Average of Synthetic Exact Filters

David S. Bolme Bruce A. Draper J. Ross Beveridge
Colorado State University
Fort Collins, CO 80521, USA
bolme@cs.colostate.edu

Abstract

This paper introduces a class of correlation filters called Average of Synthetic Exact Filters (ASEF). For ASEF, the correlation output is completely specified for each training image. This is in marked contrast to prior methods such as Synthetic Discriminant Functions (SDFs) which only specify a single output value per training image. Advantages of ASEF training include: insenitivity to over-fitting, greater flexibility with regard to training images, and more robust behavior in the presence of structured backgrounds. The theory and design of ASEF filters is presented using eye localization on the FERET database as an example task. ASEF is compared to other popular correlation filters including SDF, MACE, OTF, and UMACE, and with other eye localization methods including Gabor Jets and the OpenCV Cascade Classifier. ASEF is shown to outperform all these methods, locating the eye to within the radius of the iris approximately 98.5% of the time.

1. Introduction

A common way to detect patterns in images is through correlation with an example template [4]. The simplicity and efficiency of this approach continually draws researcher attention despite many known weaknesses, and a steady progression of advances has continued to breath new life into this well established technique.

One commonly recognized weakness of simple template matching is that while the response to a perfect example of the template pattern will always be high, the relative strength of responses to alternative patterns can be unpredictable. A family of correlation filters has been developed that endeavors to overcome this weakness by suppressing responses to near-miss or distractor patterns, while preserving strong responses to the target pattern. The differences among these filters lie in how they are constructed from training samples. Examples include Synthetic Discriminant Functions (SDF)[5], Minimum Variance Synthetic Discriminant Functions (MVSDF) filters[15], Minimum Average Correlation Energy (MACE) filters[8], Optimal Tradeoff Filters (OTF)[12] and Unconstrained MACE (UMACE)[14].

While these approaches greatly extend the performance range of correlation filters, there is still further room for improvement. Specifically, we propose a new class of filters called Average of Synthetic Exact Filters (ASEF) that differ from these prior methods in two important respects. First, an entire correlation response surface is specified for each training instance during filter construction. Second, the resulting filters, one per training image, are then simply averaged. The resulting filters are less susceptible to over-fitting the training data than other methods, and can therefore be trained over larger and more inclusive training sets. As a result, they out perform previous methods.

On the task of finding eyes in face images, ASEF out performs all of the filter types mentioned above, as well as Haar-based Cascade Classifiers [17] and Gabor Jet-based methods[19]. This new family of correlation filters thus shows state-of-the-art performance on the task of eye finding and we anticipate similar performance gains on many challenging object detection tasks.

2. Background

Accurate registration of face images is an important first step in face recognition, and one common way of establishing face registration is by finding eyes. As noted below, correlation filters have been applied to eye finding before [3]. This section reviews the eye finding task and past work on synthetic correlation filters. Emphasis is placed on algorithms that appear in the direct comparisons with ASEF in Section 4.

2.1. Eye Finding/Localization

Generally, eye finding algorithms return the pixel coordinates of the center of the left and right eyes in frontal, or near frontal, face images. To be counted as a success, the algorithm must return the true eye location to within some tolerance, typically measured as a fraction of the interocular

978-1-4244-3991-1/09/$25.00 ©2009 IEEE

distance, i.e. the distance between the centers of the eyes. As is common, results in Section 4 will report percentage of eyes correctly located over a range of interocular distances.

Two flavors of eye localization will be considered here. The first presumes accurate prior knowledge, in essence assuming the true eye locations are already known to within a modest number of pixels. This obviously easier task arises in practice when face detection accurately locates the whole face. Our results suggest this task can be equally well solved using several methods. The more difficult task is to accurately locate the left and right eye on a face given no prior constraints, and it is on this task that the superior performance of ASEF filters becomes apparent.

While there has been a tremendous amount of prior work on eye finding, we will mention just a few either because they also used synthetic correlation filters or because they are particularly well known. In terms of synthetic correlation filters and eye finding, the only prior work we know of is by Brunelli and Poggio in 1997. They applied SDF and Least Squares Synthetic Discriminant Functions (LSSDF) correlation filters to eye detection [3]. Although this work showed promise, the results are not competitive with more recent work.

A well known face detection algorithm, the Viola and Jones cascade classifier[17], has been adopted by many researchers for eye detection: see [13], [6], and [18]. One such system, designed by Castrillon-Santana, *et al.* [13], uses a cascade classifier in conjunction with skin tone analysis. In this work we have adopted the cascade detector from their paper to produce our own cascade based eye locator. On the easier task where an approximate eye location is known, the cascade classifier performs well. However, when the approximate location constraint is removed, the cascade classifier produces many false detections and consequently performs poorly.

