Incremental Updating of Advanced Correlation Filters for Biometric Authentication Systems

Marios Savvides, Krithika Venkataramani, B.V.K. Vijaya Kumar,
Electrical and Computer Engineering, Carnegie Mellon University
msavvid@ri.cmu.edu, krithika@ece.cmu.edu, kumar@ece.cmu.edu

Abstract

In this paper we show mathematical formulation of incrementally building advanced correlation filters used in authentication systems that are based on face and fingerprint images as biometrics for verification. This method is crucial for incorporating such algorithms on small devices with limited memory and computational resources. We also present results that show that these correlation filters perform well for face and fingerprint images. We used the PIE (Pose, Illumination and Expression) database from CMU to test the verification performance using face images. Similarly, for fingerprint images we used the NIST Special Database 24 to evaluate verification performance.

1. Introduction

Traditional Correlation filter based methods [1] are based on using Matched Filters, which are optimal for detecting a known reference image in the presence of additive white Gaussian noise. However, detection degrades rapidly as the input image is even slightly different from the known reference image (this can be due to natural changes such as scale, rotation and pose).

Correlation filter designs [2] have changed to handle such types of distortions; the minimum average correlation energy (MACE) filter [3] is one such design which has been successful in the field of automatic target recognition as well as recent applications to biometric verification [4] [5]. Correlation filters are attractive techniques due to their in-built shift-invariance (i.e. if the input image is shifted, the correlation output will also shift by the same amount), thus classification performance are invariant to shifts of the input image. MACE filters are also popular for their high discrimination performance, making them ideal for such verification applications. They are different from Matched Filters in that we use more than one training image to synthesize a single filter template which we call as the MACE filter template. This design process, as we will show in the next section, is performed in the frequency domain for efficiency. Figure 1, illustrates the schematic of the correlation process.

1.1 Minimum Average Correlation Energy (MACE) Filters

The goal of MACE filters is to produce sharp peaks that resemble 2D delta-type correlation outputs when the input image belongs to the class of images that were used to build the filters (typically referred to as the authentic class). There are two variants which achieve this goal; the first is the MACE filter [3]. This filter minimizes the average correlation energy of the training images while constraining the correlation output at the origin to a specific value (usually set to 1) for each of the training images. This optimization is solved using Lagrange multipliers to yield:

$$h = D^{-1}X(X^*D^{-1}X)^{-1}c$$  \hspace{1cm}  (1)

Eq. (1) is the closed form solution to the linear constrained quadratic minimization. Here $D$ is a diagonal matrix with the average power spectrum of the training images placed along the diagonal elements. $X$ contains the Fourier transform of the training images lexicographically re-ordered and placed along each column e.g. if there are $N$ training images of size 64x64(=4096), then $X$ will be a 4096x$N$ matrix. Finally, $c$ is a column vector (of length $N$) containing the desired correlation outputs at the origin for each of the training images. Typically, we set all the entries of this column vector to 1.

The second MACE filter variant is the unconstrained minimum average correlation energy (UMACE) [6] filter which also minimizes the average output correlation plane energies resulting from the training images while maximizing the correlation output at the origin. This
optimization takes a form similar to Raleigh quotient which leads to the following closed form solution.

\[ \mathbf{h} = \mathbf{D}^{-1} \mathbf{m} \]  

(2)

Note that \( \mathbf{D} \) is the same \( \mathbf{D} \) as in Eq. (1), and \( \mathbf{m} \) a column vector containing the mean of the Fourier transforms of the training images. We can see by comparing Eq. (1) and Eq. (2), that Eq. (2) is simpler to implement from a computational viewpoint as it only involves inverting a diagonal matrix, whereas Eq. (1) requires the inversion of a \( \mathbf{N} \times \mathbf{N} \) matrix.

