PAPER

Roughness Classification with Aggregated Discrete Fourier Transform

Chao LIANG[†,††,†††a)], **Wenming YANG**[†,††,†††b)], ***Nonmembers***, **Fei ZHOU**[†,††,†††c)], ***Member***, ***and*** **Qingmin LIAO**[†,††,†††d)], ***Nonmember***

SUMMARY In this paper, we propose a texture descriptor based on amplitude distribution and phase distribution of the discrete Fourier transform (DFT) of an image. One dimensional DFT is applied to all the rows and columns of an image. Histograms of the amplitudes and gradients of the phases between adjacent rows/columns are computed as the feature descriptor, which is called aggregated DFT (ADFT). ADFT can be easily combined with completed local binary pattern (CLBP). The combined feature captures both global and local information of the texture. ADFT is designed for isotropic textures and demonstrated to be effective for roughness classification of castings. Experimental results show that the amplitude part of ADFT is also discriminative in describing anisotropic textures and it can be used as a complementary descriptor of local texture descriptors such as CLBP.

key words: *aggregated DFT (ADFT), completed local binary pattern (CLBP), roughness texture classification*

1. Introduction

Surface roughness is a key feature of materials because it relates to a material's abrasion resistance, fatigue strength, resistance to corrosion, etc. Real-time roughness evaluation of castings is key to check whether the product is acceptable. There are two ways to address the roughness evaluation problem. One is to measure the roughness value accurately, the other is to classify a specimen into a predefined roughness class [1], [2], which represents a range of roughness values. The most common parameter to describe roughness values is average roughnesses (Ra), which is the arithmetic average of absolute values on a straight line.

Traditional instruments for roughness evaluation are contact profilometers, which use a stylus tracing a straight line to obtain the profile of the specimen. Contact profilometers are accurate and the measurement precision can reach up to 10 nanometres. But this kind of methods has three shortcomings. 1) The specimen and the stylus will abrade each other. Because the stylus needs to keep contact with the specimen while sampling, the specimen will be slightly scratched and the stylus needs to be replaced periodically. 2) Contact profilometers are expensive. 3) Contact methods are slow. Because the stylus needs to move from point to point, it takes about 2 seconds for each measurement, making contact methods improper for on-line measurement.

Computer vision based techniques do not need to contact the specimen. Structured light technology is one of them and the height resolution of structured light technology is up to about 0.2 μm. The more precise the structured light is, the more expensive the device is, and the narrower the measurement range is. Meanwhile, most structured light devices are slow, taking about 2 seconds to scan over an area, making it improper for on-line measurement.

Texture classification is another computer vision based technique to address the roughness evaluation problem. Manual comparison against a surface roughness comparator is a standard way to estimate the roughness level of the specimen. A roughness comparator consists of a range of specimens with pre-defined roughness levels. The test specimen is classified into one of the predefined classes, which is actually a classification problem. There exists roughness evaluation methods [3], [4] that use the anisotropic characteristics, i.e. the scratched stripes, of polished surface. Different from polished surface, castings surface is isotropic and irregular, making it more difficult to evaluate their roughness. In this paper, we focus on roughness evaluation of castings using texture classification technology for real-time applications.

Texture classification is a hot topic in the last twenty years. Model based methods, filtering based methods, transform based methods and statistical methods are commonly used for texture feature extraction.

Model based approaches assume that different textures are generated through models with different parameters. Autoregressive model [5], Markov random field [6], [7] are commonly used models. However, generative models are more suitable for representation, rather than classification. And it will be too complicated for the model to represent textures accurately.

Filter and transform based methods are also called signal processing methods. Filter based methods use filters, such as Gabor filter [8], Gaussian and Gaussian derivative filters [9], to convolve with the image. Responses of the filters are collected and a feature selection step usually follows to reduce the dimension of the features. Transforms such as discrete Fourier transform (DFT) [10]–[13], dis-

Manuscript received March 10, 2014.
Manuscript revised May 29, 2014.

[†]The authors are with the Department of Electronic Engineering/Graduate School at Shenzhen, Tsinghua University, Beijing, China.
[††]The authors are with the Shenzhen Key Laboratory of Information Science and Technology, Guangdong, China.
[†††]The authors are with the Visual Information Processing Lab, Tsinghua-PolyU Biometrics Joint Lab, Guangdong, China.
a) E-mail: chao.liang.thu@gmail.com
b) E-mail: yangelwm@163.com
c) E-mail: flying.zhou@163.com
d) E-mail: liaoqm@tsinghua.edu.cn

DOI: 10.1587/transinf.2014EDP7082

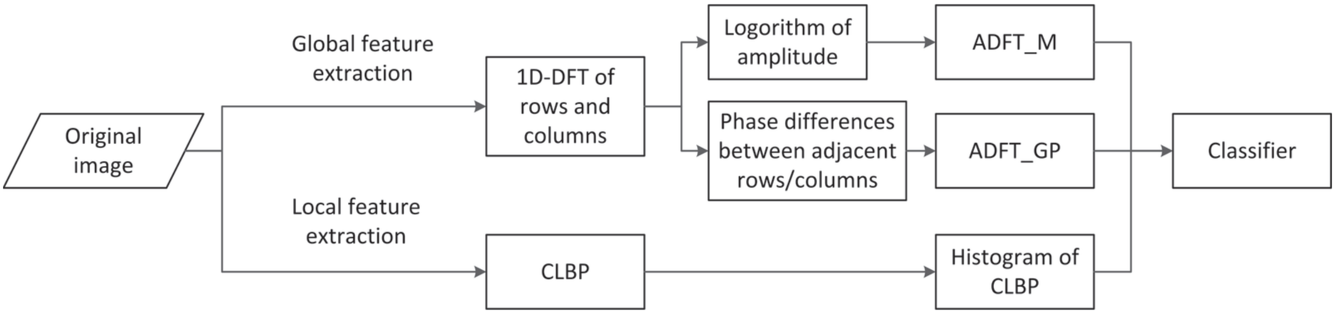


Fig. 1 Flowchart for joint ADFT and CLBP classification.

crete wavelet transform (DWT) [14], [15] transform the image into another space for feature extraction. Most of the DFT-based methods perform transform in local areas, for instance, square areas with $3 * 3$ pixels.

Statistical methods use first-order, second-order, or even higher-order statistics of an image as the discriminative information. Gray level run lengths [16] and Co-occurrence matrix [17] are representative ones in the early years. E. Gadelmawla [2] has applied co-occurrence matrix to characterize roughness of polished surfaces. Local binary pattern (LBP) [18] and its modified version [19] use statistics of differences between adjacent pixels as features. They are proven to be fast and with high recognition rate for many widely used data sets. After that, tens of variations of LBP have been proposed [20]. Local phase quantization (LPQ) [21] is another local texture descriptor, which uses phase information of local two-dimensional DFT and proved to be blur insensitive.

