Template Learning using Wavelet Domain Statistical Models

Karthikeyan Natesan Ramamurthy, Jayaraman J. Thiagarajan and Andreas Spanias

Abstract Wavelets have been used with great success in applications such as signal denoising, compression, estimation and feature extraction. This is because of their ability to capture singularities in the signal with a few coefficients. Applications that consider the statistical dependencies of wavelet coefficients have been shown to perform better than those which assume the wavelet coefficients as independent. In this paper, a novel Gaussian mixture model, specifically suited for template learning is proposed for modeling the marginal statistics of the wavelet coefficients. A Bayesian approach for inferring a low dimensional statistical template with a set of training images, using the independent mixture and the hidden Markov tree models extended to the template learning case, is developed. Results obtained for template learning and pattern classification using the low dimensional templates are presented. For training with a large data set, statistical templates generated using the proposed Bayesian approach are more robust than those generated using an information-theoretic framework in the wavelet domain.

1 Introduction

Wavelet domain statistical models have found extensive applications in image denoising and coding. In this paper, the application of wavelet statistical models to the problem of template learning for generating a low dimensional statistical template in a Bayesian approach is addressed. There are two facets to this problem, one is to provide an appropriate statistical model for exploiting the wavelet coefficient dependencies, and the other is to use this statistical model for learning a low dimensional template in the wavelet domain for synthesis and classification. A specific type of Gaussian mixture modeling of the wavelet coefficients for a template learning ap-

Karthikeyan Natesan Ramamurthy, Jayaraman J. Thiagarajan, Andreas Spanias
School of Electrical, Computer and Energy Engineering, Arizona State University, USA.
e-mail: {knatesan,jjayaram,spanias}@asu.edu

M. Bramer et al. (eds.), *Research and Development in Intelligent Systems XXVI*,
DOI 10.1007/978-1-84882-983-1_13, © Springer-Verlag London Limited 2010

plication is described. The IM model proposed in [4] and wavelet HMT models proposed in [1] are extended to the case of multiple training images. The Viterbi algorithm for estimating the most likely states given a set of training images and the estimated parameters of the statistical model is derived. The problem of registering a set of training patterns to the template is also described in detail as a part of the template learning procedure. The proposed Bayesian approach for template learning is compared with the Template Learning from Atomic Representations (TEMPLAR), which is an information-theoretic framework and the advantages that our procedure has in terms of avoiding overfitting is demonstrated.

The relevant prior work on template learning and wavelet domain statistical models is presented in Section 2. Section 3 presents the wavelet domain statistical models developed for the purpose of template learning. The proposed approach for template learning, along with the Viterbi algorithm for estimating the states and a method for registration of training images with the template are presented in Section 4. Section 5 discusses the results for generating the low dimensional template with the training images and pattern classification using the generated template. The discussion concludes with comments in Section 6.

2 Prior Work

2.1 Template Learning

Template learning is the process of learning a representative pattern from the set of training patterns under consideration. The need to register the training images is inherent to the problem of template learning. Separation of background from the pattern of interest and modeling the local deformations are also key problems associated with template learning. An approach for template learning and classification in the wavelet domain, TEMPLAR, has been proposed in [2]. In this framework [2], the wavelet coefficients are assumed to follow a two-state Gaussian mixture distribution locally and an independence assumption is imposed on the coefficients. In general, edges represent the significant information in any pattern. Hence, TEMPLAR exploits the edge detection properties of the wavelet transform. The Minimum Description Length (MDL) principle, an information theoretic criterion, is used to select the significant coefficients that represent the edges in an image.