Gabor jets have also been studied extensively as an eye localization technique. As part of the Elastic Bunch Graph Matching (EBGM) face recognition algorithm proposed by Wiskott *et al.* [19], Gabor jets were used to locate many fiducial points on faces including the eyes. When we compare ASEF filters to a Gabor jet based eye detector based on Wiskott's algorithm, we find the Gabor jet algorithm is at least 20 times as computationally demanding and is only applicable to the easier problem where the eye location is approximately known.

2.2. Correlation Filters

Correlation with an example template works well if the appearance of the target does not change significantly from image to image. Unfortunately, in most domains the appearance of the target does change across images, due to variations among target instances and changes in imaging condition (e.g. lighting, pose). There is also the threat that a template may respond to visually similar non-target objects. The result is that templates are often poor discriminators in many object detection tasks.

A large famly of correlation filters have been developed that improve the response to a variety of input stimuli. Chronologically, SDF [5] filters were introduced first; they respond well to positive training images while suppressing responses to negative training examples. Next, MVSDF[15] filters, MACE[8] and then OTF[12] filters were introduced. These refine aspects of the filter design to improve performance relative to noise and spatial resolution of response. All four of these methods are similar in the way that they are trained. Specifically, they all require a zero/one (target/non-target) constraint on each training image. It has been found that these hard constraints are unnecessary and can even be detrimental for producing robust correlation filters[7]. Unconstrained correlation filters such as MACH[7] and UMACE[14] relax these constraint and instead favor high correlation responses on the average training image.

Two other prior approaches to the design of synthetic discriminant filters deserve mention because they address the entire correlation surface. Minimum Squared Error Synthetic Discriminant Functions (MSESDF) [16] allows the correlation surface to have an arbitrary response shape. This type of filter was not tested in this work because there is no canonical implementation and it has been shown that when the desired output shape is selected to minimize the non-centered pixels this filter is equivalent to MACE. The second approach, Distance Classifier Correlation Filters (DCCF) [9], produces output shapes that optimally discriminate between classes of objects. While this is useful for discriminating between objects, it has little use when accurately locating objects. Both these methods still require the training images to be centered on the target.

In this paper we will compare ASEF to two common optimal tradeoff filters that are described in [14]. The first type of filter, OTF, is based on the SDF formulation which imposes hard constraints on the output of the filter.

For these filters, the desired output of the filter is captured in a variable u_i. For positive training examples, $u_i = 1$ and for negative examples $u_i = 0$. In this work only positive examples are used. The corresponding constraint for a single training image is:

$$u_i = \mathbf{h}^\top \mathbf{x}_i \tag{1}$$

Because there are fewer constraints than pixels in the filter there are multiple filters that will satisfy these constraints. To produce a single filter SDFs impose additional constraints by requiring the filter to be a linear combination of the training images, while MACE requires the filter to minimize the average output energy over the training set. Under this process the filter h is defined as:

$$\mathbf{h} = \mathbf{D'}^{-1}\mathbf{X}(\mathbf{X}^\top \mathbf{D'}^{-1}\mathbf{X})^{-1}\mathbf{u} \tag{2}$$

The distinction between SDF, MVSDF, MACE, and OTF filters lies entirely in how $\mathbf{D}'$ is defined.

A MVSDF suppresses frequencies corresponding to noise. To do this we define $\mathbf{D}' = \mathbf{C}$ where $\mathbf{C}$ is a diagonal matrix such that each element of the diagonal corresponds to the power spectrum of the noise. Like [14], we will assume white noise in which C becomes the identity matrix ($\mathbf{C} = \mathbf{I}$). MVSDF filters typically emphasize lower frequencies which suppresses noise, but this also has the effect of producing smoother peaks that are more difficult to detect.

MACE attempts to produces sharp detectable peaks by minimizing the average correlation plane energy for the training set. To do this we define $\mathbf{D}' = \mathbf{D}$ where $\mathbf{D}$ is a diagonal matrix containing the average power spectrum of the training images. MACE filters typically emphasize high frequencies. This produces sharp peaks, but also makes the filter much more sensitive to noise.

