A Chi-Squared-Transformed Subspace of LBP Histogram for Visual Recognition

Jianfeng Ren, *Student Member, IEEE,* Xudong Jiang, *Senior Member, IEEE,* and Junsong Yuan, *Senior Member, IEEE*

***Abstract*—Local binary pattern (LBP) and its variants have been widely used in many recognition tasks. Subspace approaches are often applied to the LBP feature in order to remove unreliable dimensions, or to derive a compact feature representation. It is well-known that subspace approaches utilizing up to the second-order statistics are optimal only when the underlying distribution is Gaussian. However, due to its non-negative and simplex constraints, the LBP feature deviates significantly from Gaussian distribution. To alleviate this problem, we propose a Chi-squared transformation (CST) to transfer the LBP feature to a feature that fits better to Gaussian distribution. The proposed CST leads to the formulation of a two-class classification problem. Due to its asymmetric nature, we propose a CST-APCA approach to better remove the unreliable dimensions in the CST feature space. We have conducted extensive experiments on spatial LBP for face recognition, protein cellular classification, and spatial-temporal LBP for dynamic texture recognition. All experiments show that the proposed feature transformation significantly enhances the recognition accuracy.**

***Index Terms*—Local Binary Pattern, Truncated Gaussian Model, Chi-squared Transformation, Asymmetric PCA, Image Recognition**

I. Introduction

LOCAL binary pattern captures image micro-patterns by encoding the signs of the pixel differences between a pixel and its neighbors as a binary code. The histogram of these codes is then used for further analysis. LBP and its variants have been widely used in many recognition applications, e.g. texture classification [1]–[4], dynamic texture recognition [5]–[7], facial analysis [8]–[11], scene classification [12]–[14] and others [15]–[19]. LBP is popular thanks to its desirable properties: insensitive to illumination variations, robust to alignment error, simple and fast to extract.

However, LBP has some limitations. As shown in [9], [10], LBP is sensitive to image noise. A small image variation may alter LBP codes dramatically. A more severe problem is that the LBP histogram may not be reliably estimated due to: 1) The occurrence frequencies of LBP codes vary significantly. Histogram bins corresponding to infrequent patterns are poorly estimated. 2) To preserve locality, a small image patch is often used to construct the histogram, leading to insufficient pixels to produce a reliable histogram.

To remove the unreliable dimensions of LBP histogram or to derive a compact feature representation, subspace approaches are often applied. For CENTRIST [12], principal component analysis (PCA) was applied to the LBP histogram of each patch. CENTRIST was improved in [20] by mining discriminative co-occurrence patterns. In [21], a random projection tree was utilized to encode LBP-like features, and PCA was applied to reduce the dimensionality. PCA has also been applied on the feature vector concatenated using LBP histograms of all patches [22]–[25], and multi-level HOG-LBP feature [26].

It is well-known that subspace approaches utilizing up to the second-order statistics are optimal under Gaussian assumption [27]–[29]. However, the LBP histogram has two properties that cause its underlying distribution to deviate far from Gaussian. 1) Non-negative constraint: each histogram bin is non-negative. 2) Simplex constraint: all histogram bins sum to 1. As a result, the LBP-histogram bin is bounded in the range of $[0, 1]$. In general, the expected value of LBP feature is close to 0. The lower bound at 0 has a severe truncation effect on the probability density function (PDF) of LBP feature so that it largely deviates from Gaussian.

Our objective is to find a transformation that produces a feature with a distribution close to Gaussian. Towards this end, we propose a Chi-squared transformation (CST) based on the following observations. On one hand, the LBP-histogram bins are severely unbalanced, i.e. the occurrence frequencies of frequent patterns are much higher than those of infrequent ones. On the other hand, the covariance matrix used in subspace approaches is constructed using Euclidean distance. Frequent patterns may overwhelm infrequent patterns when Euclidean distance is used. To enhance the discriminative power of infrequent patterns, distance measures other than Euclidean distance such as histogram intersection, Chi-squared distance and G-statistic are often employed [3], [8], [10]. The proposed transformation inherits the advantages of Chi-squared distance, which well handles the unbalanced nature of LBP histogram. More importantly, our analysis and experiments show that it produces a feature that fits better to Gaussian.

The proposed Chi-squared transformation converts original multi-class classification problem into a two-class one. This problem is highly asymmetric since the number of inter-class sample pairs is much larger than the number of intra-class pairs. To handle this asymmetric nature, we propose to use asymmetric PCA [28], [29] instead of using PCA to remove the unreliable dimensions in the CST feature space. We name it

Copyright (c) 2013 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

J. Ren is with BeingThere Centre, Institute of Media Innovation, Nanyang Technological University, 50 Nanyang Drive, Singapore 637553. (e-mail: jfren@ntu.edu.sg).

X. Jiang and J. Yuan are with School of Electrical & Electronics Engineering, Nanyang Technological University, 50 Nanyang Avenue, Singapore 639798. (e-mail: {exdjiang, jsyuan}@ntu.edu.sg)

as CST-APCA. It is validated extensively on face recognition, protein cellular classification and dynamic texture recognition. The proposed approach consistently enhances the recognition performance in all experiments.

II. The Proposed Approach

A. Modeling the PDF of LBP Histogram

Due to its nonnegative and simplex constraints, the PDF of LBP-histogram feature deviates far from Gaussian. The distributions of some LBP features[1] of the LFW database [30] are plotted in Fig. 1(a)-(d). They look very much like a truncated Gaussian distribution, i.e. truncation at the left-hand side. The fitted truncated Gaussian distributions are shown by red curves in Fig. 1. More evidences are given in Section III-A.

Thus, we use truncated Gaussian to model the PDF of LBP histogram. A truncated Gaussian random variable h has a Gaussian distribution $\mathcal{N}(\mu, \sigma^2)$ within the interval $[a, b]$ and zero PDF elsewhere, as shown in Fig. 2. Its PDF is given by

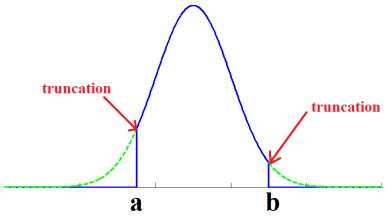


Fig. 2. Illustration of a truncated Gaussian distribution.

$$f(h;\mu,\sigma,a,b) = \begin{cases} \frac{\frac{1}{\sigma}\phi(\frac{h-\mu}{\sigma})}{\Phi(\frac{b-\mu}{\sigma})-\Phi(\frac{a-\mu}{\sigma})} & \text{if } h \in [a,b], \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

Here $\phi(x) = \frac{1}{\sqrt{2\pi}}\exp(-\frac{1}{2}x^2)$ is the standard Gaussian PDF, and $\Phi(x) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{x} e^{-t^2/2}dt$ is its cumulative distribution function. Denote $\alpha = \frac{a-\mu}{\sigma}$, $\beta = \frac{b-\mu}{\sigma}$. The mean and variance are:

$$\mu_T = \mu + \frac{\phi(\alpha)-\phi(\beta)}{\Phi(\beta)-\Phi(\alpha)}\sigma, \quad (2)$$

$$\sigma_T^2 = \sigma^2\left[1 + \frac{\alpha\phi(\alpha)-\beta\phi(\beta)}{\Phi(\beta)-\Phi(\alpha)} - \left(\frac{\phi(\alpha)-\phi(\beta)}{\Phi(\beta)-\Phi(\alpha)}\right)^2\right]. \quad (3)$$

Now we analyze the implications of nonnegative and simplex constraints of LBP histogram. Let $\mathbf{h} \in \mathbb{R}^d$ denote the LBP histogram, where $h_i \in [0,1], i = 1,2,\ldots,d$ is one of its bins. The truncations happen at both sides, i.e. $a = 0, b = 1$. Due to simplex constraint $\sum_{i=1}^{d} h_i = 1$, we have $\sum_{i=1}^{d}\mathbb{E}\{h_i\} = 1$. Due to non-negative constraint $h_i \geq 0, \forall i$, we have $\mathbb{E}\{h_i\} > 0$. Consider both constraints, and the fact that $d \gg 1$, $\mathbb{E}\{h_i\}$ is much closer to 0 than to 1. Thus, the truncation at $a = 0$ is significant, whereas the truncation at $b = 1$ is almost negligible since it is at the far right tail. These are evidenced from Fig. 1(a)-(d). Thus, $\phi(\beta) \approx 0$ and $\Phi(\beta) \approx 1$, and the simplified truncated Gaussian model has mean and variance:

$$\mu_S = \mu + \frac{\phi(\alpha)}{1-\Phi(\alpha)}\sigma, \quad (4)$$

$$\sigma_S^2 = \sigma^2\left[1 + \frac{\alpha\phi(\alpha)}{1-\Phi(\alpha)} - \left(\frac{\phi(\alpha)}{1-\Phi(\alpha)}\right)^2\right]. \quad (5)$$

Now $\alpha = -\frac{\mu}{\sigma}$. α can be used to measure how severe the left truncation is. A smaller α indicates that the truncation at $a = 0$ happens at the far left tail, which will alter the Gaussian distribution less significantly.

Given mean μ_S and variance σ_S^2 of a truncated Gaussian, it is difficult to derive μ and σ of the respective Gaussian distribution. Thus, we define a truncation effect measure α_S directly using μ_S and σ_S,

$$\alpha_S = \frac{a-\mu_S}{\sigma_S} = -\frac{-\alpha + \frac{\phi(\alpha)}{1-\Phi(\alpha)}}{\sqrt{1 + \frac{\alpha\phi(\alpha)}{1-\Phi(\alpha)} - \left(\frac{\phi(\alpha)}{1-\Phi(\alpha)}\right)^2}}. \quad (6)$$

α_S as the function of α in Eqn. (6) is plotted in Fig. 3. Clearly, α_S increases monotonically with α for $\alpha \leq 0$. Since $\alpha_S = f(\alpha)$ is a bijection function w.r.t. α for $\alpha \leq 0$, there is a unique inverse mapping from α_S to α, i.e. $\alpha = f^{-1}(\alpha_S)$, where $f^{-1}(\alpha_S)$ is a monotonically increasing function w.r.t. α_S. A larger α_S, or equivalently a smaller $|\alpha_S|$ indicates a more severe truncation.