To evaluate the roughness in real-time, the speed of feature extraction and classification is a key point. Because the utilization of fast Fourier transform algorithm, DFT can be applied in real-time. D. Tsai et al. selected energy rings of power spectrum for roughness classification of castings [1], which evaluate the roughness of a specimen using two-dimensional DFT. On the other hand, roughness value can be calculated on a single line of the specimen, which results in the one-dimensional DFT approach that we propose in this paper.

We propose aggregated DFT (ADFT) in this paper, where distributions of one-dimensional DFT coefficients are utilized. ADFT is a global descriptor, i.e. ADFT covers a large range of the image to extract the distributions. We combine ADFT with another state-of-the-art texture descriptor completed local binary pattern (CLBP) [22]. CLBP mainly captures local information of the texture. The two complementary features describe the textures in a both global and local ways and can be computed in real-time. The flowchart for joint ADFT and CLBP classification algorithm is illustrated in Fig. 1.

The rest of this paper is organized as follows. In Sect. 2, we introduce the motivations behind ADFT. In Sect. 3, we introduce the proposed ADFT texture descriptor and give a brief review of CLBP. Then we discuss the differences between distinct classifiers and metrics in Sect. 4. Data acquisition equipment, experiments and discussions follow in Sect. 5. Finally, Sect. 6 concludes this paper.

2. Motivation

As mentioned before, a standard way of surface roughness evaluation is comparing the specimen with a roughness comparator manually. It is reasonable to utilize texture classification technology to liberate the workforce and obtain a more objective result.

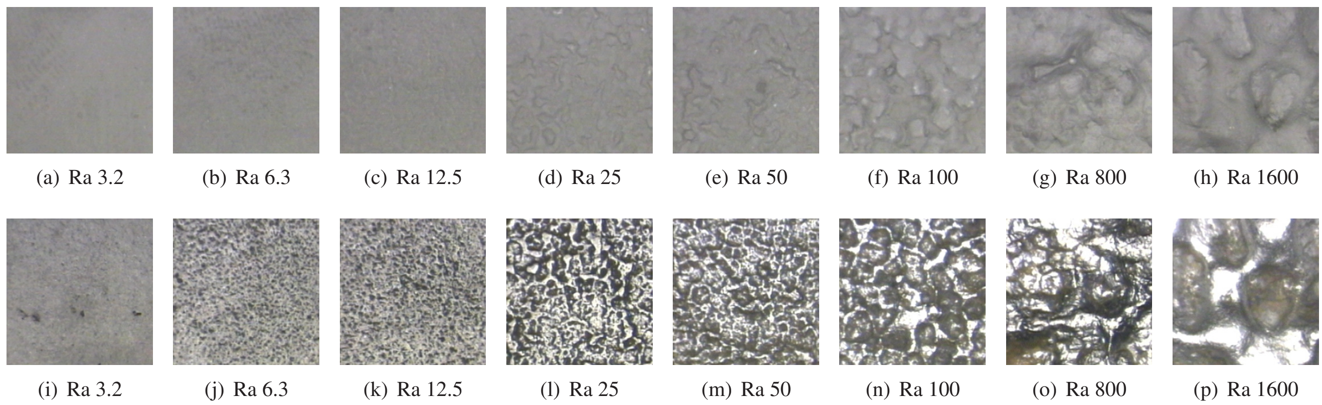
Textures of casting images are affected by many factors, such as roughness of the castings, the material of the castings and illumination condition, etc.

The illumination condition of production environment is constant. Therefore, we do not take illumination variance into consideration. The casting image database used in this paper are acquired in constant illumination condition to simulate the product environment.

One of the difficulties is that appearances of casting specimens with different materials are very different, as shown in Fig. 2. Specimens with diffuse reflection have a narrow range of gray value distribution, while gray values of specimens with specular reflection distribute in a wide range. As illustrated in Fig. 3, gray values of the one in Fig. 2 (f) range from 120 to 180 while gray values of the one in Fig. 2 (n) range from 70 to 255 and there are many pixels with the greatest value 255. The large intra-class variation makes it difficult for classification. One solution is to treat them as two different classes and the other is to use robust texture descriptors.

In this paper, we propose a robust texture descriptor based on two observations. First, textures of castings are isotropic. This is intuitive since that castings are solidified from liquid steel, and liquid is isotropic. For this reason, the descriptor does not need to be rotation invariant. Second, contact profilometers only sample on a straight line to evaluate the roughness of a specimen, which means a single row or column of a casting image is sufficient to evaluate the roughness of the specimen.

Based on the above observations, we propose ADFT to extract the global information of casting textures. The ADFT descriptor consists of two parts, i.e. the amplitude part (ADFT_M) and the phase part (ADFT_GP). ADFT explores features of textures in frequency domain. ADFT_M



(a) Ra 3.2 (b) Ra 6.3 (c) Ra 12.5 (d) Ra 25 (e) Ra 50 (f) Ra 100 (g) Ra 800 (h) Ra 1600

(i) Ra 3.2 (j) Ra 6.3 (k) Ra 12.5 (l) Ra 25 (m) Ra 50 (n) Ra 100 (o) Ra 800 (p) Ra 1600

Fig. 2 Casting textures with diffuse reflection (first row) and specular reflection (second row).

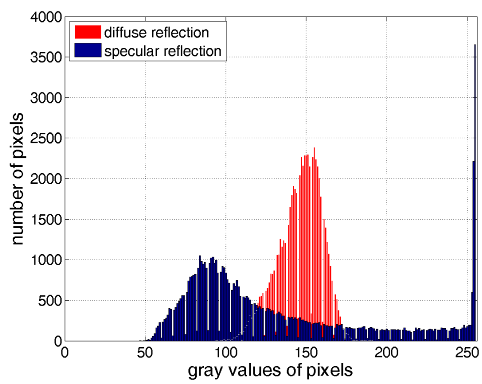


Fig. 3 Histograms of Ra 800 with different materials.

uses the amplitude distribution of DFT of each rows and columns to describe the texture and ADFT_GP uses gradients of phases between adjacent rows/columns to describe the texture.

In addition, CLBP is proved to be a robust texture descriptor that captures local discriminative information. Combining ADFT and CLBP, the descriptor can capture both global and local information, and a better classification result is expected.

3. Roughness Feature Selection

3.1 Global Feature Extraction Using ADFT

ADFT is based on one-dimensional DFT. The DFT of a single row / column with N pixels is given by

$$F(u) = \mathrm{DFT}[f(x)] = \frac{1}{N}\sum_{x=0}^{N-1} f(x)\exp\left(-j\frac{2\pi}{N}ux\right) \qquad (1)$$

for $u = 0, 1, \cdots, N$, where $f(\cdot)$ is the gray value a pixel, and x is the position of the pixel.

Because the gray levels of images are all real numbers and because of the conjugation property of DFT, we obtain

$$|F(u)| = |F(N-u)| \qquad (2)$$

$$\arg[F(u)] = -\arg[F(N-u)] \qquad (3)$$

where $|\cdot|$ and $\arg[\cdot]$ denote the amplitude and the angle of a complex number, respectively.