Finally, OTF finds an optimal balance between the properties of MVSDF and MACE.

$$\mathbf{D}' = (\mathbf{D}\alpha + \mathbf{C}\sqrt{1-\alpha^2})^{-1} \quad (3)$$

This introduces the parameter α which is used to tune the filter between the noise tolerance of MVSDF and the sharp easily detected peaks of MACE. In the case that $\alpha = 0$, $\mathbf{D}'$ becomes the identity matrix and the filter is equivalent to the SDF filter. If $\alpha = 1$, $\mathbf{D}' = \mathbf{D}$ and the OTF filter is equivalent to MACE. Therefore, the process of optimizing alpha discussed in Section 4 also considers the two special cases of SDF ($\alpha = 0.0$) and MACE ($\alpha = 1.0$).

The second type of optimal tradeoff of filter is an unconstrained filter called UMACE. Instead of requiring the filter to satisfy a set of hard constraints on the correlation output, UMACE only requires a high average response to the training examples. The resulting filter is defined as:

$$\mathbf{h} = \mathbf{D}'^{-1}\mathbf{m} \quad (4)$$

where $\mathbf{m}$ is the average of the columns of $\mathbf{X}$ (or the average training image) and $\mathbf{D}'$ is the same as defined in Equation 3. Here the filter for $\alpha = 0.0$ becomes the average training images and for $\alpha = 1.0$ has similarities to MACE in that it produces sharp correlation peaks.

The MVSDF, MACE, and OTF filters mentioned above are all based on similar assumptions, and have many of the same issues. Each training image is given a single "synthetic correlation value", which is the value the filter should return when the filter is centered upon the image. The result is too few constraints relative to the degrees of freedom in the filter, leading to over-fitting of the training data. While unconstrained filters such as UMACE eliminate this over-fitting, they still share many other problems with these filters; in particular the filters do not specify the response at any other location in the training tile.

3. ASEF Correlation Filters

ASEF filters differ from prior correlation filters in that the convolution theorem is exploited to greatly simplify the mapping between the input training image and the output correlation plane. In the Fourier domain the correlation operation becomes a simple element-wise multiplication, and therefore each corresponding set of Fourier coefficients can be processed independently. The resulting computations also naturally account for translational shifts in the spatial domain. As a result the entire correlation output can be specified for each training image.

The first major difference between the filters discussed above and ASEF filters is that ASEF filters are over constrained. Where SDF only specifies a single "synthetic correlation value" per training image, ASEF filters are trained using response images that specify a desired response at every location in each training image. This response typically is a bright peak centered on the target object of interest.

One consequence of completely specifying the correlation output is a perfect balance between constraints and degrees of freedom for each training image, and therefore a complete "exact filter" is determined for every training image. Over-fitting is avoided by averaging the filters defined from each of the N training images. The UMACE filter also averages to avoid over-fitting, but there the similarity ends, since UMACE averages the training images while ASEF averages a set of exact filters.

Finally, ASEF filters provide much more freedom when selecting training images and when specifying the synthetic output. A benefit is that the training images need not be centered on the target. For each training image, we specify the desired filter output and may place the peak wherever the target appears. Because the correlation peak moves in lockstep with the targets in the training images, all the exact filters are consequently registered by inverting the correlation process. This increases training flexibility, allowing us to customize the desired response for each training image. For example, training images may have multiple targets per training image as long as the synthetic output contains multiple corresponding peaks.

3.1. Definition of an ASEF Filter

Figure 1 illustrates the process of constructing an ASEF filter. Note first the training pairs f_i, g_i consist of a training image and associated desired correlation output. The correlation image g_i is synthetically generated with a bright peak at the center of the target, in our case the left eye, and small values everywhere else. Specifically, we define g_i to be a two dimensional Gaussian centered at the target location (x_i, y_i) and with radius σ:

