_j\} \\ \mathcal{I}m\{\hat{\mathbf{x}}_j\} \end{bmatrix}. \quad (6)$$

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

[1] B. V. K. Vijaya Kumar et al., Correlation Pattern Recognition, Cambridge University Press, 2005. 1, 2

[2] A. V. Oppenheim et al., Signals & Systems, Prentice Hall, 1996. 2

[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. 1

[4] M. D. Rodriguez et al., Action MACH a spatio-temporal Maximum Average Correlation Height filter for action recognition, CVPR, 2008. 1, 2

[5] T. Darrell et al., Classifying Hand Gestures with a View-Based Distributed Representation, NIPS, 1993. 1

[6] A. Yilmaz et al., Actions Sketch: A Novel Action Representation, IEEE CVPR, 2005. 1

[7] M. Blank et al., Actions as Space-Time Shapes, IEEE ICCV, 2005. 1

[8] I. Laptev et al., Learning Realistic Human Actions from Movies, IEEE CVPR, 2008. 1

[9] J. C. Niebles et al., Unsupervised Learning of Human Action Categories Using Spatial-Temporal Words, BMVC, 2006. 1

[10] P. Scovanner et al., A 3-Dimensional SIFT Descriptor and its Application to Action Recognition, ACM Multimedia, 2007. 1

[11] A. Efros et al., Recognizing Action at a Distance, ICCV, 2003. 1

[12] J. Little et al., Recognizing People by Their Gait: The Shape of Motion, Journal of Computer Vision Research, 1998. 1

[13] B. E. Boser, et al., A training algorithm for optimal margin classifiers, 5th COLT, 1992. 3

[14] http://www.cs.cmu.edu/ saada/proof.pdf 4

[15] P. K. Shivaswamy et al., Relative margin machines, Neural Information Processing Systems (NIPS), 2008.

3