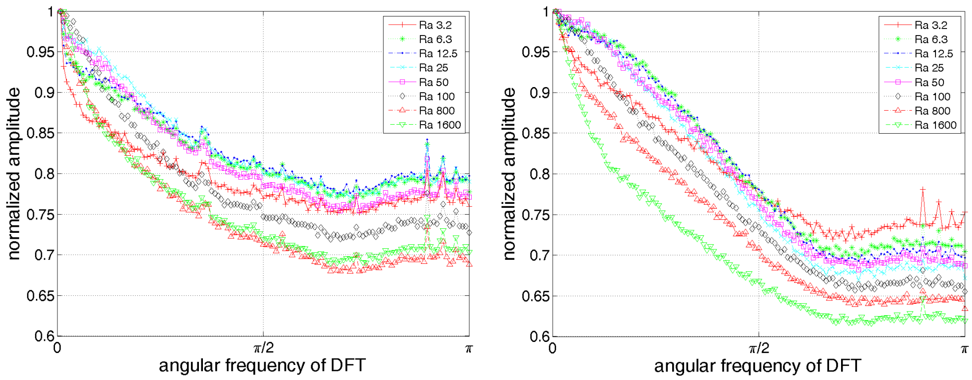
Equation (2) and (3) imply that half of the coefficients of the amplitude and the angle are enough. Because high frequency components of DFT are more sensitive to noise and illumination variation, we only reserve the low frequency components in the proposed descriptor. The zero-frequency component is also discarded because the mean intensity of a texture image mainly change with the reflection characteristics of materials rather than the roughness of a specimen.

Because casting images suffer from heavy noise, it is difficult to classify roughness textures using a single row / column. Therefore, we aggregate the DFT coefficients of each row and column to form a histogram. The more DFT coefficients we aggregate, the higher the signal-to-noise ratio we gain. ADFT is applied to square images with $N \times N$ pixels, so that DFT of rows and DFT of columns are with the same frequency resolution and the same number of coefficients. Consequently, DFT coefficients of rows and columns can be combined. The ADFT descriptor consists of two parts, i.e., the amplitude part (ADFT_SM and ADFT_M) and the differences of phases part (ADFT_GP). We refer ADFT to the concatenation of ADFT_M and ADFT_GP in the following of this paper.

We propose two different schemes for the amplitude part. The first one is simply summating the corresponding amplitudes of each DFT in each frequency (ADFT_SM), as shown in Fig. 4 and Eq. (4). The second one is computing the amplitude distributions in each frequency in a discrete space (ADFT_M), as illustrated in Fig. 5 and Eq. (5).

$$ADFT\_SM(u) = \sum_{r=1}^{N}|F_r(u)| + \sum_{c=1}^{N}|F_c(u)| \qquad (4)$$

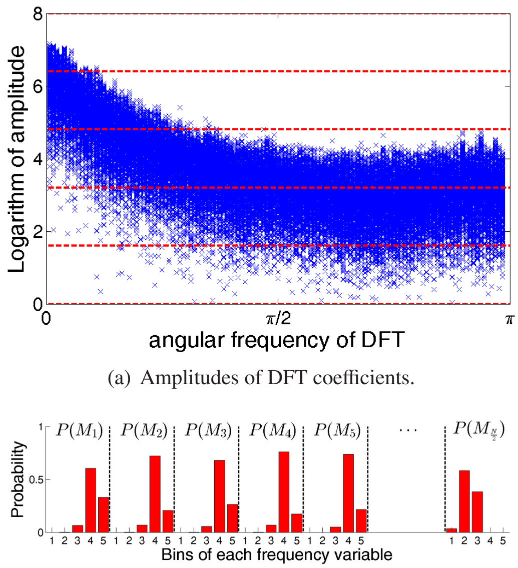
$$\begin{aligned} ADFT\_M(u,i) = &\sum_{r=1}^{N}\delta\left(Q_{i-1} < |F_r(u)| \le Q_i\right) \\ &+ \sum_{c=1}^{N}\delta\left(Q_{i-1} < |F_c(u)| \le Q_i\right) \end{aligned} \qquad (5)$$



(a) Diffuse reflection.

(b) Specular reflection.

Fig. 4 Summations of 1D-DFT amplitude of casting textures with specular reflection and diffuse reflection. The greatest amplitudes of each DFT coefficients are normalized to one for the sake of illustration.



(a) Amplitudes of DFT coefficients.

(b) ADFT_M feature of (a).

Fig. 5 ADFT_M feature extraction. If the image size is 256*256 pixels, 512 instances of amplitudes are quantized into several bins (5 here) as the red dashed line in (a) illustrates. Distributions of the amplitudes at each frequency are estimated by counting the number of data in each bin. ADFT_M is the concatenation of these distributions as (b) shows. In (b), M_u is equivalent to $|F(u)|$ for $u = 1, 2, \cdots, \frac{N}{2}$.

where r and c are the row and column indexes. $|F_r(\cdot)|$ and $|F_c(\cdot)|$ are the amplitudes of DFT for the corresponding row / column. $\delta(\cdot)$ is the indicator function. Q_i for $i = 0, 1, \cdots, N_Q$ are quantization levels, and N_Q is the number of quantization levels.

Details of the ADFT_M feature extraction algorithm are listed in Algorithm 1. The logarithmic function in the second step makes amplitudes compact at large values, and the resulting values distribute in a small range. Therefore, in the third step, a uniform quantizer performs well. After aggregation step, the resulting ADFT_M descriptor is shown in Fig. 5 (b), which shows that ADFT is the concatenation of amplitude distributions. As we mentioned before, high frequency components are discarded in practice. The algorithm for ADFT_SM feature extraction is the same as Algorithm 1 except for that we compute the summations of amplitudes in the third step.

Algorithm 1 ADFT_M feature extraction.

Input:
Casting image I with size N*N
Output:
ADFT feature of the image
1: **Perform DFT to each row and column**
2: **Compute the logarithm of the DFT amplitudes**
Amplitudes of DFT coefficients are transformed into their logarithmic form using the formula $\log(|F(u)| + 1)$.
3: **Quantization**
Quantize the amplitudes into N_Q levels.
4: **Aggregate DFT amplitudes and compute their distribution**
On each frequency of DFT, compute the discrete distribution of its amplitudes using the $2N$ data points (N from row DFTs and N from column DFTs).

ADFT_SM and ADFT_M are similar descriptors. If the signal-to-noise ratio of the texture image is low, ADFT_SM is more robust because the summation of multiple amplitudes could improve the signal-to-noise ratio. On the other hand, ADFT_M maintains more discriminative information than ADFT_SM, therefore, ADFT_M would outperforms ADFT_SM in most cases.