$$g_i(x, y) = e^{-\frac{(x-x_i)^2+(y-y_i)^2}{\sigma^2}} \quad (5)$$

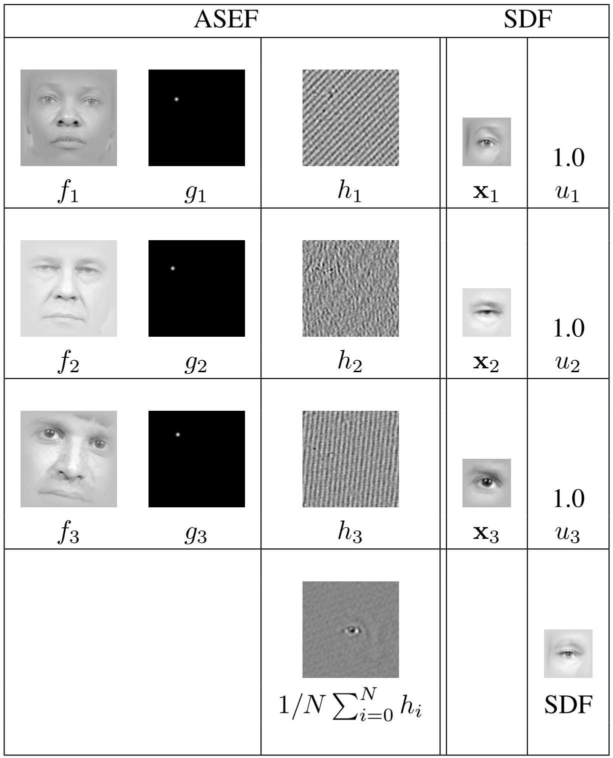


Figure 1. This figure compares ASEF training to the training for SDF. For ASEF, the image f_i is an image in the training set and g_i is the corresponding desired filter output. A correlation filter h_i is produced by in the Fourier domain that *exactly* transforms f_i to g_i. The final correlation filter is produced by taking the average of many Exact Filters. SDF and similar methods only specify one correlation value for each training image.

The role played by σ is similar to that of α in OTF: it trades off noise tolerance against peak sharpness. [1]

By the Convolution Theorem, we know that convolution in the spatial domain becomes element-wise multiplication in the Fourier domain:

$$g(x,y) = (f \otimes h)(x,y) = \mathcal{F}^{-1}(F(\omega,\nu)H(\omega,\nu)) \qquad (6)$$

This relationship forms the basis for finding synthetic exact filters, where f is the image, h is the filter, and g is the correlation output in the spatial and corresponding capital letters (F, H, and G) indicate their respective 2D Fourier transforms.

To solve for the exact filter, first note the correlation is computed by simply substituting the complex conjugate of H into Equation 6.

$$G(\omega,\nu) \quad = \quad F(\omega,\nu)H^*(\omega,\nu) \qquad (7)$$

Next, solve for the exact filter:

$$H_i^*(\omega,\nu) \quad = \quad \frac{G_i(\omega,\nu)}{F_i(\omega,\nu)} \qquad (8)$$

where the division is an element-wise division between the transformed target output G_i and the transformed training image F_i. This type of computation is not entirely new. Similar computations are used to perform deconvolutions or to produce inverse filters[11]. However, we have not yet seen a synthetic correlation plane used for this purpose in the main stream literature.

One can see from Figure 1 that the exact filters h1, h2, and h3 do not appear to have a structure that would respond well to an eye but instead are specific to each training image. To produce a filter that generalizes across the entire training set, we compute the average of multiple exact filters. Averaging emphasizes features common across training examples while suppressing idiosyncratic features of single training instances. This is visually evident in the final ASEF shown in the bottom row of Figure 1.

A deeper motivation for averaging may be found in aggregation theory also known as bagging. In particular, the exact filter can be thought of as a weak classifier that performs perfectly on a single training image. As shown by Breiman [2], a summation of a set of weak classifier's outperforms all the component classifiers and, more importantly, if the weak classifier's are unbiased, thier summation converges upon a classifier with zero variance error.

Because the Fourier transform is a linear operation, the average can be computed in either the Fourier or the spatial domain.

$$H_\mu^*(\omega,\nu) \quad = \quad \frac{1}{N}\sum_{i=1}^{N} H_i^*(\omega,\nu) \qquad (9)$$

$$h_\mu(x,y) \quad = \quad \frac{1}{N}\sum_{i=1}^{N} h_i(x,y) \qquad (10)$$

where H_μ or h_μ are the final ASEF filters. Note that if computed in the spatial domain the Exact filters can be cropped before averaging which allows ASEF filters to be constructed from training images of different size.

We have found that ASEF filters perform best when trained on as many images as possible. In this paper we have augmented the training set by introducing random similarity transforms as part of simulating the face detection process. In general, image transformations that introduce small variations in rotation, scale, and translation are beneficial to producing robust ASEF filters because they expose the filter to a greater variety of images. This family of transforms also focuses the filter on regions near the peaks and therefore producing a filter that emphasizes the image data near the target object.

[1] The synthetic output need not be peaks. For example an edge filter could be learned by specifying a bright response along edges of interest.

4. Experimental Evaluation

All experiments used the FERET dataset [10]. This dataset contains 3,368 images of 1,204 people, with manually selected eye coordinates for each image. For these experiments, the FERET data set was randomly partitioned by subject into two sets of 602 people and 1699 images each. One of these sets was further partitioned by image into a training set with 1024 images and a validation set with 675 images. The training and validation sets were used to tune the algorithms. The other set of 602 people was sequestered during training and used as a testing set.