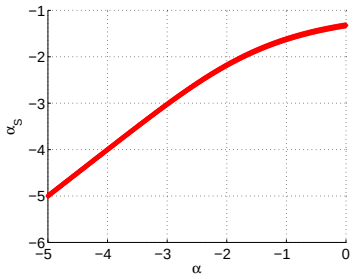


Fig. 3. α_S increases monotonically with α.

To study how severe the truncation is, we plot $|\alpha_S|$ for all LBP features of the LFW database in Fig. 4. For a better view, $|\alpha_S|$ is sorted in an ascending order. If $|\alpha_S| > 3$, it means that the truncation at $a = 0$ happens beyond three times of the standard derivation from the mean, and hence is negligible. However, from Fig. 4 we can see that for most LBP features $|\alpha_S| < 1$, which indicates a severe truncation.

In summary, the PDF of LBP feature deviates largely from Gaussian, as shown in Fig. 1(a)-(d). We model it as a truncated Gaussian, and show that $|\alpha_S| = \frac{|a-\mu_S|}{\sigma_S}$ can be used to measure how severe the truncation is. For most LBP features, $|\alpha_S| < 1$, and hence their PDFs are severely truncated.

B. Proposed Chi-Squared Transformation

LBP histogram has been used as a robust image representation for many recognition tasks. In real applications, it often produces a high-dimensional feature vector, which adversely

[1]We use $LBP_{8,2}^{u2}$, a uniform LBP with 59 histogram bins extracted using 8 neighbors evenly spaced on a circle of radius 2.

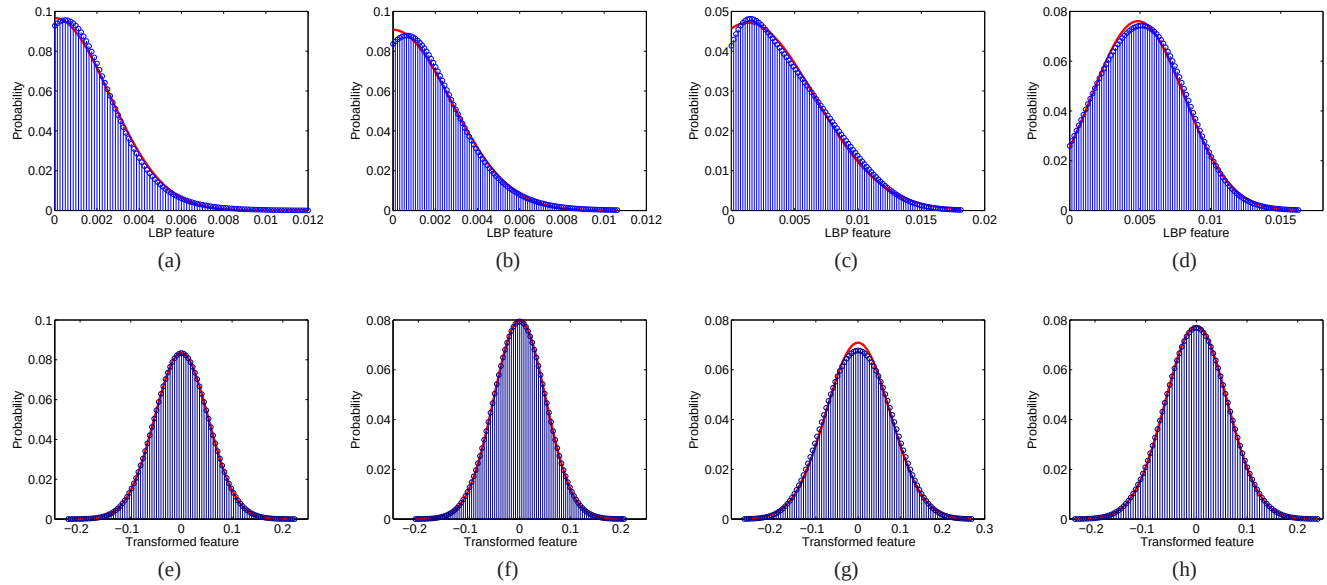


Fig. 1. Comparison of the distributions of the features before and after the proposed Chi-squared transformation. (a)-(d) The distributions of 4 LBP features and the fitted truncated Gaussian distributions in red. (e)-(h) The distributions of the features derived by the proposed transformation on the features with distributions shown in (a)-(d) and the fitted Gaussian distributions in red.

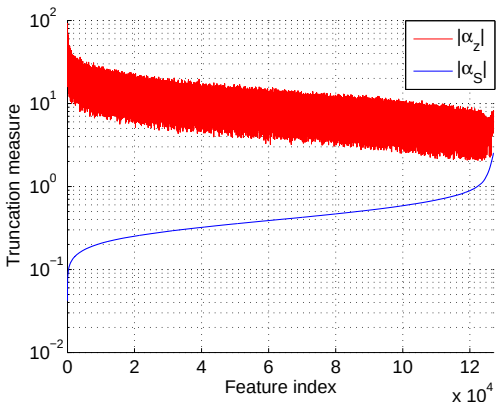


Fig. 4. Plot of truncation measure $|\alpha_S|$ of the LBP features and $|\alpha_z|$ of the features after the proposed Chi-squared transformation. For a better view, $|\alpha_S|$ is sorted in an ascending order and plotted together with respective $|\alpha_z|$. $|\alpha_z|$ is consistently larger than $|\alpha_S|$, which indicates that the truncation is much less severe after the proposed transformation.

affects the robust classification. It is hence not a surprise that many recent approaches utilize linear subspace methods to remove the unreliable dimensions in LBP histogram or to derive a compact feature representation [12], [21]–[26]. However, subspace approaches utilizing up to the second-order statistics are only optimal when the underlying distribution is Gaussian [28], [29]. Under Gaussian assumption, the maximum a posterior (MAP) rule that minimizes the classification error is simplified to evaluating the following discriminant function [29]:

$$g_i(\mathbf{h}) = -\frac{1}{2}(\mathbf{h}-\bar{\mathbf{h}}_i)^T\boldsymbol{\Sigma}_i^{-1}(\mathbf{h}-\bar{\mathbf{h}}_i) + \theta_i, \qquad (7)$$

where $\bar{\mathbf{h}}_i$ is the class mean, $\boldsymbol{\Sigma}_i$ is the class covariance matrix and θ_i is a threshold to control the error rate of class i at a price of others. It is basically a minimum Mahalanobis distance classifier (mMDC). As the covariance matrix $\boldsymbol{\Sigma}_i$ of individual class may not be reliably estimated in many applications, a pooled covariance matrix called within-class scatter matrix is often used instead in Eqn. (7). Unfortunately, the Mahalanobis distance cannot be reliably evaluated in the high-dimensional space due to limited training samples. Thus, various subspace approaches are developed to solve this problem. PCA+LDA [31] removes dimensions corresponding to unreliable small eigenvalues so that the Mahalanobis distance can be evaluated reliably in a small PCA+LDA subspace. Bayesian Maximum Likelihood approaches [32], [33] partition the feature space into two subspaces where the Mahalanobis distance is evaluated reliably in one and the Euclidian distance is applied in the other to circumvent the unreliable evaluation of the Mahalanobis distance. The Euclidean distance can be viewed as a degenerated Mahalanobis distance, where the covariance matrix is assumed to be a scalar matrix. ERE approach [27] regularizes the covariance matrix before computing the Mahalanobis distance to attenuate this problem.

However, it is shown in the previous section that the PDF of LBP feature deviates far from Gaussian. Therefore, Mahalanobis distance and Euclidian distance are not optimal for LBP features. This is evidenced by the fact that other distance measures such as Chi-squared distance, histogram intersection and G-statistic [3], [8], [10] have been widely used in the LBP feature space $\mathbb{R}^d$ and demonstrated a superior performance to Euclidian distance. (Mahalanobis distance cannot be applied in $\mathbb{R}^d$ because $\boldsymbol{\Sigma}_i$ is singular.) To utilize linear subspace approach in solving the high-dimensionality problem of LBP feature, we propose to transform it to a new feature space so that the resulting feature fits better to Gaussian distribution.

Given two samples of LBP features represented by $\mathbf{x}, \mathbf{y}$, we define a new feature vector $\mathbf{z} = \{z_1, z_2, \ldots, z_d\}$, where each

element z_i is:

$$z_i = \frac{x_i - y_i}{\sqrt{x_i + y_i}}. \tag{8}$$

As the l_2-norm of $\mathbf{z}$ is the Chi-squared distance between feature vectors $\mathbf{x}, \mathbf{y}$,

$$\mathbf{z}^T\mathbf{z} = \sum_{i=1}^{d} \frac{(x_i - y_i)^2}{x_i + y_i}, \tag{9}$$

we name the proposed transformation in Eqn. (8) as Chi-squared transformation.

Similar to Eqn. (6), the following criterion is defined for z to measure how severe the truncation is:

$$|\alpha_z| = \frac{|a_z - \mu_z|}{\sigma_z}, \tag{10}$$

where a_z denotes where the truncation happens. μ_z and σ_z are mean and standard deviation of z, respectively.

Suppose that we have a LBP feature h that has a truncated Gaussian distribution with mean μ_S, and two independent samples x, y drawn from this distribution. We have $\mu_z = \mathbb{E}\left\{\frac{x-y}{\sqrt{x+y}}\right\} = \mathbb{E}\left\{\frac{x}{\sqrt{x+y}}\right\} - \mathbb{E}\left\{\frac{y}{\sqrt{x+y}}\right\}$. Since x and y are drawn from the same distribution and independent, $\frac{x}{\sqrt{x+y}}$ and $\frac{y}{\sqrt{x+y}}$ have the same PDF. Thus, $\mu_z = 0$ as $\mathbb{E}\left\{\frac{x}{\sqrt{x+y}}\right\} = \mathbb{E}\left\{\frac{y}{\sqrt{x+y}}\right\}$. Intuitively, for every sample pair $\{x, y\}$, there is a corresponding sample pair $\{y, x\}$, i.e. $\tilde{z} = \frac{y-x}{\sqrt{x+y}}$. The summation of z and $\tilde{z}$ is zero. Thus, $\mu_z = 0$. Since $x, y \geq 0$, $\mathbb{E}\left\{\frac{4xy}{x+y}\right\} \geq 0$, we have $\sigma_z^2 = \mathbb{E}\left\{\frac{(x-y)^2}{x+y}\right\} = \mathbb{E}\left\{x + y - \frac{4xy}{x+y}\right\} \leq \mathbb{E}\{x+y\} = 2\mu_S$.