The phase part of ADFT is derived from the isotropic characteristic of castings. We assume that images of castings consist of a series of sinusoidal signals, as the DFT does. Under the assumption that the roughness of the specimen is all the same on each row and column, the corresponding amplitudes of each frequency on each row and column are the same while the phase may differ from each other. Taking the sinusoidal signal with angular frequency w for example, these sinusoidal components of the first row and the first column are given by

$$I_w(x,0) = A \times \sin(wx + \phi_0) \tag{6}$$
$$I_w(0,y) = A \times \sin(wy + \phi_0) \tag{7}$$

where $I_w(\cdot,\cdot)$ is the sinusoidal component with angular frequency w of a pixel, x and y are coordinates, and A is the amplitude of the sinusoidal signal.

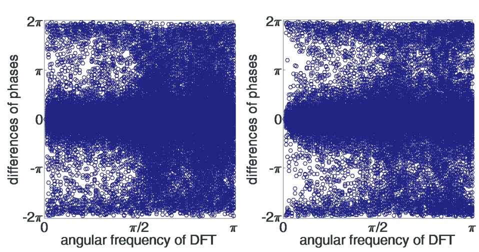
Substituting 1 for y in Eq. (7), we obtain $I(0,1) = A \times \sin(w + \phi_0)$, and because we assume the amplitude of the sinusoidal signal with angular frequency w of the second row is the same as the first row's, we obtain

$$I_w(x,1) = A \times \sin(wx + w + \phi_0) \tag{8}$$

Equation (6) and (8) indicate that the only difference between the first row and the second row on the sinusoidal component with angular frequency w is the phase difference w. The same result holds for other adjacent rows. Because actual images do not fully satisfy our assumption, sinusoidal components with small amplitudes are more probably disobeying the above inference. In other words, if the amplitude of a sinusoidal component is small, the distribution of phase differences of this component is nearly uniform as noise is the main contributor. Because specimens with different roughness values are composed of sinusoidal signals with different distributions of amplitudes, their distributions of phase differences should differ from each other. Therefore, we can use the phase differences as a texture descriptor.

We call this feature ADFT_GP because it is the gradient of phases on the direction perpendicular to the direction in which DFT is performed. If the frequency is a dominant one of the image, the difference of phases is stable, otherwise the distribution will be more even. As the illustrated in Fig. 6, low frequencies are the majority for castings with large Ra value, and the gradients of phases are more stable. We extract ADFT_GP by the same way as ADFT , except for that we quantize the gradient of phases between adjacent rows/columns here. Different from Fig. 6, we constraint the gradient to a range from 0 to π, therefore, a smaller number of parameters are needed to estimate and the model we obtain is less likely to overfit.

ADFT is robust to illumination change because only the low frequency parts of DFT are used. At the same time, ADFT is fast. The feature extraction time of ADFT is $O(N_I \log N_I)$, where N_I is the number of pixels in an image. These two features of ADFT make it proper for online detection.



(a) Ra 6.3 (b) Ra 800

Fig. 6 Differences of phases between adjacent rows/columns.

3.2 Local Feature Extraction Using CLBP

3.2.1 Review of LBP

Local binary pattern (LBP) [19] is a binary code of the gray value differences. For each pixel in the image, the binary code is given by the differences of the gray values between the neighbors with the central pixel. If the neighbor pixel is greater than the central pixel, the bit is assigned to 1, and 0 otherwise. Neighbors with gray values The P neighbors are uniformly distributed on the circle of radius R, and this kind of LBP is called $\mathrm{LBP}_{P,R}$. For instance, the pattern in Fig. 7 (c) is $\mathrm{LBP}_{8,1}$ code of Fig. 7 (a). Gray values of neighbors that are not on the grid of the image are interpolated using bilinear interpolation. It is worth noting that there are certain independent number of neighbors given the radius of the circle, e.g. eight independent neighbors when the radius is one, so that the parameter P is limited by the parameter R.

To make LBP compact and robust, two improvements, i.e., rotation invariant LBP and uniform patterns of LBP, are proposed in [19]. The rotation invariant LBP, denoted by $\mathrm{LBP}^{\mathrm{ri}}$, transform a binary pattern to its minimum value using circular shift. For instance, 10000000 and 00001000 are all transformed to 00000001. Uniform patterns are binary patterns with two or less transitions between 0 and 1. Non-uniform patterns are grouped to the same bin because their probability of occurrence are much less than uniform patterns. Combining the rotation scheme with the uniform pattern scheme, $\mathrm{LBP}^{\mathrm{riu2}}_{P,R}$ is obtained. There are $P+2$ bins for $\mathrm{LBP}^{\mathrm{riu2}}_{P,R}$ while for $\mathrm{LBP}_{P,R}$, the number of bins is 2^P.

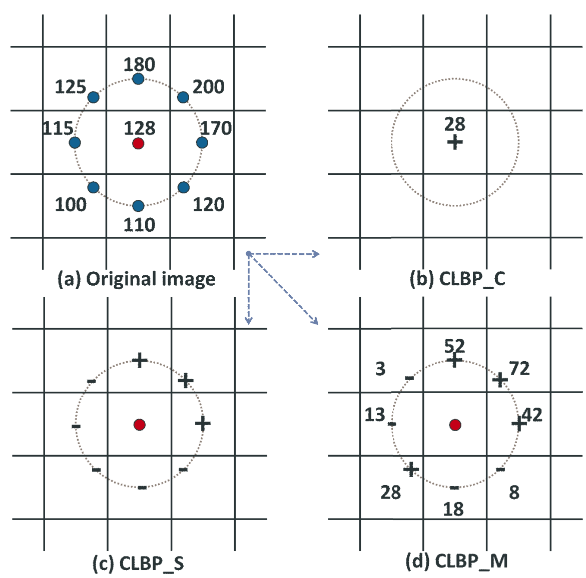


Fig. 7 $\mathrm{CLBP}_{8,1}$, the + and − sign represent 1 and 0, respectively. (a) the original image. (b) CLBP_C, we assume m_I is 100 here. (c) CLBP_S, it is the same as LBP. (d) CLBP_M, we assume m_M is 20 here.

3.2.2 Review of CLBP

CLBP [22] is an extension of LBP. As shown in Fig. 7 (c), CLBP_S (LBP) only utilizes the sign of the difference between the central pixel and its surroundings as the discriminative information. Z. Guo found that while the sign is powerful for discrimination, the magnitude of the difference is still discriminative. CLBP_M encodes the magnitude part that LBP discards in a consistent way as LBP. Magnitudes less than the mean magnitude of the image (m_M) are encoded as 0 and the left encoded as 1. CLBP_C is a one-bit pattern that thresholds the central pixel using the mean gray value of the image (m_I). Gray values less than m_I are encoded as 0 and the left encoded as 1. CLBP is the concatenation of CLBP_C, CLBP_S and CLBP_M. For instance, the pattern in Fig. 7 (a) is 1, 11100000, 11100100. The rotation invariant scheme and uniform pattern scheme still applies to CLBP_S and CLBP_M. As a result, $\mathrm{CLBP}_{P,R}^{\mathrm{riu2}}$ contains $2(P+2)(P+2)$ bins.