Faces were initially found in all images using the OpenCV face detector. This detector places the eyes very close to their true location most of the time, which made eye detection too easy for adequate testing of alternative eye finding methods. To make the eye localization problem more difficult, face detection is simulated by first aligning the faces to produce 128×128 images with the eyes located at $(32.0, 40.0)$ and $(96.0, 40.0)$, and then applying a random similarity transform that rotates by up to $\pm\pi/16$, scales by up to 1.0 ± 0.1, and translates by up to ± 4.0 pixels. Each of the initial $1,024$ training images was randomly perturbed 8 times yielding $8,192$ training images.

For the correlation filters tested in this paper, each image tile was normalized by first taking the log $(log(v+1))$ of the pixel values to reduce the effect of shadows and intense lighting, and then normalizing the values to have a mean of 0.0 and a squared sum of 1.0, to give the images a consistent intensity. Finally, a cosine window is applied to the image which reduces the frequency effects of the edge of the image when transformed by the Fast Fourier Transform (FFT). ASEF was trained on the full 128×128 image tile, while the other correlation filters were trained on 64×64 image centered on an eye. Localization is performed by correlating a testing or validation image with the left and right filters and selecting the global maximum in the correlation output.

Evaluation of the eye location algorithms tested in this paper is based on the distance from the manually selected eye coordinate, normalized by the interocular distance. For example, the left eye normalized distance is computed as follows:

$$D = \frac{||P_l - M_l||}{||M_l - M_r||} \tag{11}$$

where D is the normalized distance, P_l is the predicted eye location from the algorithm and M_l and M_r are the manually selected left and right eye coordinates. We have chosen the operating point of $D < 0.10$ as the criteria for a successful localization. This corresponds to a target that is approximately the size of the human iris. In most cases, this paper only shows the results for the left eye. Results for the right eye were similar and always corroborate conclusions drawn here. Right eye results and more analysis can be found in the supplemental material.

4.1. Experiment 1: Localization Restricted to Eye Regions

This experiment is intended to simulate an eye localization problem. In this section we compare ASEF filters to the competing correlation filters and to Cascade Classifier and Gabor Wavelet eye localization algorithms. For this experiment all the algorithms have been configured to only search for an eye within a small region surrounding the expected location of the eye. This is consistent with many real systems where the face is first located using a face detector. Typically, the eyes end up in the same regions of the face detection window.

Before running a formal test, the training and validation data where used to find an optimal configuration for each of the algorithms. These results can be found in Figure 3a.

The OTF correlation filters were trained using set sizes sampled densely from 1 to 24. Additional large set sizes were also tried to illustrate the over-fitting of this type of filter. Values for α where also sampled on a range from 0.0 to 1.0 at an interval of 0.1. The configuration that performed best on the validation set was selected for the final test.

Because ASEF and UMACE filters do not over-fit the datasets, these algorithms were trained on all 8192 training images to obtain the best generalization over the dataset. The validation set is used to select values for α for UMACE and σ (sampled at 1, 2, 3, 4, 5, and 6) for ASEF. Figure 3a also showed the effect of training on smaller data sets.

For this experiment, the correlation filters are restricted to a region within 20 pixels of the mean eye coordinate. This is required to eliminate many false alarms that appear in other parts of the face such as the “wrong” eye, the nose, and the mouth.

The cascade classifier we have chosen is the OpenCV cascade classifier trained for eyes [13]. In preliminary experiments we found that the cascade classifier on average produces 3.5 false alarms per face, of which one is often the ”wrong” eye, and the others correspond to the nose, mouth, or the center of the forehead. The detection that is closest to the mean eye coordinate is selected as the predicted eye coordinate.

The Gabor jet locator is similar to the algorithm described in [19]. The algorithm uses 256 exemplar jets. The Gabor jet algorithm performs a gradient decent search for the location of the eye starting at the mean coordinate. This search is naturally restricted to a basin of attraction near that mean coordinate and therefore will only produce localizations near that mean.

The results shown in Figure 3b indicate that all of the algorithms perform quite well when the search is restricted to the region near the eye. One of the most interesting results is that two correlation filter methods (ASEF and UMACE) perform significantly better than the two common eye localization methods. These are the two filters that do not over fit the training data and therefore train on all 8,192 training images. These results suggest that a large training set is important for accurate localization. Furthermore, sliding window approaches do not evaluate their classification functions at every pixel. To achieve real time performance, sliding windows typically use a gate. Correlation computes correlation at every pixel simultaneously and therefore should produce more accurate localizations. The filters also will not miss a target just because it was not centered properly in the sliding window.