As $x, y \in [0, 1]$, the PDF of z is truncated in $[-1, 1]$, and hence $|a_z| = 1$. Plug a_z, μ_z and σ_z into Eqn. (10), we have:

$$|\alpha_z| = \frac{1}{\sigma_z} \geq \frac{1}{\sqrt{2\mu_S}}. \tag{11}$$

For LBP features, we have $\sum_{i=1}^{d} \mathbb{E}\{h_i\} = 1, \mathbb{E}\{h_i\} \geq 0$, $\forall i$ and $d \gg 1$. Thus, $\mathbb{E}\{h_i\} \ll 1$, or equivalently $\mu_S \ll 1$. It is easy to satisfy that $\frac{1}{\sqrt{2\mu_S}} > 3$ [2], or $|\alpha_z| > 3$, which means that the truncation for the transformed feature happens beyond three times of the standard derivation from the mean, and hence is almost negligible. We apply the proposed transformation on the high-dim LBP features of the LFW database, and plot $|\alpha_z|$ for the resulting features in Fig. 4. It shows that $|\alpha_z|$ is much larger than $|\alpha_S|$. The proposed transformation greatly reduces the truncation effect.

The PDFs of the features transformed from features in Fig. 1(a)-(d) are shown in Fig. 1(a)-(d), respectively, in which the truncations happen at the far tails, and are almost negligible. We fit the transformed features into Gaussian distribution using maximum-likelihood estimation, and overlay the fitted distribution in the same figures. Visually, the transformed features fit well to Gaussian. In the next section, we further analyze the normality using moments.

[2]For example, for 59.13%, 74.66% and 94.35% of the high-dim LBP features of the LFW dataset, we have $\mu_S < 0.01, 1/59$ and $1/18$, respectively.

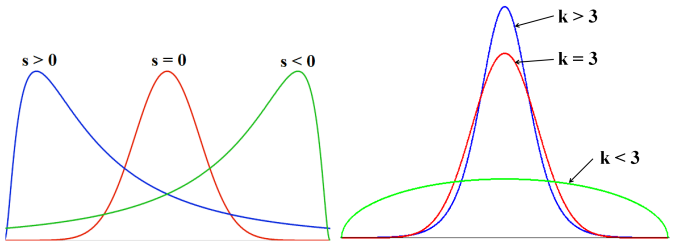
C. Normality Analysis Using Moments

Moments have been widely used to test the normality of a distribution [34]–[37]. Among them, skewness and kurtosis are the simplest but effective measures. As shown in Fig. 5, the skewness is a measure of the asymmetry of a PDF about its mean, and the kurtosis is a measure of the "peakedness" of a distribution, which are respectively defined as:

$$s = m_3/m_2^{3/2}, \tag{12}$$

$$k = m_4/m_2^2, \tag{13}$$

where $m_r = \mathbb{E}\{(h - \mu_h)^r\}$ is the r-th order moment, and μ_h is the mean.



(a) Skewness (b) Kurtosis

Fig. 5. Illustration of the skewness and the kurtosis.

We analyze the normality of the features before and after the proposed CST using Jarque-Bera test (JBTest) [34], which utilizes both skewness s and kurtosis k:

$$J = \frac{n}{6}\left(s^2 + \frac{(k-3)^2}{4}\right), \tag{14}$$

where n is the number of samples. For a Gaussian distribution, $s = 0$ and $k = 3$. Deviation from these two values indicates non-normality. Thus, a smaller J indicates a better fit to Gaussian distribution.

We first calculate JBtest statistic J_t for LBP feature under truncated Gaussian model. Its skewness and kurtosis are a function of α:

$$s_t = \frac{\frac{2\phi(\alpha)+\alpha^2\phi(\alpha)}{1-\Phi(\alpha)}}{\left(1 + \frac{\alpha\phi(\alpha)}{1-\Phi(\alpha)}\right)^{3/2}}, \tag{15}$$

$$k_t = \frac{3\left(1 + \frac{\alpha\phi(\alpha)+\frac{1}{3}\alpha^3\phi(\alpha)}{1-\Phi(\alpha)}\right)}{\left(1 + \frac{\alpha\phi(\alpha)}{1-\Phi(\alpha)}\right)^2}, \tag{16}$$

where $\alpha = (a - \mu)/\sigma = -\mu/\sigma$, as the truncation occurs at $a = 0$. The derivation is given in Appendix A.

For JBtest statistic J_z of the transformed feature z, its skewness $s_z = 0$ and kurtosis k_z is also a function of α. The derivation of this function is given in Appendix B.

We plot J_t and J_z vs. α in Fig. 6, since they are both a function of α. As we derive JBTest statistic using modeled distribution rather than using samples, n is set to 1 for both J_t and J_z for simplicity. J_z is consistently smaller than J_t. Thus, we conclude that the proposed transformation produces a feature that fits better to Gaussian distribution under JBtest.

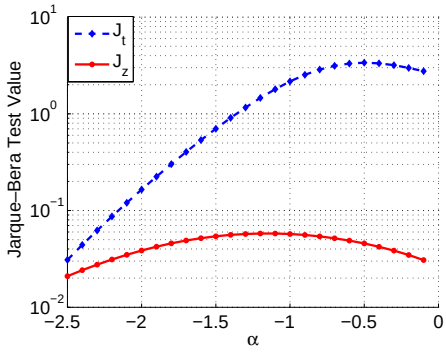


Fig. 6. Comparison of the Jarque-Bera test statistic J_t before and J_z after the proposed Chi-squared transformation.

D. Chi-Squared-Transformed Subspace Analysis

The transformed feature $\mathbf{z}$ is defined on a pairwise basis, i.e. each element $z_i = \frac{x_i - y_i}{\sqrt{x_i + y_i}}$ is defined using a sample pair $\{\mathbf{x}, \mathbf{y}\}$. It naturally transfers the original classification problem into a two-class one: class Ω_I containing the sample differences from the same class, and class Ω_E containing the sample differences from different classes. It is easy to show that both classes have zero mean. The covariance matrices for class Ω_I and Ω_E are respectively constructed as $\mathbf{\Sigma}_I = \frac{1}{N_I}\sum_{\mathbf{z}\in\Omega_I}\mathbf{z}\mathbf{z}^T$ and $\mathbf{\Sigma}_E = \frac{1}{N_E}\sum_{\mathbf{z}\in\Omega_E}\mathbf{z}\mathbf{z}^T$, where N_I and N_E are the number of samples for class Ω_I and Ω_E, respectively.

Eqn. (7) that minimizes the classification error decides that a sample $\mathbf{z}$ belongs to class Ω_I if

$$\mathbf{z}^T\mathbf{\Sigma}_E^{-1}\mathbf{z} - \mathbf{z}^T\mathbf{\Sigma}_I^{-1}\mathbf{z} > \theta, \tag{17}$$

where θ is a threshold. Let λ_k^I, λ_k^E and Φ_k^I, Φ_k^E denote the eigenvalues and eigenvectors of $\mathbf{\Sigma}_I$, $\mathbf{\Sigma}_E$, respectively. Let p_k and q_k denote the projection of $\mathbf{z}$ on Φ_k^I and Φ_k^E, respectively. The decision rule (17) is simplified to

$$\sum_{k=1}^{d}\frac{p_k^2}{\lambda_k^E} - \sum_{k=1}^{d}\frac{q_k^2}{\lambda_k^I} > \theta. \tag{18}$$

Eigenvalues λ_k^E and λ_k^I are an estimate of the class true (ensemble) variance based on available training data. The decision rule will overfit the training samples, and hence result in a poor generalization performance if these eigenvalues differ significantly from the ensemble variances.

As pointed out in [27]–[29], small eigenvalues λ_k^E and λ_k^I are biased downwards severely, and result in misclassification. PCA applies eigen-decomposition on the total scatter matrix $\mathbf{\Sigma}_t = \mathbf{\Phi}_t\mathbf{\Lambda}_t\mathbf{\Phi}_t^T$, where $\mathbf{\Sigma}_t$ is calculated by

$$\mathbf{\Sigma}_t = \frac{N_I}{N_I+N_E}\mathbf{\Sigma}_I + \frac{N_E}{N_I+N_E}\mathbf{\Sigma}_E. \tag{19}$$

Here, the between-class scatter matrix is zero as both classes have zero mean. By removing the subspace of $\mathbf{\Sigma}_t$ corresponding to small eigenvalues of λ_k^E and λ_k^I, PCA improves the classification accuracy [28], [29]. However, $\mathbf{\Sigma}_I$ is less reliable than $\mathbf{\Sigma}_E$ as $\mathbf{\Sigma}_I$ is estimated using much less samples. Thus, PCA on $\mathbf{\Sigma}_t$ does not effectively remove the unreliable dimensions as it puts a small weight $N_I/(N_I+N_E)$ on $\mathbf{\Sigma}_I$. The problem is severe here as the two classes constructed by the proposed transform are highly asymmetric.