3.2.3 Roughness Feature with CLBP

We consider CLBP to be the complement of ADFT based on three considerations: First, as shown in Fig. 2, casting textures with small Ra value change rapidly while textures with large Ra value change slowly. As a consequence, more transitions occur in CLBP_S patterns of textures with small Ra value than in textures with large Ra value. Second, CLBP is fast. The feature extraction time of CLBP is $O(N_I)$, where N_I is the number of pixels in an image. Third, CLBP is generally a local feature descriptor, while ADFT is global. The combination of CLBP and ADFT is considered to be better than each of them because they capture complementary information of an image.

4. Metrics for Classification

4.1 Metrics for Histogram Evaluation

To classify a test sample, we need to compute the similarity (or the distance) between the sample and each model. Ojala et al. [19] uses Log-likelihood statistic as follows:

$$L(T,M)=\sum_{b=1}^{B} T_b \log M_b \tag{9}$$

where B is the number of bins, T_b and M_b represent the probabilities of the test sample and the model in the b^{th} bin, respectively.

For CLBP [22], the chi-square distance is applied. The chi-square distance (χ^2 distance) between a test sample T and the model M is

$$D_{\chi^2}(T,M)=\sum_{b=1}^{B}(T_b-M_b)^2/(T_b+M_b) \tag{10}$$

where the usages of symbols are the same as that in Eq. (9).

Log-likelihood measures the similarity between a sample and a model. The bigger the log-likelihood is, the more similar the sample and the model are. While the chi-square distance measures the distance between the sample and the model, the smaller the distance is, the more similar the sample and the model are. The main difference between Log-likelihood and chi-square distance is that they assign different weights for small / large probabilities of the model. Chi-square distance outperforms log-likelihood when the probability of each bin is small [23]. Since there are 648 bins for $\mathrm{CLBP}_{P,R}^{\mathrm{riu2}}$, the probability in each bin is small, meaning that chi-square distance is better than log-likelihood. In this article, we prefer chi-square distance as the metric for CLBP.

Chi-square distance can be viewed as a weighted Euclidean distance (without the final square root operation), and the weight of small probability is larger than the weight of large probability. As Euclidean distance is a special case of l^p distance, it is naturally to substitute chi-square distance with other l^p distances as metrics for histogram evaluation. l^p distance is given as follows:

$$D_{l^p}(T,M)=\left(\sum_{n=1}^{N}|T_n-M_n|^p\right)^{1/p} \tag{11}$$

where p is any fixed real number that no less than one.

If large differences of probability need weight more than small ones, we should select a large p. Specifically, l^4 distance punishes more than l^2 distance for larger differences. For instance, $S_{eg}=(1,0,0)$ and $M_{eg}=(0.5,0.2,0.3)$ are two samples of a distribution with three bins. The absolute difference of S_{eg} and M_{eg} is $(0.5,0.2,0.3)$, resulting in $D_{l^4}(S_{eg},M_{eg})=0.5184$ and $D_{l^2}(S_{eg},M_{eg})=0.6164$. So that the l^4 distance is closer to the dominant difference (0.5) in the first bin. Consequently, the l^4 distance is generally determined by bins with large probability.

4.2 Metrics for ADFT

ADFT_M is the joint distribution of several discrete variables, where the variables refer to the quantized amplitudes of each frequency. If all the variables are mutually independent, we obtain

$$\begin{aligned}\log P(M_1,M_2,\cdots,M_N)&=\log\{P(M_1)P(M_2)\cdots P(M_N)\}\\&=\sum_{u=1}^{N}\log P(M_u)\end{aligned} \tag{12}$$

where M_u is the u^{th} amplitude of DFT coefficients.

As mentioned in Eq. (5), distribution of an amplitude variable is described in a discrete way, i.e. the distribution is with a histogram form. ADFT_M is a concatenation of all the histograms of each frequency. Under the independence assumption, Log-likelihood, chi-square distance and l^p distance still applies to ADFT_M. We utilize l^4 distance rather than chi-square distance as the metric for ADFT_MAs illustrated in Fig. 5, most of amplitudes are distributed in a few bins. These bins are more discriminative and robust because if the number of amplitudes change by one, the relative change is small. Therefore, we shall select a metric that large probability weight more than low probability. As discussed in the previous subsection, l^4 distance is proper for this kind of task. Experiments show that other distance, such as l^5 distance, are of comparative recognition rate as l^4 distance.

Theoretically, amplitudes of DFT coefficients are not independent with each other. The independence assumption is a simplification which results in fast recognition speed. To check whether the independence assumption works well, we also train a tree-structured probabilistic graphical model (Tree-PGM) [24] for each training sample. Tree-structured model is a Bayesian network which can capture conditional probabilities of two variables. Experiments on the roughness database (will be introduced in the next section) show that using tree-PGM, we can obtain 3.2% improvement of recognition rate. Because Tree-PGM give a similarity as log-likelihood, rather than a distance, it is difficult to tune tree-PGM with CLBP which uses chi-square distance. Therefore, we prefer l^4 distance as the metric for ADFT_M.

For ADFT_GP, chi-square distance is applied since that the distribution of ADFT_GP is more even than ADFT_M.

4.3 Combination of Metrics

Although ADFT_M , ADFT_GP and CLBP are all with histogram form, we apply different metrics to them. Chi-square distance is proper for CLBP and ADFT_GP , while l^4 distance is proper for ADFT. Different metrics are combined in a simple way as follows:

$$D(T, M) = \sum_{i=1}^{N_d} \lambda_i \frac{D_i(T, M)}{\mathrm{mean}_v(D_i)} \tag{13}$$

where λ_i is the weight of distance metric D_i, and $\mathrm{mean}_v(\cdot)$ is the average distance between a sample and the nearest v models in the training set. $\mathrm{mean}_v(\cdot)$ is a normalization term that make each distance metric weight nearly the same if λ_i is not considered. In our experiment, v is set to 5.

With this combined distance measure, the nearest neighborhood classifier is used to classify the test samples. In all our experiments, we fix the weights of ADFT_M, ADFT_GP, $\mathrm{LBP}_{P,R}^{\mathrm{riu2}}$ and $\mathrm{CLBP}_{P,R}^{\mathrm{riu2}}$ to 1, 1, 4 and 4, respectively.

5. Experiments and Discussion

The ADFT texture descriptor is designed for isotropic textures, especially casting textures that have dominant frequencies. We first introduce the casting texture database we acquired and then we evaluate the ADFT descriptor on the casting texture database. Finally, we apply ADFT to the well-known CUReT database [25], which contains 61 classes of real world textures, to demonstrate the effectiveness of ADFT.