Noise tolerance can be built into both MACE filter variants. This was shown by Refrieger [7] and Kumar et al [8], that optimally trading-off discrimination tolerance for noise tolerance yields the following unconstrained optimal trade-off synthetic discriminant (UOTSDF) filter.

\[ \mathbf{h} = (\alpha \mathbf{D} + \sqrt{1-\alpha^2} \mathbf{C})^{-1} \mathbf{m} \]  

(3)

where \( \mathbf{C} \) is the power spectral density of the noise. For most applications we assume a white noise power spectral density, therefore \( \mathbf{C} \) reduces to the identity matrix. Also the \( \alpha \) term is typically set to be close to 1. This helps to achieve good performance even in the presence of noise, but also this helps improve generalization to distortions outside the training set.

In testing whether a test image is similar to the authentic class or not, we use the peak-to-sidelobe (PSR) ratio metric [4].

\[ \text{PSR} = \frac{\text{peak} - \text{mean}}{\sigma} \]  

(4)

where the peak is the largest value in resulting from the correlation output of a test image with a correlation filter. Mean is the average value of the sidelobe region (typically located as an annular region outside a 5x5 window centered at the peak. And \( \sigma \) is the standard deviation of this region. More details can be found in [4].

2. Incremental Building of UMACE/UOTSDF Filters

There are many practical situations where we would like to perform authentication on devices with limited memory resources. In such cases, storing all the training images or alternatively all the Fourier transforms of the training images might not be a feasible approach. Instead we must figure an incremental way to update the filter given a single image at a time, thus at any one time we only need to store a single training image. For such an approach, clearly using UMACE/UOTSDF is simpler framework for incremental building, as the MACE in Eq. (1), as \( \mathbf{N} \) (the number of training images increases), we need to find the inverse of an \( \mathbf{N} \times \mathbf{N} \) matrix (\( \mathbf{X}^\mathbf{T} \mathbf{D}^{-1} \mathbf{X} \)). Whereas for the UMACE/UOTSDF framework, we only need to update \( \mathbf{D} \) and \( \mathbf{m} \). The computations for inverting the diagonal is constant for any \( \mathbf{N} \), thus unlike MACE filters, we do not need to invert a variable size matrix. We can incrementally update a UMACE/UOTSDF type filter in the following way.

\[ \mathbf{D}_n = \frac{n-1}{n} \mathbf{D}_{n-1} + \frac{1}{n} \mathbf{X}_n \mathbf{X}_n^\mathbf{T} \]  

(5)

\[ \mathbf{m}_n = \frac{n-1}{n} \mathbf{m}_{n-1} + \frac{1}{n} \mathbf{x}_n \]  

(6)

where \( \mathbf{X}_n \mathbf{X}_n^\mathbf{T} \) in Eq. (4) is a diagonal matrix containing the power spectrum of the \( n \)th training image along its diagonal, and \( \mathbf{x}_n \) is a column vector containing the Fourier transform of the \( n \)th training image.

However, in the case of UMACE filter in Eq. (2), it can be shown that Eq. (4) and Eq. (5) can be simplified as shown below (since dividing by \( N \) both in \( \mathbf{D} \) and \( \mathbf{m} \) cancels out in Eq.(2) ) to give the following update:

\[ \mathbf{D}_n = \mathbf{D} + \mathbf{X}_n \mathbf{X}_n^\mathbf{T} \]  

(7)

\[ \mathbf{m}_n = \mathbf{m} + \mathbf{x}_n \]  

(8)

where the UMACE filter is given by \( \mathbf{h} = \mathbf{D}^{-1} \mathbf{m} \).

3. Face Verification using PIE database

For authentication systems, we can assume that the user is co-operative as he/she wants to be authenticated, therefore the assumption we make is that the user will position his head to provide a frontal pose. However, it is unreasonable to expect the user to control the illumination conditions of his environment for authentication, thus the bigger problem is to perform reliable authentication in the presence of illumination variations. To evaluate this, we used the two illumination subsets of the CMU PIE database [9]. Both datasets contain 65 people, with 21 images (we resized them to 100x100 pixels) taken under variable illumination, however one dataset was taken with ambient background lights on (which we will refer to as PIE-L) and the other without any ambient lights on (we refer to this dataset as PIE-NL). The PIE-NL dataset is the
hardest of the two as the variations due to illumination changes have a greater effect on the face images. An example set of 21 images from PIE-NL dataset is shown in Figure 2, for person 2.