4.2. Experiment 2: Localization Without Restrictions

This experiment looks at the more difficult problem of finding an eye when the approximate location of the eye is not known a priori. This experiment is relevant to many other vision problems, like generalized object detection, where the location of the object in the scene may be unpredictable.

The correlation filters were trained and configured using the same training sets, validation sets, and parameters as in the previous experiments. The difference between the experiments is that the search space is expanded to include the entire image.

The Gabor jet method was excluded from this experiment because there was no easy way to extend the algorithm to search the entire image. The cascade classifier was also excluded, because its high false alarm rate made selecting the correct detection without the help of prior information infeasible.

Figure 3c and 3d shows that the best ASEF filters are considerably more accurate than any of the other training methods. An investigation into this issue found that the OTF and UMACE filters where still producing strong responses for the correct eye, but were often distracted by stronger responses to other locations of the face, typically the ”wrong” eye or the nose. This result is almost expected because the left and right eyes have similar appearances.

What is surprising is that ASEF filters rarely detect the wrong eye. The correlation outputs for ASEF typically show a very high response to the correct eye, and a very low or non-existant response to the “wrong” eye. The majority of the ASEF mistakes tend to be responses to background or to unusual features of the face such as dark rimmed glasses. Interestingly, ASEF is the only method tested in this research that does not require prior knowledge of the location of the eye to achieve good performance on this problem.

We believe this is caused by two features unique to the ASEF training process. First, ASEF filters are trained on the entire face image, including the “wrong” eye, nose, mouth, etc. OTF and UMACE, on the other hand, were centered on the correct eye and therefore had no exposure to these or other distractions (see Figure 1). Second, because ASEF completely specifies the correlation output for the entire training image, it specifies both the high response for the correct eye and the low responses for the rest of the face. Every exact filter that becomes part of the “average” therefore has learned to ignore these other features. The result is that the most common distractors are rarely associated with mistakes.