2.2 Wavelet Domain Statistical Models

Wavelet coefficients have been assumed to be statistically independent in many applications, because the wavelet transform approximately whitens a AR-1 process. However, they exhibit significant statistical dependencies within a particular

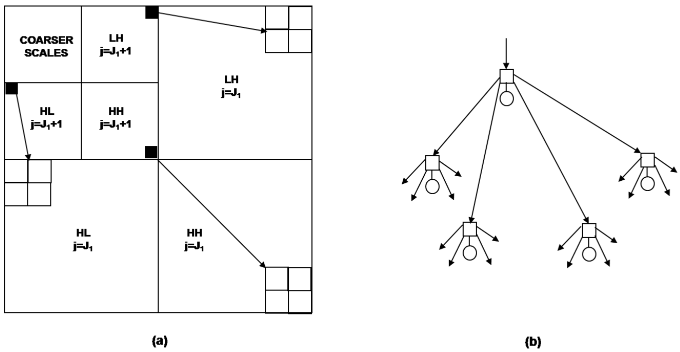

Fig. 1 (a) Parent-child relationship between wavelet coefficients across scale, (b) quad-tree structure of state connections between parents and children.

scale as well as across scales [3]. The marginal statistics of wavelet coefficients are highly non-Gaussian in nature [4] and therefore any wavelet statistical model should take the marginal statistics as well as the coefficient interdependencies into consideration. The Independent Mixture (IM) model captures the highly non-Gaussian marginal statistics of wavelet coefficients, using a two state Gaussian Mixture Model (GMM) and considers the coefficients to be independent [4]. An intuitive and effective Hidden Markov Tree (HMT) model builds on the IM model and captures the inter-scale dependencies between the coefficients [1], and this has been successfully used in denoising. The wavelet coefficients of an image form a natural quad-tree structure and a separate HMT model will be trained for the tree corresponding to each of the three subbands. The parent-child relationship and the quad-tree structure of the wavelet coefficients across scales are shown in Figure 1.

3 Wavelet Domain Statistical Models in Template Learning

Explicitly modeling the coefficient statistics when there are multiple training images, is a problem that has not been well addressed in the literature. Therefore, modeling the wavelet coefficient statistics is needed along with the statistical dependencies they exhibit with the other coefficients.

3.1 Proposed Gaussian Mixture Model

GMMs used for modeling wavelet coefficients have the form that assumes two zero mean Gaussians, one with a low variance and the other with a high variance [1]. For template learning, a different form of GMM needs to be used because the local statistics of wavelet coefficients need to be taken into consideration. The pattern

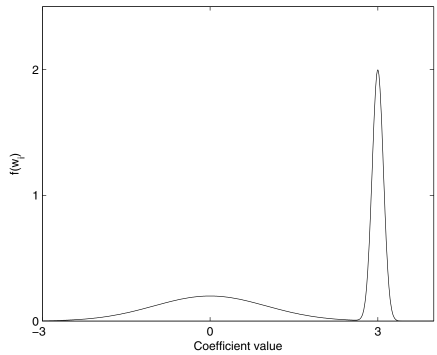

Fig. 2 Two state Gaussian mixture model for template learning. The low state (zero mean) models the background and smooth regions, whereas the high state (non-zero mean) corresponds to the pattern.

and background information need to be modeled efficiently, so that they can be clearly distinguished. The GMM proposed is a two state model and for the wavelet coefficient W_i, the density is given by,

$$f_{W_i}(w_i) = \sum_{m=1}^{M} p_{Q_i}(m) f_{W_i|Q_i}(w_i|Q_i = m). \tag{1}$$

The conditional density of the wavelet coefficients are given as $f_{W_i|Q_i}(w_i|Q_i = m) \sim \mathcal{N}(\mu_{i,m}, \sigma^2_{i,m})$. The state $Q_i = 1$ represents a zero mean Gaussian and the state $Q_i = 2$ represents the Gaussian with non-zero mean.

Each wavelet coefficient W_i is assumed to have a separate $\mu_{i,2}$ and $\sigma^2_{i,2}$, whereas $\mu_{i,1} = 0$ and $\sigma^2_{i,1}$ is constrained to be the same for all the coefficients in a subband. An illustration of this mixture model is shown in Figure 2. The state $Q_i = 1$ models the background and smooth regions with a zero mean and common variance across the subband, whereas the state $Q_i = 2$ corresponds to the pattern. For convenience, $Q_i = 1$ will be referred to as the *low* state for the coefficient i and $Q_i = 2$ will be referred to as the *high* state. This model also agrees with the intuition that less parameters must be used to model the background and more parameters must be used for the actual pattern itself.