5.1 Casting Textures Acquisition

All the specimens are taken by a Leica MZ16A stereomicroscopes, with two Mintron 73K80AHP digital cameras. The acquisition system for sampling are shown in Fig. 8. The magnification rate is about 0.4, resulting in the resolution of 770 dot per inch (DPI) in the final image. The two cameras are nearly parallel to each other, which means their x-axe are parallel and nearly collinear. Their fields of view are partially overlapped. The cameras use PAL system and the image resolution is 640*480 pixels. We keep only the center area of the images as the texture samples, with the size of 256*256 pixels. And the two cropped images are non-overlapping.

There are about 20 LED lights uniformly distributed under the surroundings of the objective lens. When the lights are turned on, the objective table are much more brighter than its surroundings. So that the specimens are considered to be under uniform luminance. The aperture of the microscope is adjusted to the minimum to obtain a large depth of field.

We sample the roughness textures from 7 roughness comparators, which contain 8 classes of roughness samples in total. Two groups of roughness textures we acquire are shown in Fig. 2. There are 253 valid samples, with each class contains 12 to 42 samples in variety. The average roughnesses (Ra) of different roughness classes are 3.2, 6.3, 12.5, 25, 50, 100, 800, 1600 μm, respectively.

5.2 Experimental Setup

We compare our ADFT feature with LBP [19], CLBP [22], VZ-MR8 [9], and VZ-Joint [26], Energy Ring [1], Circular DFT [11] and Local Fourier Histogram (LFH) [12]. Experimental settings of the last three methods are given below.
VZ-MR8 (10 textons per class) [9]: Images are firstly gray-scale normalized and then 38 filters are applied to the image. The MR8 filter bank consists of two rotationally symmetric

Fig. 8 The casting texture acquisition system.

and two kinds of anisotropic filters. The symmetric ones are a Gaussian and a Laplacian of Gaussian. The anisotropic ones are first order and second order Gaussian derivative filters. Six orientations of the anisotropic filters are utilized at three scales and only the maximum responses in each scale are kept as the features. So that the VZ-MR8 feature is eight dimensional in total. Filter responses of training samples are clustered using k-means to produce 10 textons per class, resulting in $10 * Nc$ textons that partition the feature space into Voronoi cells. Each filter response of each pixel in a image is labeled by the nearest texton and a normalized histogram of the labels are computed for each image as its descriptor. In the classification stage, the test image is assigned to the nearest model. And the chi-square distance is utilized as the metric.

VZ-MR8 (40 textons per class) [9]: The method is just as above, except that filter responses are clusted into 40 textons per class.

VZ-Joint 7 × 7 (40 textons per class) [26]: The methodology is just as the VZ-MR8 method, except that the textons are gray values of 7×7 image patches.

Energy Ring [1]: Images are firstly transformed using 2D-DFT. Energies in different rings or circles are extracted as the feature descriptor.

CircularDFT [11]: Amplitudes of DFT coefficients on circular neighborhoods are calculated. The means and standard deviations of each amplitude are extracted as the feature descriptor. And the chi-square distance is utilized as the metric.

LFH [12]: Amplitudes of DFT coefficients on 3×3 window (the central pixel is excluded) are quantized. A histogram of these quantized amplitudes are calculated the same way as ADFT. And as the original pater mentioned, we utilize the l^1 distance for LFH.

5.3 Experiment #1 Roughness Classification

The roughness texture database we acquire contains 8 classes of roughness patterns. In all experiments, half of the images in each class are selected randomly as training images and the left as testing images. The recognition rates listed in Table 2 are the average results over 100 independent experiments. Although VZ-Joint 7 × 7 approach and VZ-MR8 perform well on other databases, e.g., Outex database [27] and CURet database [25], their recognition rate on our roughness database are relatively lower. The reasons are that colors of all specimens with the same material are all similar and the great intra-class. Different from these two approaches, LBP captures the local feature of the specimens using only the sign of differences between adjacent pixels, still $\mathrm{LBP}^{\mathrm{riu2}}_{8,1}$ obtains the recognition rate of 86.1%, which is a little bit lower than VZ-Joint 7 × 7's 86.2%. Our proposed ADFT approach obtain the recognition rate of 88.4%, which outperforms other DFT-based methods. As an improved version of LBP, CLBP get the highest recognition rate among single descriptors. Noting that recognition rate of ADFT_M using l^4 distance is lower than ADFT_M using Tree-PGM. But to combine ADFT_M with CLBP in a simple way, we prefer ADFT_M with l^4 distance here.

LBP and CLBP capture local information of the textures, while the two DFT approaches capture the global information of the textures in frequency domain, the combination of them are expected to improve the recognition rate. And this is also proved by the experimental results. Even though the performance of energy ring is not well, the combination of energy ring and CLBP still outperforms CLBP. This fact further demonstrates that global information in the frequency is complementary with local information such as CLBP.

The best recognition rate is obtained by the combination of ADFT and $\mathrm{CLBP}^{\mathrm{riu2}}_{25,9}$, which improves the recognition rate by 2.6 percent compared with $\mathrm{CLBP}^{\mathrm{riu2}}_{25,9}$. And the total computation burden is still low, which takes only 0.038 second to extract the combined descriptor. The implementation of ADFT are in matlab while CLBP is implemented using C++, further improvements can be achieved by computing DFT of each row and column parallelly. So that the approach is proper for real-time application.

We utilize different metrics for CLBP and ADFT_M and ADFT_GP, and different metrics are combined using Eq. (13). The reason for this consideration is two-fold. 1) χ^2 distance has been proved to be a good metric for histogram comparison and is widely used for LBP descriptor families. 2) We assume that roughness samples have their dominant frequencies. Because l^4 distance punishes large differences more than small ones, still, it is proved by experiment that it is proper for the ADFT_M descriptor.

Finally, we list the confusion matrix of ADFT+$\mathrm{CLBP}^{\mathrm{riu2}}_{25,9}$ approach in Table 1. Each row in Table 1 corresponds to one class of sample that are classified into different classes. For instance, the first row represents that 99.8% of samples with Ra 3.2 are classified into the class with Ra 3.2 and the other 0.2% of samples with Ra 3.2 are classified into the class with Ra 6.3. As we can see the misclassified ones are generally classified to their adjacent classes, this property makes the descriptor more proper for production evaluation.

5.4 Experiment #2 Evaluation on the CUReT Database

The CUReT database contains 61 classes of textures. We follow the setup of M. Varma [9] here, which means that 92

Table 1 Confusion matrix (%) of ADFT+$\mathrm{CLBP}^{\mathrm{riu2}}_{25,9}$.