In our evaluation experiments, we used three training images to synthesize a single UMACE filter (biometric template) for each person. The three images selected were that of extreme lighting conditions, i.e. left shadow (image 3), frontal lighting (image 7) and right shadow (image 16). We then cross-correlated each person’s filter with every image in the whole dataset (65*21=1365 images) to see the verification/rejection performance. Therefore we performed a total of 65*1365=88,725 correlations for each dataset.

In our results, for all 65 persons we have achieved 100% verification rates. This implies that for every person and their biometric filter, there exists a single threshold that distinguishes him/her from all other 64 people when we perform cross-correlation with a test face image. Figure 3, is a typical peak-to-sidelobe ratio plot demonstrating that MACE/UMACE filters are very similar in performance. The bottom dotted and solid line is the maximum PSR from the 64 other impostors, thus we observe a desired attribute of using MACE type filters that show that impostors PSRs are controlled and fall and below a fixed-global threshold, regardless of person or illumination variation.

4. Fingerprint Verification using NIST Special Database 24

The NIST special database 24 [10] is used here since it is a standard publicly available database and it contains images from a digital fingerprint sensor. This database has 2 sets of data – a plastic distortion set and a rotation set. In the plastic distortion set, fingers are rolled and twisted and so the images get distorted. In the rotation set, fingers are placed at different angles without being twisted or rolled. Since there will be distortion between fingerprint images obtained during enrollment and those obtained in verification, we choose the plastic distortion set for our study in order to gauge the distortion tolerance potential of correlation filters. This would be a harder database than the rotation set. We use a subset of the plastic distortion set, the thumb fingers from 10 people, with 300 images (zero-padded to 512x512 pixels) for each finger. Certain fingers, like the thumb fingers of persons 9 and 10 are excessively twisted, so the images appear more distorted. Sample images of person 10 are shown in Figure 4. Certain fingers, like the thumb fingers of persons 3 and 7, are rolled but hardly twisted, therefore there is not much variation in these classes.

For most of the classes we used 20 uniformly sampled images from each class to generate a single UOTSDF filter [8]. However, for classes 4, 9 and 10, we generated five UOTSDF filters from the 20 training images, i.e. 4 images per filter, as there is more variation in the images. It should be noted that images from other people were not used as impostor images to train the filter. The filters are built incrementally and updated with new training images only if they are not already well represented in the filter. For each class, we cross-correlated the filter(s) of the same class with 300 authentic images of the same class and 300*9=2700 impostor images of other classes. For a given verification PSR threshold $\theta_v$, for a class, the performance can be measured by false acceptance (FA) and false rejection (FR), defined below.

$$\text{FA} = \frac{\text{Number of impostor images having PSR} > \theta_v}{\text{Total number of impostor images}}$$ (8)

$$\text{FR} = \frac{\text{Number of authentic images having PSR} < \theta_v}{\text{Total number of authentic images}}$$ (9)

![Figure 3. PSR plots comparing verification performance using Person 2's MACE/UMACE filter on the whole PIE illumination dataset (with no ambient lighting).](image)

![Figure 4. Sample images of the thumb finger of person 10.](image)
Figure 5 shows FA and FR plotted against the verification PSR threshold for all classes. The dotted lines correspond to FA and the solid lines correspond to FR. Thresholds could be found for all classes such that there is no classification error, except for one image of class 4 and three images of class 10 which were falsely rejected when FA is zero.

In addition, it can be seen from the Figure 5 that there is a wide margin of separation between the minimum threshold where FA is zero and the maximum threshold where FR is zero for classes with little variation such as classes 3 and 7. There is a good margin of separation for other classes except for classes 4 and 10.

5. Conclusion

We have shown that UMACE/UOTSDF filters achieve high verification performance for biometrics such as face and fingerprints. In order to incorporate such algorithms on limited memory devices, we have proposed the incremental updating scheme which is computationally attractive and only requires the storage of a single training image at a time. Furthermore, this updating method can allow us to iteratively select which of the captured images (during the enrollment stage) needs to be used to update the filter.

Acknowledgements

This research is supported in part by SONY Corporation.

References