3.2 Extending the IM Model

The wavelet coefficients can be modeled as independent Gaussian mixtures using the prior density proposed in Section 3.1. It is assumed that there are T training images and the wavelet coefficient i of the training image t is given by w_i^t. w_i^t are assumed to be independent realizations of the random variable W_i that follows the GMM. $\mathbf{w}_i$ is a vector that has all the T realizations w_i^t of W_i. The EM algorithm is used to compute the parameters of the IM model given by $\Theta_{IM} = \{p_{Q_i}(1), \mu_{i,1}, \mu_{i,2}, \sigma_{i,1}^2, \sigma_{i,2}^2\}$, where $\mu_{i,1} = 0$ and $\sigma_{i,1}^2$ is constrained to be the same for all coefficients in a subband.

In the iteration l, the E-step computes the following conditional probability,

$$p(Q_i = m|w_i^t, \Theta_{IM}^{(l)}) = \frac{p(w_i^t|Q_i = m, \Theta_{IM}^{(l)})p_{Q_i}(m)}{\sum_{m=1}^{M} p(w_i^t|Q_i = m, \Theta_{IM}^{(l)})p_{Q_i}(m)}. \tag{2}$$

The M-Step estimates the parameters as,

$$p_{Q_i}(m) = \frac{1}{T}\sum_{t=1}^{T} p(Q_i = m|w_i^t, \Theta_{IM}^{(l)}), \tag{3}$$

$$\sigma_{i,1}^2 = \frac{\sum_{t=1}^{T}\sum_{k\in SB(i)} (w_k^t)^2 p(Q_k = 1|w_k^t, \Theta_{IM}^{(l)})}{\sum_{t=1}^{T}\sum_{k\in SB(i)} p(Q_k = 1|w_k^t, \Theta_{IM}^{(l)})}, \tag{4}$$

$$\mu_{i,2} = \sum_{t=1}^{T} w_i^t p(Q_i = 2|w_i^t, \Theta_{IM}^{(l)}) \big/ (T p_{Q_i}(2)), \tag{5}$$

$$\sigma_{i,2}^2 = \sum_{t=1}^{T} (w_i^t - \mu_{i,2})^2 p(Q_i = 2|w_k^t, \Theta_{IM}^{(l)}) \big/ (T p_{Q_i}(2)), \tag{6}$$

where $SB(i)$ returns the indices of all the coefficients in the subband corresponding to the coefficient i. The low state mean, $\mu_{i,1} = 0$ and the low state variance, $\sigma_{i,1}^2$ is the same for all coefficients in the subband.

3.3 Extending the HMT Model

The HMT model proposed in [1] will be extended to the case of multiple training images using the GMM proposed in the Section 3.1. Assuming that T is the number of training images and each image is decomposed into maximum possible scales, there are totally $3T$ independent wavelet trees. This is because of the assumption that each of the three quad-trees in the wavelet decomposition of an image will be considered independent of each other. In this discussion only one of the three trees per wavelet decomposition is considered, as the generalization to the case of multiple trees in an image is trivial.

The value of the wavelet coefficient at node i in a tree t is indicated by w_i^t. In the case of IM model, i indexes all the coefficients in the wavelet decomposition whereas in the HMT model, i indexes the coefficients in a tree corresponding to one of the subbands. The posterior probabilities $p(Q_i = m|\mathbf{w}^t, \Theta^{(l)}_{HMT})$ and $p(Q_i = m, Q_{\pi(i)} = n|\mathbf{w}^t, \Theta^{(l)}_{HMT})$ are computed using the relevant equations in [9]. The upward-downward step is equivalent to the E-step [9] and the parameter update is equivalent to the M-step. The parameters of the HMT model at iteration l are then computed using,

$$p_{Q_i}(m) = \frac{1}{T}\sum_{t=1}^{T} p(Q_i = m|\mathbf{w}^t, \Theta^{(l)}_{HMT}), \tag{7}$$

$$a^{mn}_{i,\pi(i)} = \sum_{t=1}^{T} p(Q_i = m, Q_{\pi(i)} = n|\mathbf{w}^t, \Theta^{(l)}_{HMT}) \big/ (T p_{Q_{\pi(i)}}(n)), \tag{8}$$