	3.2	6.3	12.5	25	50	100	800	1600
3.2	99.8	0.2	0.0	0	0	0	0	0
6.3	0.0	91.2	8.8	0	0	0	0	0
12.5	0	2.9	96.3	0.5	0.3	0	0	0
25	0	0.0	1.1	96.6	2.3	0	0	0
50	0	0	1.4	2.3	95.8	0.5	0	0
100	0	0	0	0.0	0.5	98.5	0.9	0.1
800	0	0	0	0	0	0.4	98.6	1.0
1600	0	0	0	0	0	0	1.4	98.6

Table 2 Recognition rate and feature extraction time on roughness database.

		distance metric	recognition rate (%)	feature extraction time (s)
Global descriptors	Energy ring	χ^2	73.3	0.008
	ADFT_SM	l^4	86.7	0.018
	ADFT_M	l^4	85.6	0.019
	ADFT_GP	χ^2	81.3	0.019
	ADFT	$\chi^2 + l^4$	88.4	0.024
Local descriptors	MR8(40centers/class)	χ^2	81.9	0.7
	VZ-Joint 7×7(40centers/class)	χ^2	86.2	6.4
	$\mathrm{LBP}_{8,1}^{riu2}$	χ^2	86.1	0.003
	$\mathrm{CLBP}_{8,1}^{riu2}$	χ^2	86.6	0.007
	$\mathrm{CLBP}_{16,2}^{riu2}$	χ^2	91.8	0.011
	$\mathrm{CLBP}_{25,9}^{riu2}$	χ^2	94.0	0.014
	CircularDFT	χ^2	61.0	0.086
	LFH	l^1	76.9	0.023
Global+Local descriptors	Energy ring+$\mathrm{CLBP}_{25,9}^{riu2}$	χ^2	93.5	0.022
	ADFT+$\mathrm{LBP}_{8,1}^{riu2}$	$\chi^2 + l^4$	91.8	0.027
	ADFT+$\mathrm{CLBP}_{8,1}^{riu2}$	$\chi^2 + l^4$	92.5	0.031
	ADFT+$\mathrm{CLBP}_{16,2}^{riu2}$	$\chi^2 + l^4$	94.7	0.033
	ADFT+$\mathrm{CLBP}_{25,9}^{riu2}$	$\chi^2 + l^4$	**96.6**	0.038

Table 3 Classification Rate (%) on CUReT database.

	descriptor	number of training samples			
		46	23	12	6
Global descriptors	ADFT_SM	84.57	77.40	69.38	59.55
	ADFT_M	87.04	79.96	71.72	61.61
	ADFT_GP	66.69	61.09	55.19	48.26
	ADFT	90.75	84.80	77.27	67.88
Local descriptors	$\mathrm{LBP}_{8,1}^{riu2}$	80.87	74.92	67.60	58.73
	$\mathrm{CLBP}_{8,1}^{riu2}$	95.54	91.15	84.53	74.60
	$\mathrm{CLBP}_{16,2}^{riu2}$	95.62	91.63	85.46	76.05
	$\mathrm{CLBP}_{24,5}^{riu2}$	94.53	90.02	83.36	73.78
	CircularDFT	75.56	66.72	57.14	46.51
	LFH	77.90	70.63	62.91	54.11
Global +Local descriptors	ADFT+$\mathrm{LBP}_{8,1}^{riu2}$	94.91	90.35	83.61	74.49
	ADFT_M+$\mathrm{CLBP}_{8,1}^{riu2}$	96.25	92.51	86.51	77.42
	ADFT+$\mathrm{CLBP}_{8,1}^{riu2}$	96.88	93.41	87.73	78.87
	ADFT_M+$\mathrm{CLBP}_{16,2}^{riu2}$	97.03	93.74	88.27	79.87
	ADFT+$\mathrm{CLBP}_{16,2}^{riu2}$	**97.28**	**94.24**	**89.03**	**80.74**
	ADFT_M+$\mathrm{CLBP}_{24,5}^{riu2}$	96.47	93.07	87.79	79.12
	ADFT+$\mathrm{CLBP}_{24,5}^{riu2}$	96.80	93.56	88.18	79.93

out of 205 images of each class are selected and cropped to 200*200 pixels. The images are acquired at different illumination orientations and view points, and the viewing angles of the selected images are less than 60°.

Although the ADFT texture descriptor is designed for roughness classification, we compared it with other approaches on the CUReT database. All descriptors except ADFT are announced to be rotationally invariant. As shown in Table 3, although ADFT is not rotationally invariant, the recognition rate of ADFT is 90.75% when we use half of the samples as training samples. The reason for this result is that we extract ADFT is two directions, which means that it is invariant if the rotation angle is multiples of 90°. The high recognition rate of ADFT illustrates that distribution of one-dimensional DFT coefficients is a discriminative feature for different textures. However, ADFT_GP performs badly on CUReT database, which means that ADFT_GP is not a good descriptor for anisotropic textures.

As shown in Table 3, combinations of ADFT and CLBP always achieve better recognition rates than the corresponding CLBP alone does. This result confirms again that global information in the frequency is complementary with local information such as CLBP.

6. Conclusion

In this article, we propose ADFT which utilizes the isotropic characteristics of roughness specimens. ADFT_M and ADFT_GP describe textures using the distributions of amplitudes and distributions of phase gradients, respectively. Both ADFT_M and ADFT_GP are discriminative for isotropic textures such as casting images. Combined with the well-known CLBP descriptor, which captures the local features of textures, the CLBP+ADFT descriptor captures texture features in a both global and local way. The fused feature still can be computed efficiently, which is key to on-line evaluation of the roughness levels of products. Experiments on CUReT database show that ADFT_M is also suitable for describing anisotropic textures. How to make the descriptor rotation invariant and make phase information useful for anisotropic textures will be studied in the future work.

Acknowledgements

This work was supported by the National Natural Science Foundation of China Grant (No. 61271393 and 61301183).

References

[1] D.M. Tsai and C.F. Tseng, "Surface roughness classification for castings," Pattern Recognit., vol.32, no.3, pp.389-3-405, March 1999.

[2] E.S. Gadelmawla, "A vision system for surface roughness characterization using the gray level co-occurrence matrix," NDT & E International, vol.37, no.7, pp.577–588, Oct. 2004.

[3] R. Kumar, P. Kulashekar, B. Dhanasekar, and B. Ramamoorthy, "Application of digital image magnification for surface roughness evaluation using machine vision," Int. J. Machine Tools and Manufacture, vol.45, no.2, pp.228–234, Feb. 2005.

[4] G.A. Al-Kindi and B. Shirinzadeh, "Feasibility assessment of vision-based surface roughness parameters acquisition for different types of machined specimens," Image Vis. Comput., vol.27, no.4, pp.444–458, March 2009.

[5] B.H. McCormick and S.N. Jayaramamurthy, "Time series model for texture synthesis," Int. J. Parallel Programming, vol.3, no.4, pp.329–343, Dec. 1974.

[6] R. Chellappa and S. Chatterjee, "Classification of textures using Gaussian Markov random fields," IEEE Trans. Acoust. Speech Signal Process., vol.33, no.4, pp.959–963, Aug. 1985.