To tackle this problem, we construct an asymmetric pooled covariance matrix by

$$\mathbf{\Sigma}_\gamma = \gamma\mathbf{\Sigma}_I + (1-\gamma)\mathbf{\Sigma}_E, \tag{20}$$

where γ is the weighting factor of $\mathbf{\Sigma}_I$. If $\gamma = N_I/(N_E+N_I)$, $\mathbf{\Sigma}_\gamma$ is the same as $\mathbf{\Sigma}_t$ in PCA. If we set $\gamma > \frac{N_I}{N_I+N_E}$, more dimensions of small λ_k^I will be removed than that of PCA. This leads to more effective dimensionality reduction than PCA for enhancing the classification accuracy. To remove the unreliable dimensions, eigen-decomposition is applied on $\mathbf{\Sigma}_\gamma$,

$$\mathbf{\Sigma}_\gamma = \mathbf{\Phi}_\gamma\mathbf{\Lambda}_\gamma\mathbf{\Phi}_\gamma^T, \tag{21}$$

where $\mathbf{\Phi}_\gamma$ and $\mathbf{\Lambda}_\gamma$ correspond to the eigenvectors and eigenvalues of $\mathbf{\Sigma}_\gamma$, respectively. We select eigenvectors $\mathbf{\Phi}_{\gamma,m}$ corresponding to m largest eigenvalues, and project $\mathbf{z}$ into the subspace as:

$$\hat{\mathbf{z}} = \mathbf{\Phi}_{\gamma,m}^T\mathbf{z}. \tag{22}$$

After removing the unreliable dimensions corresponding to small eigenvalues that are biased downwards, the bias problem of eigenspectrum is not fully solved, i.e. the large eigenvalues are still biased upwards [28]. The eigenspectrum of $\mathbf{\Sigma}_I$ is estimated less reliably than $\mathbf{\Sigma}_E$ using much less samples. Thus, the remaining large eigenvalues of $\mathbf{\Sigma}_I$ are biased upwards more severely than $\mathbf{\Sigma}_E$. To correct this, we use the following classifier:

$$\hat{\mathbf{z}}^T\hat{\mathbf{\Sigma}}_E^{-1}\hat{\mathbf{z}} - \hat{\mathbf{z}}^T(\psi\hat{\mathbf{\Sigma}}_I)^{-1}\hat{\mathbf{z}} > \hat{\theta}, \tag{23}$$

where $\hat{\mathbf{\Sigma}}_E = \mathbf{\Phi}_{\gamma,m}^T\mathbf{\Sigma}_E\mathbf{\Phi}_{\gamma,m}$, $\hat{\mathbf{\Sigma}}_I = \mathbf{\Phi}_{\gamma,m}^T\mathbf{\Sigma}_I\mathbf{\Phi}_{\gamma,m}$, and ψ is a weighting constant. The value of ψ depends on how far the large eigenvalues of $\mathbf{\Sigma}_I$ deviate from their ensemble variances compared with those of $\mathbf{\Sigma}_E$.

We name the proposed approach as CST-APCA, i.e. first apply the proposed Chi-squared transformation on the LBP histogram to derive a feature that fits better to Gaussian distribution, and then apply asymmetric PCA on the resulting feature to remove unreliable dimensions. The pseudo-code is given in Algorithm 1. The proposed transformation does not significantly increase the computational cost or memory cost. Compared with traditional PCA-based approaches, it mainly requires some additional memory and computational cost to perform Chi-squared transformation in Eqn. (8).

III. Experimental Results

We first conduct Chi-square goodness-of-fit (Chi2GOF) [38] test to further validate that the LBP features fit to truncated Gaussian distribution. Then, we compare the features before and after the proposed transformation under Jarque-Bera normality test. Finally, we compare the proposed approach with 10 approaches on 8 datasets in classifying face, protein cellular and dynamic texture, to validate the proposed transformation.

The nearest-neighbor classifier (NNC) with Euclidean distance in the original LBP feature space is used as the baseline algorithm, denoted as NNC-ED. As Euclidean distance may not well handle the infrequent patterns of LBP histogram,

Algorithm 1 Asymmetric PCA on the Chi-squared-transformed features

Input: Training samples $\{\mathbf{h}_i \in \mathbb{R}^d\}$ and corresponding class labels $\{\omega_i\}$, $i = 1, 2, \ldots, N$, testing sample $\mathbf{h}_t \in \mathbb{R}^d$
Parameters: Weighting constants γ, ψ, feature dimension m.
Output: Class label ω_t of $\mathbf{h}_t$
Chi-squared transformation for training data:
for i = 1:N **do**
for j = 1:N **do**
a. Derive the Chi-squared-transformed feature vector $\mathbf{z}_{i,j} \in \mathbb{R}^d$, where each element of $\mathbf{z}_{i,j}$ is $z_{i,j,k} = \frac{h_{i,k}-h_{j,k}}{\sqrt{h_{i,k}+h_{j,k}}}$, $k = 1, 2, \ldots, d$, as in Eqn. (8).
b. If $\omega_i = \omega_j$, $\mathbf{z}_{i,j} \in \Omega_I$ else $\mathbf{z}_{i,j} \in \Omega_E$.
end for
end for
Asymmetric PCA on training data:
1. Derive covariance matrices for class Ω_I and Ω_E as $\mathbf{\Sigma}_I = \frac{1}{N_I}\sum_{\mathbf{z}\in\Omega_I}\mathbf{z}\mathbf{z}^T$ and $\mathbf{\Sigma}_E = \frac{1}{N_E}\sum_{\mathbf{z}\in\Omega_E}\mathbf{z}\mathbf{z}^T$, respectively. Here $\mathbf{z}$ is a simplified notation of $\mathbf{z}_{i,j}$.
2. Construct asymmetric pooled covariance matrix $\mathbf{\Sigma}_\gamma = \gamma\mathbf{\Sigma}_I + (1-\gamma)\mathbf{\Sigma}_E$ as in Eqn. (20).
3. Perform eigen-decomposition on $\mathbf{\Sigma}_\gamma$, $\mathbf{\Sigma}_\gamma = \mathbf{\Phi}_\gamma\mathbf{\Lambda}_\gamma\mathbf{\Phi}_\gamma^T$ as in Eqn. (21).
4. Compute covariance matrices in the subspace spanned by eigenvectors $\mathbf{\Phi}_{\gamma,m}$ corresponding to m leading eigenvalues of $\mathbf{\Sigma}_\gamma$ as: $\hat{\mathbf{\Sigma}}_E = \mathbf{\Phi}_{\gamma,m}^T\mathbf{\Sigma}_E\mathbf{\Phi}_{\gamma,m}$, $\hat{\mathbf{\Sigma}}_I = \mathbf{\Phi}_{\gamma,m}^T\mathbf{\Sigma}_I\mathbf{\Phi}_{\gamma,m}$.
Classification of all testing samples:
For every query sample $\mathbf{h}_t$:
for i=1:N **do**
a. Perform Chi-squared transformation using sample pair $(\mathbf{h}_i, \mathbf{h}_t)$ to obtain feature $\mathbf{z}_{i,t}$ as defined in Eqn. (8).
b. Project $\mathbf{z}_{i,t}$ into the subspace as $\hat{\mathbf{z}}_{i,t} = \mathbf{\Phi}_{\gamma,m}^T\mathbf{z}_{i,t}$.
c. Compute the difference between distances from two classes $S_i = \hat{\mathbf{z}}_{i,t}^T\hat{\mathbf{\Sigma}}_E^{-1}\hat{\mathbf{z}}_{i,t} - \hat{\mathbf{z}}_{i,t}^T(\psi\hat{\mathbf{\Sigma}}_I)^{-1}\hat{\mathbf{z}}_{i,t}$.
end for
$i^* = \mathrm{argmax}_i\, S_i$, and output $\omega_t = \omega_{i^*}$.

other distance measures such as Chi-squared distance are often used [3], [8], [10]. Hence, the second compared algorithm is NNC + Chi-squared distance, denoted as NNC-Chi2D. As the dimensionality of LBP feature is often high, Mahalanobis distance cannot be directly calculated in the original feature space, and PCA is often applied to reduce the dimensionality [12], [21]–[26]. Thus, the third one is PCA-mMDC.

We also compare the proposed algorithm with other state-of-the-art approaches. Power mean support vector machine (PmSVM) [39] has achieved a superior performance in classifying LBP features. It is claimed in [39] that when features are histograms, SVMs of additive kernels are most effective. PmSVM utilizes a special family of additive kernels, i.e.

$$M_p(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{d}\left(\frac{x_i^p + y_i^p}{2}\right)^{1/p}, \tag{24}$$

for any $p < 0$. It includes some popular additive kernels, e.g. χ^2 kernel if $p = -1$ and histogram-intersection kernel if $p = -\inf$. We compare the proposed approach with both, i.e. PmSVM-Chi2 and PmSVM-HI. [3] As suggested in [39], we use $p = -8$ for PmSVM-HI, and cost parameter $C = 0.01$ in all experiments. Very recently, a simple transform that square-roots every feature value [40] is shown to improve the fitness to Gaussian for video features. We use it with PCA-mMDC, and name it as SR-PCA-mMDC. [4] Besides these, the proposed approach is also compared with other state-of-the-art solutions to dynamic texture recognition.

The proposed transformation can be also applied to other histogram features such as HOG features [41] to improve its fitness to Gaussian. To verify this, we conduct a comparison experiment on the LFW dataset using HOG features.

A. Chi-Square Goodness-of-Fit Test

We conduct Chi2GOF test [38] on 7 datasets to validate that LBP features fit to truncated Gaussian distribution, and the transformed features fit to Gaussian distribution. We fit LBP features to the following distributions through maximum-likelihood estimation: Rice, Rayleigh, Gaussian and truncated Gaussian. The first two are often used to model non-negative random variables. We then conduct Chi2GOF test to check whether they fit to one of these distributions. For example, the null hypothesis for Gaussian distribution is that the underlying distribution is Gaussian. A feature passes the test if the null hypothesis cannot be rejected at the 5% significance level. More specifically, Chi-square test [38] is performed by grouping data into bins, calculating observed and expected bin counts, and computing the test statistic

$$\chi^2 = \sum_{i=1}^{d}(O_i - E_i)^2/E_i, \tag{25}$$

where O_i and E_i are observed and expected bin counts, respectively.

The pass rate [5] is used as the evaluation metric, and results are shown in Table I. For all datasets, truncated Gaussian distribution consistently has the highest pass rate, and they are all about 80% and some even higher. This validates that LBP features fit well to truncated Gaussian.