$$\mu_{i,2} = \sum_{t=1}^{T} w^t_i p(Q_i = 2|\mathbf{w}^t, \Theta^{(l)}_{HMT}) \big/ (T p_{Q_i}(2)), \tag{9}$$

$$\sigma^2_{i,1} = \frac{\sum_{t=1}^{T} \sum_{k \in SB(i)} (w^t_k)^2 p(Q_k = 1|\mathbf{w}^t, \Theta^{(l)}_{HMT})}{\sum_{t=1}^{T} \sum_{k \in SB(i)} p(Q_k = 1|\mathbf{w}^t, \Theta^{(l)}_{HMT})}, \tag{10}$$

$$\sigma^2_{i,2} = \sum_{t=1}^{T} (w^t_i - \mu_{i,2})^2 p(Q_i = 2|\mathbf{w}^t, \Theta^{(l)}_{HMT}) \big/ (T p_{Q_i}(2)). \tag{11}$$

Note that $\sigma^2_{i,1}$ is common for all coefficients in the subband and $\mu_{i,1} = 0$ as in the case of IM model.

The EM procedure can be used to estimate the parameters of all the three quadtrees in the wavelet decomposition of an image. The final set of parameters for the three trees together is denoted by Θ^A_{HMT} and it can be used to estimate the low dimensional template Θ_{LD}.

4 Proposed Approach

The Bayesian approach for learning the parameters of a low dimensional template from a set of noisy observations using the IM and HMT wavelet domain statistical models is presented in this section. It essentially combines the three steps of parameter estimation using the IM or HMT models proposed in the previous section, state estimation using the Viterbi algorithm and registration of training images to the current estimate of the template.

4.1 Viterbi Algorithm for Estimating the Most Likely States

The Viterbi algorithm for a HMT model is presented in [5], where it is used for thresholding the wavelet coefficients of an image to denoise and enhance the edges. In this paper, we extend it to the case of multiple training images for the purpose of estimating a low dimensional template from a set of noisy, training images. The proposed Viterbi algorithm estimates the most likely states for the model using all the training images. This, in essence, fixes the state pointwise in the template so that conditional independence assumption can be imposed on the wavelet coefficients.

Given the observations of multiple trees of wavelet coefficients $\mathbf{w}^1,...,\mathbf{w}^T$, the problem is to estimate the set of most likely states $\mathbf{q}$ and this can be expressed as,

$$\hat{\mathbf{q}} = \underset{\mathbf{q}}{\operatorname{argmax}}\, p\left(\mathbf{q}|\mathbf{w}^1,...,\mathbf{w}^T,\Theta_{HMT}\right). \tag{12}$$

Let $\mathscr{P}_i^t$ be the set of wavelet coefficients at the nodes in the shortest path on the tree t, between the root node and the node i, and $\mathscr{Q}_i$ be the states on the path. $\delta_i(q)$ is defined as the highest likelihood along a single path that ends at node i in state q and is calculated as,

$$\delta_i(q) = \max_{\mathscr{Q}_{\pi(i)}} f\left(\mathscr{P}_i^1,...,\mathscr{P}_i^T,\mathscr{Q}_{\pi(i)},Q_i = q|\Theta_{HMT}\right). \tag{13}$$

In order to find the best possible state sequence, the following steps are performed.

1. At the coarsest scale compute $\delta_1(q)$, for $q \in \mathscr{S}$, where $\mathscr{S} = \{1,2\}$ is the set of possible states.
$$\delta_1(q) = p_{Q_1}(q)\prod_{t=1}^{T} g\left(w_1^t|\mu_{1,q},\sigma_{1,q}^2\right) \tag{14}$$
2. Moving down the tree compute the following for each node in a subband
$$\delta_i(q) = \max_{z\in\mathscr{S}}\left(\delta_{\pi(i)}a_{i,\pi(i)}^{qz}\right)\prod_{t=1}^{T} g\left(w_i^t|\mu_{i,q},\sigma_{i,q}^2\right) \tag{15}$$
$$\xi_{\pi(i)}(\mathscr{C}) = \underset{z\in\mathscr{S}}{\operatorname{argmax}}\left(\delta_{\pi(i)}a_{i,\pi(i)}^{qz}a_{i,\pi(i)}^{sz}a_{i,\pi(i)}^{uz}a_{i,\pi(i)}^{vz}\right), \tag{16}$$
for $q \in \mathscr{S}$ and $\mathscr{C} = \{q,s,u,v\}$, where each quantity in $\mathscr{C}$, takes a value from the set $\mathscr{S}$. $\xi_{\pi(i)}(\mathscr{C})$ is the most likely state at node $\pi(i)$ to have the four children $\mathscr{C}$.
3. Compute the best possible state for each coefficient in the finest scale,
$$\hat{q}_i = \underset{z\in\mathscr{S}}{\operatorname{argmax}}\left(\delta_i(z)\right). \tag{17}$$
4. Estimate $\hat{q}_i$ for the coefficients at node i, moving up the scale and backtracking the tree,
$$\hat{q}_i = \xi_i\left(\hat{q}_{c(i)}\right). \tag{18}$$