[7] F.S. Cohen, Z. Fan, and M.A.S. Patel, "Classification of rotated and scaled textured images using Gaussian Markov random field models," IEEE Trans. Pattern Anal. Mach. Intell., vol.13, no.2, pp.192–202, Feb. 1991.

[8] J.G. Daugman, "Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters.," J. Opt. Soc. Am. A, Optics and Image Science, vol.2, no.7, pp.1160–1169, July 1985.

[9] M. Varma and A. Zisserman, "A Statistical Approach to Texture Classification from Single Images," Int. J. Comput. Vis., vol.62, no.1-2, pp.61–81, April 2005.

[10] D.C. He and L. Wang, "Texture features based on texture spectrum," Pattern Recognit., vol.24, no.5, pp.391–399, Jan. 1991.

[11] H. Arof and F. Deravi, "Circular neighbourhood and 1-D DFT features for texture classification and segmentation," Vision, Image and Signal Processing, IEE Proceedings -, vol.145, no.3, pp.167–172, June 1998.

[12] F. Zhou, J.F. Feng, and Q.Y. Shi, "Texture feature based on local Fourier transform," 2001 International Conference on Image Processing, 2001. Proceedings., pp.610–613 vol.2, Oct. 2001.

[13] U.A. Ahmad, K. Kidiyo, and R. Joseph, "Texture features based on Fourier transform and Gabor filters: an empirical comparison," International Conference on Machine Vision, 2007. ICMV 2007. pp.67–72, Dec. 2007.

[14] T. Chang and C.C.J. Kuo, "Texture analysis and classification with tree-structured wavelet transform," IEEE Trans. Image Process., vol.2, no.4, pp.429–441, Oct. 1993.

[15] M. Unser, "Texture classification and segmentation using wavelet frames," IEEE Trans. Image Process., vol.4, no.11, pp.1549–1560, Nov. 1995.

[16] M.M. Galloway, "Texture analysis using gray level run lengths," Computer Graphics and Image Processing, vol.4, no.2, pp.172–179, June 1975.

[17] R.M. Haralick, K. Shanmugam, and I. Dinstein, "Textural features for image classification," IEEE Trans. Syst., Man Cybern., vol.SMC-3, no.6, pp.610–621, Nov. 1973.

[18] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," Pattern Recognit., vol.29, no.1, pp.51–59, Jan. 1996.

[19] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," IEEE Trans. Pattern Anal. Mach. Intell., vol.24, no.7, pp.971–987, July 2002.

[20] N. Doshi and G. Schaefer, "A comparative analysis of local binary pattern texture classification," 2012 IEEE, Visual Communications and Image Processing (VCIP), pp.1–6, Nov. 2012.

[21] V. Ojansivu and J. Heikkilä, "Blur Insensitive Texture Classification Using Local Phase Quantization," in Image and Signal Processing, ed. A. Elmoataz, O. Lezoray, F. Nouboud, and D. Mammass, Lecture Notes in Computer Science, vol.5099, ch. 27, pp.236–243, Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

[22] Z. Guo, L. Zhang, and D. Zhang, "Rotation invariant texture classification using LBP variance (LBPV) with global matching," Pattern Recognit., vol.43, no.3, pp.706–719, March 2010.

[23] T. Ahonen, A. Hadid, and M. Pietikäinen, "Face Recognition with Local Binary Patterns," in Computer Vision - ECCV 2004, ed. T. Pajdla and J. Matas, Lecture Notes in Computer Science, vol.3021, ch. 36, pp.469–481, Springer Berlin Heidelberg, Berlin, Heidelberg, 2004.

[24] D. Kollar and N. Friedman, Probabilistic graphical models: principles and techniques, MIT Press, 2009.

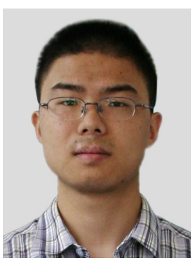
[25] K.J. Dana, B. van Ginneken, S.K. Nayar, and J.J. Koenderink, "Reflectance and texture of real-world surfaces," ACM Trans. Graph., vol.18, no.1, pp.1–34, Jan. 1999.

[26] M. Varma and A. Zisserman, "A Statistical Approach to Material Classification Using Image Patch Exemplars," IEEE Trans. Pattern Anal. Mach. Intell., vol.31, no.11, pp.2032–2047, Nov. 2009.

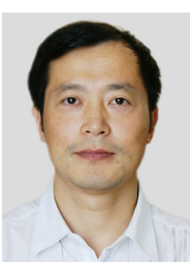
[27] T. Ojala, T. Maenpaa, M. Pietikainen, J. Viertola, J. Kyllonen, and S. Huovinen, "Outex - new framework for empirical evaluation of texture analysis algorithms," 16th International Conference on Pattern Recognition, 2002. Proceedings. pp.701–706 vol.1, 2002.

Chao Liang received the bachelors degree in information science and engineering from Southeast University, Nanjing, China in 2009. He is currently working toward the PhD degree in Department of electronics engineering, Tsinghua University, Beijing, China. His research interests include texture classification and machine learning.

Wenming Yang received the B.S. and M.S. in material science from Harbin University of Science & Technology, Harbin, China in 2000 and 2003, respectively. In 2006, he received the Ph.D in electronic engineering from Zhejiang University, Hangzhou, China. From 2007 to 2009, He was a Postdoctoral Fellow in Department of Electronic Engineering, Tsinghua University. Since 2010, He has been a lecturer in Department of Electronic Engineering at Graduate School at Shenzhen, Tsinghua University. His research interests include image processing and pattern recognition, computer vision, biometrics, video surveillance, image super-resolution and non-destructive testing based on artificial vision.



Fei Zhou was born in Jingzhou, HuBei, China, in 1984. He received the B.Eng. degree from Department of Electronics and Information Engineering, Huazhong University of Science and Technology, Wuhan, China, in 2007, and received the Ph.D. degrees from Department of electronics engineering, Tsinghua University, Beijing, China, in 2013. From 2013, he has worked as a postdoctoral fellow with Shenzhen Graduate School, Tsinghua University, China. His research interests include applications of image processing and pattern recognition in video surveillance, image super-resolution, image interpolation, image quality assessment, and object tracking.



Qingmin Liao was born in 1963. He received the B.S. degree in radio technology from the University of Electronic Science and Technology of China, Chengdu, China, in 1984, and the M.S. and Ph.D. degrees in signal processing and telecommunications from the University of Rennes 1, Rennes, France, in 1990 and 1994, respectively. Since 1995, he has been joining with Tsinghua university, Beijing, China. He became Professor in the Department of Electronic Engineering of Tsinghua University, in 2002. From 2001 to 2003, he served as the Invited Professor with a tri-year contract at the University of Caen, France. Since 2010, he has been Director of the Division of Information Science and Technology in the Graduate School at Shenzhen, Tsinghua University, Shenzhen, China. His research interests include image video processing, transmission and analysis; biometrics; and their applications to medicine, industry, and sports. He has published over 90 papers internationally.