We further conduct Chi2GOF test [38] for the transformed features on Gaussian distribution. The pass rates are shown in the last row of Table I. All datasets have high pass rate, which indicates that the transformed features fit well to Gaussian distributions under Chi-square test.

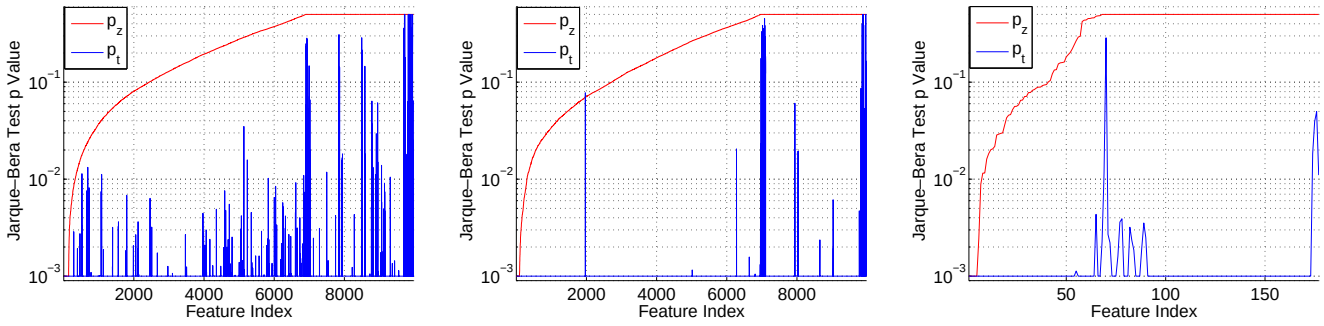
B. Jarque-Bera Normality Test

In Section II-C, we show that theoretically the proposed transformation results in a feature that fits better to Gaussian distribution under JBTest [34]. Now, we conduct JBTest on real images: AR [42], O2FN [43] and 2D Hela [44] databases. The results on other datasets are similar to these three, and

[3] We use the source codes of power mean SVM downloaded from https://sites.google.com/site/wujx2001/home/power-mean-svm.
[4] Before applying PCA-mMDC, every feature value is square-rooted.
[5] The pass rate is defined as the number of features passing the test normalized by the total number of features.

TABLE I
PASS RATE OF LBP FEATURE AT 5% SIGNIFICANCE LEVEL IN CHI-SQUARE GOODNESS-OF-FIT TEST FOR RICE, RAYLEIGH, TRUNCATED GAUSSIAN AND GAUSSIAN DISTRIBUTIONS (FIRST 4 ROWS), AND THAT OF TRANSFORMED FEATURES FOR GAUSSIAN DISTRIBUTION.

Distributions	AR [42]	O2FN [43]	LFW Baseline LBP [25], [30]	LFW High-Dim LBP [25], [30]	2D Hela [44]	UCLA [45], [46]	DynTex++ [47]
Rice	0.06%	0.05%	0.00%	0.00%	0.00%	0.00%	0.00%
Rayleigh	5.24%	5.86%	1.78%	1.45%	0.00%	1.82%	13.15%
Truncated Gaussian	88.91%	79.10%	82.58%	84.63%	91.53%	85.55%	80.34%
Gaussian	5.67%	3.39%	2.59%	3.09%	2.82%	4.69%	8.33%
Gaussian (after CST)	90.02%	82.43%	86.97%	88.86%	92.66%	88.22%	83.72%



(a) The AR database (b) The O2FN database (c) The 2D Hela database

Fig. 7. Comparison of the p-value p_t and p_z of Jarque-Bera test for features before and after the proposed transformation on (a) the AR database, (b) the O2FN database, (c) the 2D Hela database. For a better view, p_z is sorted in an ascending order.

hence omitted. We plot p_t and p_z in Fig. 7, which represent p-values of JBTest for features before and after the proposed transformation, respectively. A larger p-value indicates a better fit to Gaussian. For a better view, p_z is sorted in an ascending order, and plotted together with corresponding p_t.

We can see that most p_t are very small, which indicates that LBP features deviate far from Gaussian distribution. After the proposed transformation, p_z are significantly larger than p_t on all three databases. Indeed, 0.70%/87.01%, 0.44%/85.37% and 0.56%/87.57% of features before/after the transformation pass JBTest at 5% significance level on AR, O2FN and 2D Hela databases, respectively. This validates that the proposed transformation produces a feature that fits better to Gaussian distribution under Jarque-Bera test.

C. Face Recognition on the AR Database

We normalize the image to 150×130 pixels, apply an ellipse mask to remove background pixels and divide it into patches of 21×18 pixels. The feature descriptor is $LBP^{u2}_{8,2}$. Initial experiments show that a better performance is achieved when there is 50% overlapping between neighboring patches. In total, there are $13 \times 13 = 169$ patches. The concatenated feature has $59 \times 169 = 9971$ dimensions. 75 subjects are chosen from the AR database [42], each with two sections. Each section contains 7 images: one neutral, 3 with different facial expressions and 3 in different illumination conditions. We use the first section for training and the second for testing.

The comparison with other approaches is shown in Table II. NNC-Chi2D performs better than NNC-ED, which indicates that Chi-squared distance is a better measure for LBP features. PCA-mMDC performs even worse than NNC-Chi2D, mainly because of non-normality of LBP features. SR-PCA-mMDC outperforms PCA-mMDC, as it improves the fitness to Gaussian by square-rooting features. The proposed CST-APCA performs much better than PCA-mMDC, which indeed demonstrates the superiority of the proposed transformation. Surprisingly, PmSVM-Chi2 and PmSVM-HI achieve an error-free classification on this dataset.

TABLE II
COMPARISON OF ERROR RATES FOR DIFFERENT APPROACHES ON THE AR DATABASE.

Algorithms	Error Rate
NNC-ED	3.43%
NNC-Chi2D	1.33%
PCA-mMDC	2.48%
SR-PCA-mMDC [40]	1.33%
PmSVM-Chi2 [39]	**0.00%**
PmSVM-HI [39]	**0.00%**
Proposed Approach	**0.38%**

D. Face Recognition on the O2FN Mobile Database

The O2FN database [43] contains 50 subjects, each with 40 images of 144×176 pixels. The images are captured using the front camera of a mobile phone with native settings, and hence they are of low image quality and severely distorted by noise. They also contain some pose variations and significant illumination variations. We use first 20 images of each subject for training and the rest for testing. LBP features are extracted similarly as for the AR database, which have 9971 dimensions.

The comparison with others is shown in Table III. NNC-Chi2D slightly reduces the error rate compared with NNC-ED. As images are noisy, many dimensions of LBP features are not

reliable. PCA-mMDC greatly reduces the classification error by removing these unreliable dimensions. SR-PCA-mMDC further boosts the performance by improving the fitness to Gaussian. On top of these, the proposed CST-APCA significantly outperforms all compared approaches.

TABLE III
COMPARISON OF ERROR RATES FOR DIFFERENT APPROACHES ON THE O2FN DATABASE.

Algorithms	Error Rate
NNC-ED	33.20%
NNC-Chi2D	30.00%
PCA-mMDC	5.00%
SR-PCA-mMDC [40]	2.80%
PmSVM-Chi2 [39]	4.00%
PmSVM-HI [39]	4.10%
Proposed Approach	**1.50%**

E. Face Recognition on the LFW Database

The LFW database has been widely used as a benchmark dataset to evaluate face recognition algorithm [25], [48], [49]. It consists of 13,233 images of 5,749 subjects. All images are collected from the internet with large pose, illumination and facial-expression variations, as well as occlusions.

In [25], it is shown that high-dimensional LBP features achieve a better performance than baseline LBP features. In this paper, we report the results on both datasets. [6] In image understanding and recognition, histogram-like features are widely used, e.g. LBP histogram, histogram of oriented gradient (HOG), scale-invariant feature transform, color histogram and gray-level co-occurrence histogram. The proposed transformation can improve the performance of other histogram-like features. To verify this, we conduct experiments on HOG features. These features are extracted as follow:

Baseline LBP Feature: The image is normalized to 100×100 pixels by an affine transformation derived using 5 landmarks (two eyes, nose and two mouth tips). Then, it is partitioned into 10×10 non-overlapped patches. An LBP histogram is extracted from each patch. All histograms are concatenated to form the final feature vector of $59 \times 10^2 = 5900$ dimensions.

Baseline HOG Feature: The image is normalized to 128×128 pixels. A sliding window of 16×16 pixels is used to scan the whole image, at the stripe size of 8 pixels. Each sliding window is divided into 4 cells. A HOG histogram of 9 bins is extracted from each cell. The histograms of one window are $l2$-normalized. The concatenated feature vector has $9 \times 4 \times 15^2 = 8100$ dimensions.

High-Dim LBP Feature: The normalized images are resized to five scales, i.e. the side lengths in each scale are 300, 212, 150, 106 and 75. In each scale, an image cell of 40×40 pixels is extracted from each of 27 facial landmarks, and further divided into non-overlapped patches of 10×10 pixels. An LBP histogram is extracted from each patch. The concatenated feature vector has $59 \times 27 \times 5 \times 16 = 127,440$ dimensions. This high-dim LBP feature has achieved the state-of-the-art performance on the LFW dataset [25], but also imposes a great challenge on both memory and computational complexity.

[6]Features can be downloaded from http://home.ustc.edu.cn/∼chendong/.

We report the averaged results over 10 folds of View 2. More precisely, 3000 positive pairs (two images of the same subject) and 3000 negative pairs (two images of different subjects) are divided into 10 subsets. In each fold, 9 subsets are used as the training set and the hold-out subset is used for testing. The task is to determine whether an image pair comes from the same subject or not. We follow the LFW's restricted protocol. No outside dataset is used for training. We test several subspace approaches with the proposed transformation. ERE approach [27] performs best, and hence we report this result. The results are summarized in Table IV. The results of Joint Bayesian [25] are copied here for comparison.

TABLE IV
COMPARISON OF ERROR RATES FOR DIFFERENT APPROACHES ON THE LFW DATABASE.