With the estimated state sequence $\hat{\mathbf{q}}$ for all the three trees, the nodes at which $\hat{q}_i = 1$, are called as *insignificant* and the nodes at which $\hat{q}_i = 2$ are called *significant*. For generating a low dimensional statistical template, the significant coefficients in a particular location across the training images are modeled individually with a non-zero mean Gaussian. All the insignificant coefficients in a subband across the training images are modeled together using a zero-mean Gaussian. The sets N_{ins} and N_{sig} contain the indices corresponding to the insignificant and significant coefficients respectively and $N = |N_{sig} \cup N_{ins}|$. The low dimensional template is parameterized by $\Theta_{LD} = \{\mu_i, \sigma_i^2\}_{i=1}^N$. If $i \in N_{sig}$, then Θ_{LD} is estimated as,

$$\mu_i = \frac{1}{T}\sum_{t=1}^{T} w_i^t \qquad \text{and} \qquad \sigma_i^2 = \frac{1}{T}\sum_{t=1}^{T}(w_i^t - \mu_i)^2. \tag{19}$$

If $i \in N_{ins}$, then the parameters are given by,

$$\mu_i = 0 \qquad \text{and} \qquad \sigma_i^2 = \frac{1}{T|N_{ins}|}\sum_{t=1}^{T}\sum_{k\in N_{ins}} (w_k^t)^2. \tag{20}$$

This low dimensional template estimated using the Viterbi can be compared with that of TEMPLAR. TEMPLAR uses an information-theoretic criterion to estimate the template, whereas a Bayesian approach is used here. For the IM case also, an algorithm similar to the Viterbi algorithm provided above can be used to infer the best possible states and (19) and (20) can be used to estimate Θ_{LD}.

4.2 Registration with the Low Dimensional Template

The problem of registering training observations to the template is a key step in template learning. From the parameters of the low dimensional template, Θ_{LD}, the equivalent spatial domain parameters, $\Theta_{LDS} = \{\mu, \Sigma\}$ can be computed using Gaussian algebra and the orthonormality of wavelet basis functions [10]. Registration of a training observation $\mathbf{u}$ can be performed using the Maximum Likelihood (ML) approach as,

$$\hat{\ell} = \underset{\ell}{\operatorname{argmax}} \log p(\mathbf{D}\Gamma_\ell \mathbf{u}|\mu, \Sigma), \tag{21}$$

where $\mathbf{D}$ and Γ_ℓ are the Discrete Wavelet Transform (DWT) matrix and translation matrix respectively. This means that the estimation of the most likely transformation of the training observation to the template is done using a likelihood measure in the wavelet domain.

4.3 Learning the Template Parameters

Learning the parameters of the low dimensional template is done as an alternating maximization problem as done with TEMPLAR. The three steps of the iterative procedure for learning the template parameters with the HMT based algorithm are:

1. Parameter Estimation: The parameters of the HMT model, Θ^{A}_{HMT}, are estimated as per Section 3.3 using the wavelet coefficients of the registered images at the current iteration.
2. State Estimation: The most likely states, $\mathbf{q}$, of the nodes are computed using the Viterbi algorithm given in Section 4.1. The parameters of the low dimensional template Θ_{LD} are also estimated in this step using (19) to (20).
3. Registration: The registration of images to the low dimensional template in the wavelet domain, Θ_{LD}, is performed as per Section 4.2.