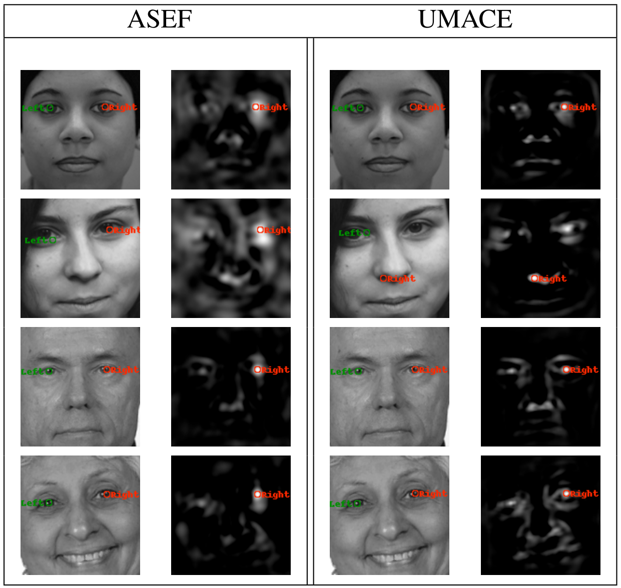
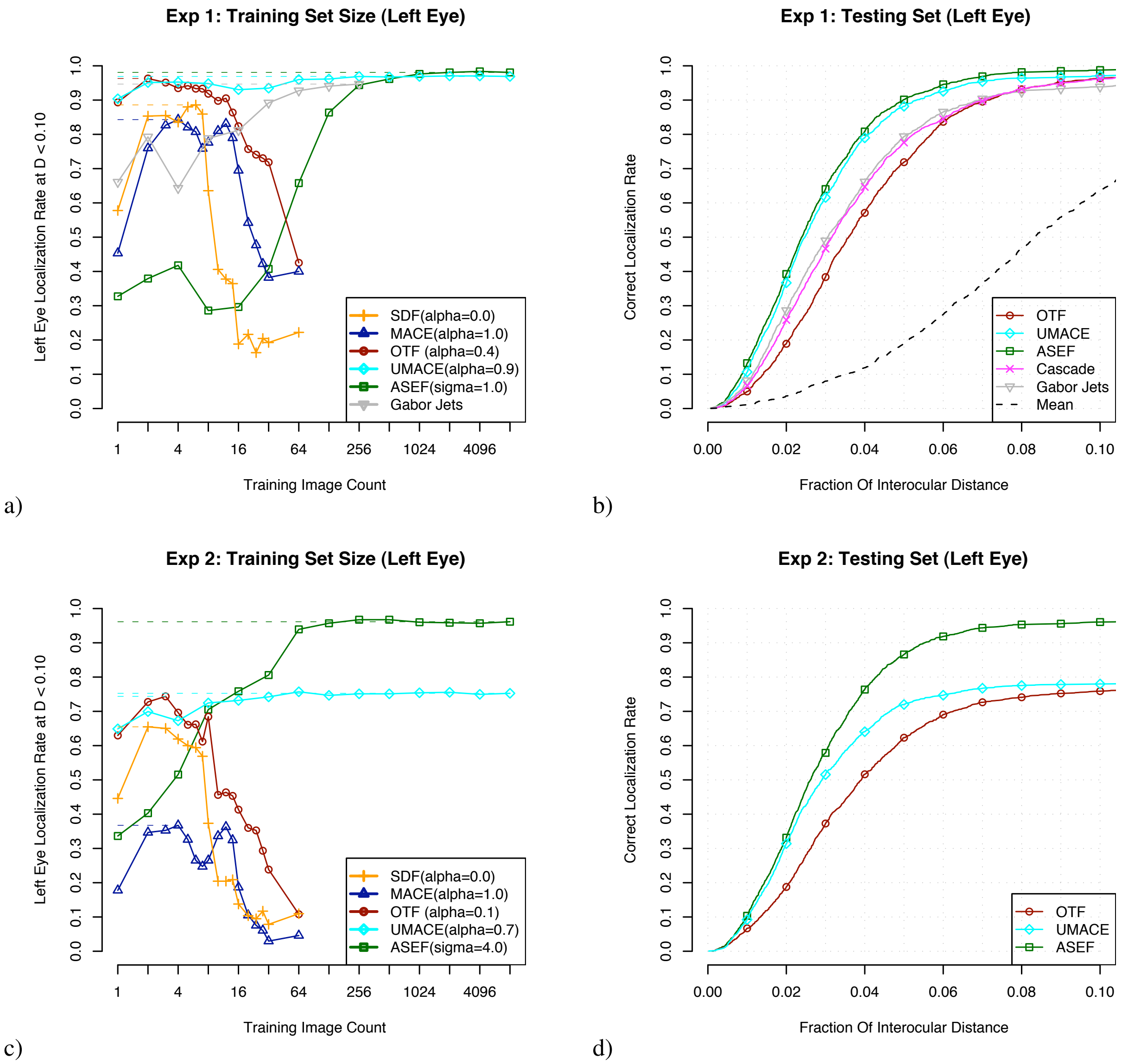


Figure 2. This figure shows the eye localization and correlation output for the best correlation filter configurations from Experiment 2. A good correlation filter should produce a small bright peak at the correct location of the right eye and a mostly dark for the rest of the image. The second row down illustrates near misses for the left eye for both filters and missed right eye for UMACE filter.

4.3. Experiment 3: Runtime Performance

One advantage of the correlation algorithm used with these filters is that the algorithm is simple and fast. As seen in the previous section, ASEF is producing much better results than the older filter based algorithms. This boost in accuracy could allow correlation filters to become fast and simple solutions to problems that were previously too difficult for correlation filter methods.

The primary performance bottle neck is the computation of the FFT. Both the left and right eye filters can be combined into one complex filter where the real part corresponds to the left eye and the imaginary part corresponds to the right eye. By pre-computing the FFT of this combined filter, eye detection can be performed using just two FFTs by using one FFT to transform the image into the Fourier domain, computing the element-wise multiplication, and then using the other FFT to compute the corre-



Exp 1: Training Set Size (Left Eye)

Exp 1: Testing Set (Left Eye)

Exp 2: Training Set Size (Left Eye)

Exp 2: Testing Set (Left Eye)

Figure 3. These plots show the results for Experiments 1 (top) and 2 (bottom). The top row shows that the eye localization problem is much easier when the algorithms use prior knowledge of the location of the eye. When this knowledge is not used ASEF clearly outperforms the other algorithms.

lation plane. We have benchmarked the FFTW library as computing 683.5 complex 2D FFTs per second at 128×128 pixels.[2] This benchmark could be improved either using specialized FFT hardware or optical correlation devices.

Correlation filters can be easily compared to the Gabor jet algorithm which requires 41 FFTs to compute the correlation of an image with the 40 complex Gabor wavelets. This means the Gabor jet algorithm requires at least 20 times as many FFTs as a correlation filter approach.