Algorithms	Error Rate		
	Baseline LBP	Baseline HOG	High-Dim LBP
NNC-ED	32.70%	35.95%	28.70%
NNC-Chi2D	28.70%	34.12%	26.35%
PCA-mMDC	25.92%	30.27%	21.05%
SR-PCA-mMDC [40]	19.43%	27.62%	15.88%
PmSVM-Chi2 [39]	21.17%	27.02%	13.67%
PmSVM-HI [39]	21.20%	26.88%	13.43%
Joint Bayesian [25]	19.95%	-	15.92%
Proposed Approach	**15.42%**	**20.00%**	**9.50%**

For all approaches, baseline LBP performs poor than high-dim LBP, but better than baseline HOG. NNC-Chi2D consistently outperforms NNC-ED. PCA-mMDC marginally reduces the error rate compared with these two. SR-PCA-mMDC further reduces the error, and achieves even slightly better performance than joint Bayesian approach [25]. PmSVMs perform worse than SR-PCA-mMDC on baseline LBP, but better on baseline HOG and high-dim LBP. It shows that PmSVMs are more effective on higher-dimensional features. The proposed approach further reduces the error rates on all three datasets. Especially on high-dim LBP feature, it significantly decreases the error rate of joint Bayesian [25], the published best results, from 15.92% to 9.50%.

F. Protein Cellular Classification on the 2D Hela Database

Protein cellular classification is useful for characterizing newly discovered genes. The 2D Hela database [44] contains 862 single-cell images (16-bit gray image of size 382×382 pixels). It has 10 classes and each with more than 70 images.

We extract LBP features at multiple scales, i.e. $LBP_{8,1}$, $LBP_{8,2}$ and $LBP_{8,3}$, same as in [50]. At each scale, the histogram has $2^8 = 256$ bins. Protein cellular only occupies a small portion of the image. The background pixels are harmful for reliable classification, and mainly fall into bin 256. Thus, bin 256 at each scale is removed. The histograms are concatenated to form the final feature vector of 765 dimensions. We randomly choose 80% of the images for training and the rest for testing. The experiments are repeated five times and the average performance is reported in Table V.

As shown in Table V, NNC-Chi2D significantly outperforms NNC-ED in the LBP feature space. PCA-mMDC greatly

TABLE V
COMPARISON OF ERROR RATES FOR DIFFERENT APPROACHES ON THE 2D HELA DATABASE.

Algorithms	Error Rate
NNC-ED	30.81%
NNC-Chi2D	24.53%
PCA-mMDC	13.49%
SR-PCA-mMDC [40]	9.42%
PmSVM-Chi2 [39]	11.63%
PmSVM-HI [39]	12.91%
Proposed Approach	**8.60%**

reduces the error rate, which shows the effectiveness of subspace approaches in removing unreliable dimensions of LBP features. SR-PCA-mMDC further boosts the performance compared with PCA-mMDC by improving the fitness to Gaussian. On top of these, the proposed CST-APCA further reduces the error rate to 8.60%. The transformed features fit better to Gaussian distribution, and hence yield a better performance.

G. Dynamic Texture Recognition on the UCLA Database

Dynamic texture (DT) is a sequence of images of moving scenes that exhibit certain stationarity properties in time [45], [46]. The UCLA database [51], [52] has been widely used as a benchmark dataset for DT recognition [46], [47], [53], [54]. It has 50 classes, each with 4 sequences. Several breakdowns were used to evaluate the dataset [46]. Among them, we choose 3 representative settings:

50-Class: Original setting. 4-fold cross-validation is used.

9-Class: Sequences of different viewpoints are grouped as 9 classes to evaluate DT recognition under viewpoint changes. We use the same settings as in [46], [47]. We randomly choose half of the dataset for training and the other half for testing. The average recognition rate over 10 trials is reported.

Shift-invariant Recognition (SIR): Each sequence is spatially partitioned into non-overlapping left and right halves. "Shift-invariant recognition" [54] was implemented to compare sequences only between halves. This setting is very challenging. We report both rank-1 (R1) and rank-5 (R5) recognition rates.

We use LBP-TOP [5] as the feature descriptor, in which LBP features are extracted from three orthogonal planes, *i.e.* XY-plane, XT-plane and YT-plane, and concatenated to form the final feature vector. We try different ways to partition DTs. The best performance is achieved when we divide DTs along time axis into two patches for 50-class and SIR, and spatially divide sequences into 3×3 patches for 9-class. The feature dimensionality is 1536, 6912, 1536 for 50-class, 9-class and SIR, respectively. The results are shown in Table VI.

On the 50-class setting, the proposed CST-APCA achieves an error-free classification. On the 9-class setting, previous published lowest error rate is 2.5% achieved by DFS [46]. PCA-mMDC slightly improves the performance, and SR-PCA-mMDC boosts it further. On top of these, the proposed approach significantly reduces it to 1.0%. On the most challenging SIR setting, the published best result is R5 error rate of 26.2% achieved by DFS [46]. PCA-mMDC and SR-PCA-mMDC decrease the error rate compared with DFS, and the proposed approach further significantly reduces it to 2.2%. On the SIR-R1 setting, the proposed CST-APCA also significantly reduces the error rate compared with PCA-mMDC and SR-PCA-mMDC. In summary, on the UCLA dataset, the proposed CST-APCA achieves an error-free classification on the 50-class setting, and a better performance on the 9-class and SIR settings compared with the state-of-the-art approaches.

TABLE VI
COMPARISON WITH THE STATE OF THE ARTS ON THE UCLA DT DATASET FOR 50-CLASS, 9-CLASS AND SIR SETTINGS.

Method	Error Rate			
	50-Class	9-Class	SIR-R1	SIR-R5
Distributed spacetime orientation [54]	19.0%	-	57.7%	40.0%
DL-PEGASOS [47]	1.0%	4.4%	-	-
DFS [46]	0.0%	2.5%	-	26.2%
NNC-ED	10.5%	2.3%	22.0%	7.7%
NNC-Chi2D	10.0%	2.2%	14.0%	5.0%
PCA-mMDC	0.5%	2.1%	17.7%	4.0%
SR-PCA-mMDC [40]	0.5%	1.5%	16.5%	3.8%
PmSVM-Chi2 [39]	7.5%	2.9%	13.5%	-
PmSVM-HI [39]	7.5%	2.7%	13.5%	-
Proposed Approach	**0.0%**	**1.0%**	**12.7%**	**2.2%**

H. Dynamic Texture Recognition on the DynTex++ Dataset

The DynTex++ dataset proposed in [47] aims to provide a rich benchmark for DT recognition. It has 36 classes, each with 100 sequences of size $50 \times 50 \times 50$. LBP-TOP [5] is used as the feature descriptor, which has 768 dimensions. We use the same setup as in [46], [47]. For each class, 50 sequences are randomly chosen as the training set, and the other 50 are used in testing. The experiments are repeated 5 times and the average performance is reported in Table VII.

TABLE VII
COMPARISON WITH THE STATE OF THE ARTS ON THE DYNTEX++ DT DATASET.

Algorithms	Error Rate
DL-PEGASOS [47]	36.30%
DFS [46]	10.10%
NNC-ED	10.43%
NNC-Chi2D	5.92%
PCA-mMDC	5.48%
SR-PCA-mMDC [40]	4.60%
PmSVM-Chi2 [39]	11.79%
PmSVM-HI [39]	11.83%
Proposed Approach	**3.02%**

In literature, the lowest error rate reported on the DynTex++ dataset was 10.10% achieved by DFS [46]. When NNC-Chi2D is used with LBP-TOP, the error rate is significantly reduced. PCA-mMDC on LBP-TOP significantly outperforms NNC-ED, but only marginally outperforms NNC-Chi2D. SR-PCA-mMDC slightly reduces the error rate compared with PCA-mMDC. The proposed CST-APCA outperforms all compared approaches, and significantly reduces the error rate to 3.02%.

I. Computational Complexity Analysis

In Section II-D, we show that the proposed approach requires some additional memory and computational costs

compared with PCA-mMDC. In this section, we compare the time consumption of different approaches on the high-dim LBP feature of the LFW dataset, including time consumption of both training and testing. PmSVM is implemented in C++, and others are implemented in Matlab. The PC has Intel(R) Core(TM) i7-3930K CPU@ 3.2GHz and 16G RAM. The results are shown in Table VIII.

TABLE VIII
COMPARISON OF TIME CONSUMPTION OF DIFFERENT APPROACHES ON THE HIGH-DIM LBP FEATURES OF THE LFW DATABASE.

Algorithms	Time (s)
NNC-ED	19
NNC-Chi2D	21
PCA-mMDC	1295
SR-PCA-mMDC [40]	1618
PmSVM-Chi2 [39]	14217
PmSVM-HI [39]	14695
Proposed Approach	5428

As expected, NNC requires least time as no training required. SR-PCA-mMDC requires more time than PCA-mMDC due to square-rooting. The proposed approach requires more time than other subspace approaches, mainly due to the memory limitations so that we have to save data to harddisk and reload them later. Even so, the proposed approach is still much faster than PmSVMs. Note that the evaluation is 10-fold cross-validation on high-dim LBP features (127,440 dimensions) of the LFW dataset (13,233 images).

J. Summary and Discussion

From all 11 aforementioned image-classification results, it is very interesting and informative to see the following facts:

1) Without dimensionality reduction, Chi-squared distance consistently outperforms Euclidean distance.
2) The dimension-reduction approaches such as PCA-mMDC, SR-PCA-mMDC and the proposed CST-APCA consistently outperform NNC-ED.
3) Without the proposed transformation, dimensionality-reduction approaches do not always outperform NNC-Chi2D, e.g. PCA-mMDC performs poorer than NNC-Chi2D on the AR database.
4) SR-PCA-mMDC outperforms PCA-mMDC consistently.
5) The proposed approach consistently leads to the best performance in 10 out of 11 cases. Some performance gains are significant.
6) Besides LBP features, the proposed approach also outperforms others on HOG features of the LFW dataset.