The three steps are repeated in sequence, till convergence is reached. The algorithm is said to have converged when the training images are perfectly aligned to the low dimensional template. Although a theoretical proof for convergence is not provided, in the experiments performed, convergence has always happened. Another important consideration is that, registration is performed using a robust and fast multiresolution approach from coarse to fine scale.

The complexity of parameter estimation using the HMT or IM algorithm for T training images of size N is order NT. State estimation using the Viterbi procedure and estimation of the low dimensional template, detailed in Section 4.1 are also of order NT complexity. The registration of training images to the low dimensional template is the most expensive procedure and for the set L of all possible transformations, the complexity is of order $|L|NT$. A low complexity procedure of order $|L|T\log N$ was developed for registering the training images to the template and reported in [10].

5 Results and Discussion

In this section, we provide the results for generating statistical low dimensional templates using the training data sets and pattern classification using the templates learned from the training sets.

5.1 Template Generation from the Training Sets

For the purpose of template generation three data sets are considered. The training data set A contains images from the MNIST database available online [6]. A total of 500 samples of each are chosen for training. Training data sets B and C are chosen

Fig. 3 Training data set B - First subject chosen from the Yale face database. 8 images are chosen from a set of 11 and they are cropped to remove most of the background.

Fig. 4 Training data set C - Fifteenth subject chosen from the Yale face database. The 8 images chosen are cropped to remove the background.

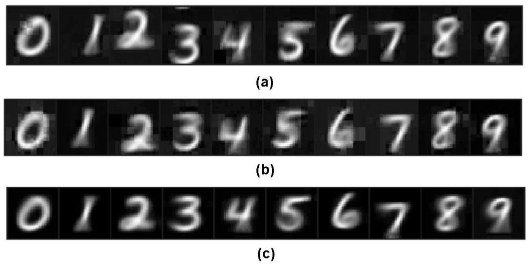

Fig. 5 Spatial mean of the template for training data set A with additive i.i.d. Gaussian noise (a) using IM based template learning, (b) using HMT based template learning, (c) using TEMPLAR.

from the Yale face database available online [7] and are shown in Figures 3 and 4 respectively.

IM and wavelet HMT based template learning algorithms are used to infer the parameters of the low dimensional statistical template Θ_{LD}. The existing MATLAB implementation for the TEMPLAR algorithm was used to generate statistical templates using the same training data sets for comparison [8]. The images in all training sets are grayscale and have pixel values between 0 and 255. For the training data set A, the original digits of size 28×28 are scaled to 32×32 and i.i.d. Gaussian noise with standard deviation 25.5 is added. The training images are Haar wavelet transform is used for decomposing the images into maximum possible levels. All the 500 samples of each digits are used for training. The mean parameter of Θ_{LD} are transformed to the spatial domain and the mean templates for the digits are given in Figure 5.

The spatial mean of the templates under conditions of no noise show that the IM and HMT based template learning have performed automatic registration and the mean templates are comparable with that of TEMPLAR. The number of significant

Table 1 NUMBER OF SIGNIFICANT COEFFICIENTS ESTIMATED BY THE THREE ALGORITHMS FOR TRAINING DATA SET A

Digit	IM Based		HMT Based		TEMPLAR	
	With Noise	No Noise	With Noise	No Noise	With Noise	No Noise
0	588	499	492	688	696	882
1	328	281	280	520	385	860
2	518	483	464	584	627	880
3	524	477	468	632	648	884
4	467	454	426	594	616	882
5	475	466	443	663	624	880
6	495	462	427	574	562	869
7	436	381	340	557	555	844
8	574	512	469	589	572	879
9	459	453	424	615	589	848

coefficients chosen by each algorithm for a given digit are given in Table 1. TEMPLAR chooses the lowest number of significant states in each case and the IM based algorithm chooses the highest. From the table it can also be seen that, for some cases the number of significant states estimated by the HMT based algorithm is quite close to that of TEMPLAR. But the IM based algorithm always does a overestimate. HMT based significant state estimation is the framework proposed for template learning using Bayesian approach and it performs comparably with the information theoretic approach of TEMPLAR using MDL principle for state estimation, in certain cases.