In addition, FFT implementations use the underlying hardware efficiently and the element wise multiplication and maxima detection are very fast. For this reason correlation filters may be faster than many sliding window object detection algorithms with an expensive or inefficient "classifier function". In this final experiment the correlation filter methods were benchmarked at over twice as fast as the cascade classifier.

A speed comparison can be found in Table 1. The simplicity of the correlation filters make them by far the fastest. All tests were performed in python using SciPy, OpenCV,

[2] Based on using both cores of a MacBook Pro Laptop with a 2.4Ghz Intel Core 2 Duo processor and 2Gb Ram.

and PyVision[1]. Most computation for the correlation filters and cascade classifier were performed by calling C or fortran implementations. A significant amount of the Gabor computation is performed in python so performance increases should be realized by converting those functions to a compiled language.

Table 1. These are the times to process all 1699 images in the testing set for each of the three classes of algorithms. All correlation filters are should to have about the same performance of which best is shown here.

Experiment	Filters	Gabor	Cascade	Mean
Testing Time	52.21s	4859.53s	142.19s	19.53s

5. Conclusions

This paper introduces Average of Synthetic Exact Filters (ASEF) as a method of constructing correlation filters for detecting objects. Eye detection experiments show ASEF filters to be superior to previously proposed synthetic correlation filters as well as to two other commonly proposed eye detection algorithms.

ASEF filters differ from previous synthetic correlation filters in two important ways. First, for every training image ASEF creates an exact filter that recreates the entire desired correlation surface. Since the background may involve distracting patterns, this improves the filter's discrimination ability. Second, the final filter is the average of the exact filters for all the training images. This averaging avoids over-fitting by emphasizing common features shared across images.

We believe that ASEF will perform well on many similar tasks. We have already extended this work to locate more points on faces, such as the nose, eye brows, and mouth with excellent results. In addition, ASEF has performed well at detecting faces and locating pupils in images from an iris sensor. In the future we hope to test ASEF in a variety problems including face detection, face verification, automatic target recognition, and medical image registration.

References

[1] D. S. Bolme. Pyvision - computer vision toolbox. http://pyvision.sourceforge.net, 2008.

[2] L. Breiman. Bagging Predictors. *Machine Learning*, 24(2):123–140, 1996.

[3] R. Brunelli and T. Poggiot. Template matching: matched spatial filters and beyond. *Pattern Recognition*, 30(5):751–768, 1997.

[4] R. Duda and P. Hart. *Pattern Classification and Scene Analysis*, chapter 7.5 Template Matching, pages 276–284. John Wiley & Sons, 1973.

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

[6] Y. Ma, X. Ding, Z. Wang, and N. Wang. Robust precise eye location under probabilistic framework. *AFGR*, 2004.

[7] A. Mahalanobis, B. Vijaya Kumar, S. Song, S. Sims, and J. Epperson. Unconstrained correlation filters. *Appl. Opt.*, 33(17):3751, 1994.

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

[9] A. Mahalanobis, B. V. K. Vijaya Kumar, and S. R. F. Sims. Distance-classifier correlation filters for multiclass target recognition. *Appl. Opt.*, 35(17):3127–3133, 1996.

[10] P. Phillips, H. Moon, S. Rizvi, and P. Rauss. The FERET Evaluation Methodology for Face-Recognition Algorithms. *PAMI*, 22(10):1090–1104, October 2000.

[11] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes in C*, chapter 13.1 Convolution and Deconvolution, pages 535–545. Cambridge University Press, 1988.

[12] P. Refregier. Optimal trade-off filters for noise robustness, sharpness of the correlation peak, and Horner efficiency. *Optics Letters*, 16:829–832, June 1991.

[13] M. C. Santana, J. L. Navarro, O. D. Suárez, and A. F. Martel. Multiple face detection at different resolutions for perceptual user interfaces. In *PRIA*, Estoril, Portugal, June 2005.

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

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

[16] B. V. K. Vijaya Kumar, A. Mahalanobis, S. Song, S. Sims, and J. Epperson. Minimum squared error synthetic discriminant functions. *Optical Engineering*, 31:915, 1992.

[17] P. Viola and M. J. Jones. Robust real-time face detection. *Int. J. Comput. Vision*, 57(2):137–154, 2004.

[18] P. Wang, M. B. Green, Q. Ji, and J. Wayman. Automatic eye detection and its validation. In *CVPR*, Washington, 2005.

[19] L. Wiskott, J.-M. Fellous, N. Kruger, and C. von der Malsburg. Face recognition by elastic bunch graph matching. *PAMI*, 19(7):775–779, 1997.