The above facts clearly lead to the following analysis:

1) Infrequent patterns play an important role in LBP-histogram matching. Chi-squared distance better handles them than Euclidean distance.
2) Subspace approaches remove the unreliable dimensions of LBP features, and hence improve the performance.
3) LBP features deviate far from Gaussian distribution, which limits the performance gain brought by subspace approaches.
4) Square-rooting the features improves the fitness to Gaussian, and hence improves the performance.
5) The proposed Chi-squared transformation produces a feature that fits better to Gaussian distribution, and hence enhances the performance gain of subspace approaches.
6) The proposed transformation improves the performance not only on LBP features, but also on other histogram-like features such as HOG.

IV. CONCLUSION

Subspace approaches have been applied to the LBP histogram in order to remove unreliable dimensions or to derive a compact feature representation. We show that LBP-histogram feature deviates far from Gaussian distribution, which largely limits the performance gain brought by subspace approaches. This work proposes a truncated Gaussian model for the LBP feature, and a Chi-squared transformation to reduce the truncation effect. Both analysis and experiments show that the transformed feature fits better to Gaussian distribution. The proposed transformation naturally leads to an asymmetric two-class classification problem. We propose a CST-APCA approach to better remove the unreliable dimensions in the CST feature space. The proposed approach is tested on face recognition, protein cellular classification and dynamic texture classification. All experiments have demonstrated consistent performance gains brought by the proposed Chi-squared transformation, some of which are significant.

APPENDIX A
DERIVATION OF SKEWNESS AND KURTOSIS OF LBP FEATURE

Based on the results from [55], the r-th moment $m_r(\alpha,\beta)$ for the standard Gaussian distribution $\mathcal{N}(0,1)$ truncated in the range of (α,β) is calculated as:

$$m_{2k}(\alpha,\beta) = (2k-1)!!\left(1 - \sum_{i=1}^{k}\frac{1}{(2i-1)!!}\frac{[z^{2i-1}\phi(z)]_\beta^\alpha}{[\Phi(z)]_\beta^\alpha}\right) \quad \text{for } k = 1,2,\ldots,$$

$$m_{2k+1}(\alpha,\beta) = -\sum_{i=0}^{k}\frac{(2k)!!}{(2i)!!}\frac{[z^{2i}\phi(z)]_\beta^\alpha}{[\Phi(z)]_\beta^\alpha} \text{ for } k = 0,1,\ldots,$$

where $n!!$ is the double factorial, i.e. $n!! = n\times(n-2)!!$ if $n \geq 2$, and $n!! = 1$ if $n = -1, 0$ or 1. $[f(z)]_\beta^\alpha = f(\alpha) - f(\beta)$.

By plugging $\phi(\beta) = 0, \Phi(\beta) = 1$ into $m_{2k}(\alpha,\beta)$ and $m_{2k+1}(\alpha,\beta)$, the first four moments can be obtained. Plug these moments into $s_t = m_3(\alpha,\beta)/m_2^{3/2}(\alpha,\beta)$ and $k_t = m_4(\alpha,\beta)/m_2^2(\alpha,\beta)$, we can derive the skewness s_t in Eqn. (15) and the kurtosis k_t in Eqn. (16).

APPENDIX B
DERIVATION OF SKEWNESS AND KURTOSIS OF TRANSFORMED FEATURE

The p-th moment of z is calculated as: $\mathbb{E}\{z^p\} = \mathbb{E}\left\{(\frac{x-y}{\sqrt{x+y}})^p\right\}$. For every sample pair $\{x,y\}$, we have a corresponding sample pair $\{y,x\}$, i.e. $\tilde{z} = \frac{y-x}{\sqrt{x+y}}$, such that $z^p + \tilde{z}^p = 0$ for odd p. Thus, the odd moment of z is zero, and hence the skewness $s_z = 0$.

We now derive the kurtosis k_z. In Appendix A, $m_k(\alpha,\beta)$ is derived for $\mathcal{N}(0,1)$ truncated in (α,β). If the Gaussian distribution before truncation is $\mathcal{N}(\mu,\sigma)$, $m_k(\alpha,\beta) = \mathbb{E}\left\{(\frac{h-\mu}{\sigma})^k\right\}$. The raw moment $m_{k,u} = \mathbb{E}\{h^k\}$ can be calculated in an iterative manner by expanding $m_k(\alpha,\beta) = \mathbb{E}\left\{(\frac{h-\mu}{\sigma})^k\right\}$, i.e.

$$m_{k,u} = \sigma^k m_k(\alpha,\beta) + \sum_{i=1}^{k} \frac{k!(-1)^{i+1}\mu^i m_{k-i,u}}{i!(k-i)!}.$$

The second-order moment of z is $m_{2,z} = \mathbb{E}\left\{\frac{(x-y)^2}{x+y}\right\}$, where x, y are two independent random variables drawn from the truncated Gaussian distribution. We apply the Taylor expansion up to the second order for $\frac{1}{x+y}$ around $\mathbb{E}\{x+y\}$,

$$m_{2,z} \approx \mathbb{E}\left\{\frac{(x-y)^2}{\mathbb{E}\{x+y\}}\right\} - \mathbb{E}\left\{\frac{(x-y)^2(x+y-\mathbb{E}\{x+y\})}{\mathbb{E}^2\{x+y\}}\right\} + \mathbb{E}\left\{\frac{(x-y)^2(x+y-\mathbb{E}\{x+y\})^2}{\mathbb{E}^3\{x+y\}}\right\}.$$

Our analysis shows that $m_{2,z}$ is well approximated using the first term only. Including other two terms only slightly reduces the approximation error, but greatly increases the complexity. For simplicity, $m_{2,z}$ is approximated by $\mathbb{E}\left\{\frac{(x-y)^2}{\mathbb{E}\{x+y\}}\right\}$. Thus, $m_{2,z} \approx \frac{\mathbb{E}\{(x-y)^2\}}{\mathbb{E}\{x+y\}} = \frac{m_{2,u}-m_{1,u}^2}{m_{1,u}}$. Similarly, $m_{4,z} \approx \frac{\mathbb{E}\{(x-y)^4\}}{\mathbb{E}\{(x+y)^2\}}$. After some algebra, we have $\mathbb{E}\{(x-y)^4\} = 2m_{4,u} - 8m_{3,u}m_{1,u} + 6m_{2,u}^2$ and $\mathbb{E}\{(x+y)^2\} = 2m_{2,u} + 2m_{1,u}^2$. k_z is then approximated by

$$k_z \approx \frac{m_{4,u} - 4m_{3,u}m_{1,u} + 3m_{2,u}^2}{m_{2,u} + m_{1,u}^2} \times \frac{m_{1,u}^2}{(m_{2,u} - m_{1,u}^2)^2}. \quad (26)$$

k_z can be represented as a function of α by utilizing $\alpha = -\frac{\mu}{\sigma}$.

ACKNOWLEDGMENT

This research is supported in part by the Singapore National Research Foundation under its International Research Centre @ Singapore Funding Initiative and administered by the IDM Programme Office.

REFERENCES

[1] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, 2002.

[2] S. Liao, M. Law, and A. Chung, "Dominant local binary patterns for texture classification," *IEEE Transactions on Image Processing*, vol. 18, no. 5, pp. 1107–1118, 2009.

[3] Z. Guo, L. Zhang, and D. Zhang, "A completed modeling of local binary pattern operator for texture classification," *IEEE Transactions on Image Processing*, vol. 19, no. 6, pp. 1657–1663, 2010.

[4] Z. Li, G. Liu, Y. Yang, and J. You, "Scale-and rotation-invariant local binary pattern using scale-adaptive texton and subuniform-based circular shift," *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 2130–2140, 2012.

[5] G. Zhao and M. Pietikainen, "Dynamic texture recognition using local binary patterns with an application to facial expressions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, pp. 915–928, 2007.

[6] G. Zhao, T. Ahonen, J. Matas, and M. Pietikainen, "Rotation-invariant image and video description with local binary pattern features," *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 1465–1477, 2012.

[7] J. Ren, X. Jiang, and J. Yuan, "Dynamic texture recognition using enhanced LBP features," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 2400–2404.

[8] T. Ahonen, A. Hadid, and M. Pietikainen, "Face description with local binary patterns: Application to face recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 12, pp. 2037–2041, 2006.

[9] X. Tan and B. Triggs, "Enhanced local texture feature sets for face recognition under difficult lighting conditions," *IEEE Transactions on Image Processing*, vol. 19, no. 6, pp. 1635–1650, 2010.

[10] J. Ren, X. Jiang, and J. Yuan, "Noise-resistant local binary pattern with an embedded error-correction mechanism," *IEEE Transactions on Image Processing*, vol. 22, no. 10, pp. 4049–4060, 2013.

[11] ——, "Relaxed local ternary pattern for face recognition," in *IEEE International Conference on Image Processing*, 2013, pp. 3680–3684.

[12] J. Wu and J. Rehg, "CENTRIST: A visual descriptor for scene categorization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1489–1501, 2011.

[13] J. Ren, X. Jiang, and J. Yuan, "Learning binarized pixel-difference pattern for scene recognition," in *IEEE International Conference on Image Processing*, 2013, pp. 2494–2498.

[14] Y. Xiao, J. Wu, and J. Yuan, "mCENTRIST: A multi-channel feature generation mechanism for scene categorization," *IEEE Transactions on Image Processing*, vol. 23, no. 2, pp. 823–836, 2014.

[15] M. Heikkilä, M. Pietikäinen, and C. Schmid, "Description of interest regions with local binary patterns," *Pattern recognition*, vol. 42, no. 3, pp. 425–436, 2009.

[16] S. Liao, G. Zhao, V. Kellokumpu, M. Pietikainen, and S. Li, "Modeling pixel process with scale invariant local patterns for background subtraction in complex scenes," in *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2010, pp. 1301–1306.

[17] A. Satpathy, X. Jiang, and H. Eng, "LBP based edge-texture features for object recognition," *IEEE Transactions on Image Processing*, vol. 23, no. 5, pp. 1953–1964, 2014.