The number of significant coefficients for the cases when the training set A is not corrupted with noise is given in Table 1. It can be seen that TEMPLAR severely overfits the data in every case because it estimates a large number of significant coefficients for all the templates. Large number of significant coefficients mean that most of the coefficients are treated as edges, whereas this is not the actual case. When overfitting happens, the generalization error increases and hence the template will not generalize well to the patterns outside the training set. IM and HMT based algorithms have much reduced overfitting when compared to TEMPLAR. When a large number of similar data are available, as in this case, a simple model such as TEMPLAR will overfit, whereas complex models such as IM and HMT will have a lesser chance of overfitting. This is because, complex models reliably estimate their parameters using the large training data set. In cases where the data set contains data with high similarity, additive noise tends to regularize and improves the generalization, as could be observed from the results of the previous experiment given in Table 1. Additive noise is also used to extend small training sets in order to prevent overfitting and improve generalization. This idea is used in the next experiment where the data sets B and C are extended by adding noise.

The templates generated for the training images of the data sets B and C are given in Figures 6 and 7 respectively. Each training data set is extended to 500 images and each image in the data set is scaled to 128×128 with i.i.d. Gaussian noise of standard deviation 25.5 added. For training set B, IM based algorithm estimates a total of 3776 significant states, HMT based algorithm estimates 2781 significant states and TEMPLAR estimates 3819 states. For the training data set C, 4124 significant

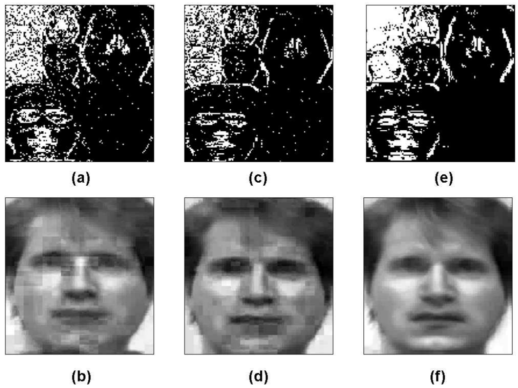

Fig. 6 Training data set B with additive i.i.d Gaussian noise: (a) and (b) State map and mean template using IM based template learning, (c) and (d) state map and mean template using HMT based template learning, (e) and (f) state map and mean template using TEMPLAR.

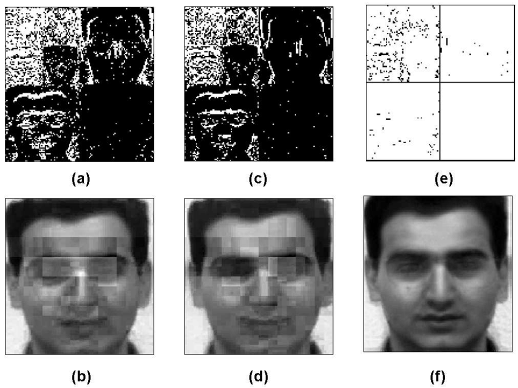

Fig. 7 Training data set C with additive i.i.d Gaussian noise: (a) and (b) State map and mean template using IM based template learning, (c) and (d) state map and mean template using HMT based template learning, (e) and (f) state map and mean template using TEMPLAR.

states are estimated by the IM based algorithm, 2993 by the HMT based algorithm and 15497 by TEMPLAR. It can be clearly seen that for the training data set C, TEMPLAR significantly overfits the data. However HMT and IM, being more complex models, do not overfit the data. Therefore, the proposed HMT and IM based models have a significant advantage over the existing TEMPLAR algorithm. Furthermore, the use of Viterbi state estimation to compute significant states and a low dimensional template guards against overfitting.