[18] J. Ren, X. Jiang, J. Yuan, and G. Wang, "Optimizing LBP structure for visual recognition using binary quadratic programming," *IEEE Signal Processing letters*, vol. 21, no. 11, pp. 1346 – 1350, 2014.

[19] J. Ren, X. Jiang, and J. Yuan, "Learning LBP structure by maximizing the conditional mutual information," *Pattern Recognition*, 2015. [Online]. Available: http://dx.doi.org/10.1016/j.patcog.2015.02.001

[20] J. Yuan, M. Yang, and Y. Wu, "Mining discriminative co-occurrence patterns for visual recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 2777–2784.

[21] Z. Cao, Q. Yin, X. Tang, and J. Sun, "Face recognition with learning-based descriptor," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2010, pp. 2707–2714.

[22] S. Yan, H. Wang, X. Tang, and T. Huang, "Exploring feature descritors for face recognition," in *Proc. IEEE International. Conference on Acoustics, Speech and Signal Processing*, vol. 1, 2007, pp. 629–632.

[23] Y. Fang and Z. Wang, "Improving LBP features for gender classification," in *Proc. International. Conference on Wavelet Analysis and Pattern Recognition*, 2008, pp. 373–377.

[24] W. Wang, F. Chang, J. Zhao, and Z. Chen, "Automatic facial expression recognition using local binary pattern," in *World Congress on Intelligent Control and Automation (WCICA)*. IEEE, 2010, pp. 6375–6378.

[25] D. Chen, X. Cao, F. Wen, and J. Sun, "Blessing of dimensionality: High-dimensional feature and its efficient compression for face verification," in *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2013, pp. 3025–3032.

[26] C. Zeng and H. Ma, "Robust head-shoulder detection by pca-based multilevel HOG-LBP detector for people counting," in *International Conference on Pattern Recognition*. IEEE, 2010, pp. 2069–2072.

[27] X. Jiang, B. Mandal, and A. Kot, "Eigenfeature regularization and extraction in face recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 3, pp. 383–394, 2008.

[28] X. Jiang, "Asymmetric principal component and discriminant analyses for pattern classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 5, pp. 931–937, 2009.

[29] ——, "Linear subspace learning-based dimensionality reduction," *IEEE Signal Processing Magazine*, vol. 28, no. 2, pp. 16–26, 2011.

[30] G. B. Huang, M. Mattar, T. Berg, E. Learned-Miller *et al.*, "Labeled faces in the wild: A database forstudying face recognition in unconstrained environments," in *Workshop on Faces in Real-Life Images: Detection, Alignment, and Recognition*, 2008.

[31] P. Belhumeur, J. Hespanha, and D. Kriegman, "Eigenfaces vs. fisherfaces: Recognition using class specific linear projection," *IEEE Trans-*

actions on Pattern Analysis and Machine Intelligence, vol. 19, no. 7, pp. 711–720, 1997.
[32] B. Moghaddam, "Principal manifolds and probabilistic subspaces for visual recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 6, pp. 780–788, 2002.
[33] X. Jiang, B. Mandal, and A. Kot, "Enhanced maximum likelihood face recognition," *Electronics Letters*, vol. 42, no. 19, pp. 1089–1090, 2006.
[34] C. M. Jarque and A. K. Bera, "A test for normality of observations and regression residuals," *International Statistical Review*, pp. 163–172, 1987.
[35] R. B. D'agostino, A. Belanger, and R. B. D'Agostino Jr, "A suggestion for using powerful and informative tests of normality," *The American Statistician*, vol. 44, no. 4, pp. 316–321, 1990.
[36] H. C. Thode, *Testing for normality*. CRC Press, 2002.
[37] G. W. Corder and D. I. Foreman, *Nonparametric statistics for non-statisticians: a step-by-step approach*. John Wiley & Sons, 2009.
[38] P. E. Greenwood, *A guide to chi-squared testing*. John Wiley & Sons, 1996, vol. 280.
[39] J. Wu, "Power mean SVM for large scale visual classification," in *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 2344–2351.
[40] J. Wu, Y. Zhang, and W. Lin, "Towards good practices for action video encoding," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2014, pp. 2577–2584.
[41] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1. IEEE, 2005, pp. 886–893.
[42] A. Martinez and R. Benavente, "The AR face database," *CVC Tech. Report #24*, 1998.
[43] J. Ren, X. Jiang, and J. Yuan, "A complete and fully automated face verification system on mobile devices," *Pattern Recognition*, vol. 46, no. 1, pp. 45 – 56, 2013.
[44] M. V. Boland and R. F. Murphy, "A neural network classifier capable of recognizing the patterns of all major subcellular structures in fluorescence microscope images of hela cells," *Bioinformatics*, vol. 17, no. 12, pp. 1213–1223, 2001.
[45] G. Doretto, A. Chiuso, Y. Wu, and S. Soatto, "Dynamic textures," *International Journal of Computer Vision*, vol. 51, no. 2, pp. 91–109, 2003.
[46] Y. Xu, Y. Quan, H. Ling, and H. Ji, "Dynamic texture classification using dynamic fractal analysis," in *International Conference on Computer Vision*, nov. 2011, pp. 1219 –1226.
[47] B. Ghanem and N. Ahuja, "Maximum margin distance learning for dynamic texture recognition," in *European Conference on Computer Vision*, 2010, pp. 223–236.
[48] H. Li, G. Hua, Z. Lin, J. Brandt, and J. Yang, "Probabilistic elastic matching for pose variant face verification," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3499–3506.
[49] K. Simonyan, O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Fisher vector faces in the wild," in *British Machine Vision Conference*, vol. 1, no. 2, 2013, p. 7.
[50] Y. Guo, G. Zhao, and M. Pietikinen, "Discriminative features for texture description," *Pattern Recognition*, vol. 45, no. 10, pp. 3834 – 3843, 2012.
[51] P. Saisan, G. Doretto, Y. Wu, and S. Soatto, "Dynamic texture recognition," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, 2001, pp. 58–63.
[52] A. Chan and N. Vasconcelos, "Probabilistic kernels for the classification of auto-regressive visual processes," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2005, pp. 846–851.
[53] A. Ravichandran, R. Chaudhry, and R. Vidal, "View-invariant dynamic texture recognition using a bag of dynamical systems," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 1651–1657.
[54] K. G. Derpanis and R. P. Wildes, "Spacetime texture representation and recognition based on a spatiotemporal orientation analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 6, pp. 1193–1205, 2012.
[55] C. Flecher, D. Allard, and P. Naveau, "Truncated skew-normal distributions: moments, estimation by weighted moments and application to climatic data," *Metron*, vol. 68, no. 3, pp. 331–345, 2010.

Jianfeng Ren received the B.Eng. from National University of Singapore, major in Communication and M.Sc. from Nanyang Technological University (NTU), major in Signal Processing in 2001 and 2009, respectively. He received Professional Engineers Board Gold Medal for his outstanding academic results among all EEE M.Sc. graduates in 2009. From 2003 to 2007, he was in industry sections. In Dec 2007, he joined NTU as a project officer, responsible for the development of the face verification system on mobile devices. In Sept 2011, he joined BeingThere Centre, Institute of Media Innovation, NTU as a research associate. Currently he is pursuing Ph.D. in Electrical and Electronic engineering at NTU. His research interests include face recognition, general pattern recognition and computer vision.

Xudong Jiang (M'02-SM'06) received the B.Eng. and M.Eng. degrees from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 1983 and 1986, respectively, and the Ph.D. degree from Helmut Schmidt University, Hamburg, Germany, in 1997, all in electrical engineering. From 1986 to 1993, he was a Lecturer with UESTC, where he received two Science & Technology Awards from the Ministry for Electronic Industry of China. From 1993 to 1997, he was a Scientific Assistant with Helmut Schmidt University. From 1998 to 2004, Dr Jiang was with the Institute for Infocomm Research, A*Star, Singapore, as a Lead Scientist and the Head of the Biometrics Laboratory, where he developed a system that achieved the most efficiency and the second most accuracy at the International Fingerprint Verification Competition (FVC'00). He joined Nanyang Technological University (NTU), Singapore, as a faculty member in 2004 and served as the Director of the Centre for Information Security from 2005 to 2011. Currently, he is a tenured Associate Professor in School of EEE, NTU.

Dr Jiang has published over hundred papers, where 20 papers in IEEE Journals: TIP (6), TPAMI (5), TSP (3), SPL (2), SPM, TIFS, TCSVT and TCS-II. He is the holder of seven patents. His research interest includes signal/image processing, pattern recognition, computer vision, machine learning, and biometrics.

Junsong Yuan (M'08-SM'14) is currently a Nanyang Assistant Professor and program director of video analytics at School of EEE, Nanyang Technological University, Singapore. He received Ph.D. from Northwestern University, USA, and M.Eng. from National University of Singapore. Before that, he graduated from Special Class for the Gifted Young of Huazhong University of Science and Technology, China. His research interests include computer vision, video analytics, large-scale visual search and mining, human computer interaction etc. He has co-authored over 120 journal and conference papers, and filed three US patents and two provisional US patents.

He co-chairs workshops at SIGGRAPH Asia'14, IEEE Conf. on Computer Vision and Pattern Recognition (CVPR'12'13'15), IEEE Conf. on Computer Vision (ICCV'13), and serves as Program Co-Chair of IEEE Visual Communications and Image Processing (VCIP'15), Organizing Co-Chair of Asian Conf. on Computer Vision (ACCV'14), Area chair of IEEE Winter Conf. on Computer Vision (WACV'14), IEEE Conf. on Multimedia Expo (ICME'14'15), and Asian Conf. on Computer Vision (ACCV'14). He also serves as guest editor for International Journal of Computer Vision (IJCV), associate editor for The Visual Computer journal (TVC) and Journal of Multimedia. He received Nanyang Assistant Professorship from Nanyang Technological University, Outstanding EECS Ph.D. Thesis award from Northwestern University, Best Doctoral Spotlight Award from CVPR'09, and National Outstanding Student from Ministry of Education, P.R.China. He gives tutorials at ACCV'14, ICIP'13, FG'13, ICME'12, SIGGRAPH VRCAI'12, and PCM'12.