5.2 Pattern Classification using Learned Templates

The classification of test data using the generated templates is performed using an ML approach. The wavelet domain and its corresponding spatial domain template, denoted by Θ_{LD}^{k} and Θ_{LDS}^{k} respectively, are generated for each class k of the training data. Classification of the test data $\mathbf{u}$ is performed by registering the test data to each spatial domain class template Θ_{LDS}^{k} and finding the most likely class $\hat{k}$ using an ML

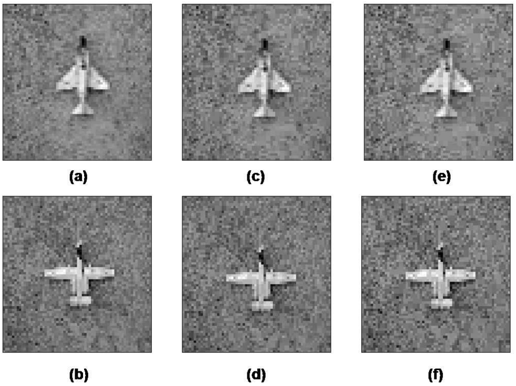

Fig. 8 (a) and (b) Original plane images, (c) and (d) spatial mean templates using IM based algorithm, (e) and (f) spatial mean templates using HMT based algorithm.

approach. The corresponding optimization problem can be posed as,

$$\hat{k} = \underset{k}{\operatorname{argmax}} \left[\max_{\ell^k} p(\mathbf{u}|\Theta^k_{LDS}, N^k_{sig}, -\ell^k) \right]. \tag{22}$$

The complexity of this step is linear with the number of transformations $|L|$. A low complexity version of this procedure is described in [10].

Two images of a plane [2] as given in Figures 8 (a) and (b) were used for classification. Each image was translated randomly to ± 3 pixels and corrupted with i.i.d. Gaussian noise of standard deviation 25.5. A total of 500 realizations of each image were used to form the template for each class using both the IM and the HMT based template learning algorithms. The templates generated with the IM based algorithm are given in Figures 8 (c) and (d) and with the HMT based algorithm are given in Figures 8 (e) and (f) respectively. Similarly, 500 realizations of each image were generated and were classified using the templates generated with no classification errors.

6 Conclusions

In this paper, we proposed a novel form of the Gaussian mixture model suited for the purpose of template learning. This was used along with the IM and HMT models, that were extended for the case of multiple training images. A Bayesian approach for learning a low dimensional template from a set of training observations using wavelet domain statistical models is the key contribution of this paper. Results show that models learned using the proposed approach are more robust when compared to models learned using an information theoretic framework, in cases of large training data sets. We can extend this framework to handle affine transformations of training images using our low-complexity framework for image registration [10]. Though this framework based on the wavelet transform provides good results in template learning and classification, sophisticated transforms such as the complex wavelet transform can be used to learn templates that are more robust to spatial transformations of the training images.

References

1. Crouse, M., Nowak, R. and Baraniuk, R.: Wavelet-based statistical signal processing using hidden markov models. IEEE Transactions on Signal Processing. **46**(4), 886–902 (1998).
2. Scott, C.: A hierarchical wavelet-based framework for pattern analysis and synthesis. M.S. thesis, Rice University, Houston, TX, USA (2000).
3. Liu, J. and Moulin, P.: Information-theoretic analysis of interscale and intrascale dependencies between image wavelet coefficients. IEEE Transactions on Image Processing. **10**(11), 1647–1658 (2001).
4. Chipman, H., Kolaczyk, E. and McCulloch, R.: Adaptive bayesian wavelet shrinkage. Journal of The American Statistical Association. **92**(440), 1413–1421 (1997).
5. Romberg, J.: A universal hidden markov tree image model. M.S. thesis, Rice University, Texas, USA (1999).
6. LeCun, Y., Bottou, L., Bengio, Y. and Haffner, P.: Gradient-based learning applied to document recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence. **86**(11), 2278–2324 (1998).
7. Yale face database. Available online at http://cvc.yale.edu/projects/yalefaces/yalefaces.html.
8. Matlab code for wavelet-based template learning and pattern classification using TEMPLAR. Available online at http://dsp.rice.edu/software/templar.shtml.
9. Ramamurthy, K. N.: Template learning with wavelet domain statistical models for pattern synthesis and classification. M.S. thesis, Arizona State University, Tempe, AZ, USA (2008).
10. Ramamurthy, K. N., Thiagarajan, J. J. and Spanias, A.: Fast image registation with non-stationary Gauss-Markov random field templates. Accepted to the IEEE International Conference on Image Processing, Cairo, Egypt (2009).