Improved Tamura Features for Image Classification using Kernel based Descriptors

Priyabrata Karmakar[1], Shyh Wei Teng[1], Dengsheng Zhang[1], Ying Liu[2], Guojun Lu[1]
1 {pkarmakar, shyh.wei.teng, dengsheng.zhang, guojun.lu}@federation.edu.au
2 {ly_yolanda@sina.com}
1 School of Engineering and Information Technology, Faculty of Science and Technology
Federation University Australia, Gippsland campus, Churchill, Victoria, Australia-3842
2 Center for Image and Information Processing
Xi'an University of Posts and Telecommunications, China-710121

***Abstract*—Tamura features are based on human visual perception and have huge potential in image representation. Conventional Tamura features only work on homogeneous texture images and perform poor on generic images. Therefore, many researchers attempt to improve Tamura features and most of the improvements are based on histogram based representation. Kernel descriptors have been shown to outperform existing histogram based local features as such descriptors do not require coarse quantization of pixel attributes. Instead, in kernel descriptor framework, each pixel equally participates in matching between two image patches. In this paper, we propose a set of kernel descriptors that are based on Tamura features. Additionally, the proposed descriptors are invariant to local rotations. Experimental results show that our proposed approach outperforms the conventional Tamura features significantly.**

***Index Terms*—Tamura features, Kernel descriptor, Rotation invariance, Image classification.**

I. Introduction

Due to the rapid increase in the number of digital images generated and stored on storage devices, an effective image representation is very crucial stage to manage images for different image processing applications like, classification, retrieval and detection. Images can be represented using different image features which can basically be classified into two broad categories: global and local features. Global features (a vector or a single entity) describe an image as a whole, such as, the group of MPEG-7 descriptors [1]. Whereas, local features aim to represent a region or a patch of an image, such as, SIFT [2] and HOG [3]. Image features can be further classified based on pixel properties, such as gradient, colour and texture. Texture is an important image feature. It provides the surface characteristics and appearance of the objects present in images. Tamura features are important texture features which are based on human visual perception. Authors of [4] have proposed a total of six features: Coarseness, Directionality, Contrast, Line-likeness, Regularity and Roughness. However, as per human visual perception, the first three features are very important as they are more effective for distinguishing different textures. Many of the subsequent research on Tamura features are done mainly on these first three features.

Tamura features are global features which are computed from the whole image without considering any homogeneity constraint. This causes weaker performance of Tamura features compared to other texture features. Therefore in [5], Coarseness and in [6], both Coarseness and Directionality features are modified into histogram based features which perform better than the conventional features. Coarseness is further improved in [7], where a model is learned which captures Coarseness by establishing a relation between computational measure and human perception. Tamura directionality uses the statistical property of directional histogram and it is prone to miscalculation of the directionality of an image. By using the geometric property of directional histogram, this limitation was overcome in [8]. Contrast is modified to provide local brightness information instead of a single global entity by using statistical moments of intensity histogram [9]. Application of Tamura features is not limited to only texture image processing. For natural scene classification, authors of [10] have used Tamura features as one of the components for their proposed human inspired feature vector. In [11], for each pixel, three main Tamura features are calculated and then they are represented using a 3D histogram like 3D-colour histograms. Similar 3D histogram of Tamura features are used in [12] for object recognition. Conventional Tamura features are for 2D images. Recently, authors of [13] have extended it to work with 3D images as well.

Since the proposal of Tamura features, it has been modified and improved by many researchers. In most cases, Tamura features are improved as a histogram based feature instead of a single global entity. However, it is known that histogram based features lose the discriminative property because they do not capture the spatial information of an image. Therefore, in [14], authors have proposed kernel descriptors (KDES). In KDES framework, the similarity measure between the image patches is calculated using a kernel function which is referred as match kernels. Match kernels are defined over pixel attributes (gradient, colour and local binary pattern (LBP)). After that descriptors are extracted from these match kernels.

This research was partially supported by Australian Research Council Discovery Projects scheme: DP130100024

978-1-5386-2839-3/17/$31.00 ©2017 IEEE

Originally, kernel descriptors are proposed for recognition task, but later on the applications are extended towards depth image classification [15], [16], action recognition [17] and material recognition [18].

KDES framework can turn any kind of pixel attributes to a patch based descriptor which inherits the properties of an underlying match kernel that measures the similarity between different patches. Tamura features are based on human visual perception which makes them very significant in image representation. However, the conventional and the subsequent modified versions could not explore the potential of Tamura features in an appropriate way. Therefore, there is a gap in research to effectively use Tamura features. Additionally, there is no such work done on Tamura features to embed rotation invariance into the features. In this paper, we propose a kernel descriptor approach for Tamura features where per pixel based Tamura features (attributes) equally participates into the matching between two different image patches. Moreover, we have included rotation invariance into the patch based features to deal with the local rotation effects in the texture images.

The rest of paper is organised as follows. Section II discusses related works. Section III discusses the proposed work followed by experiment and results in Section IV. Finally, Section V concludes the paper.

II. Related Works

In this section, we will briefly describe the two main works which are related to this paper.

A. Tamura Features

Here, we will discuss the first three Tamura features as per [4].

Coarseness: It measures the image granularity. It is calculated as the average of the largest window sizes needed to identify the texture elements centred at different pixel positions. An image may contain texture at different scales. Coarseness aims to find out the largest size at which a texture exists, even if a smaller micro texture exists.

The calculation of coarseness is summarised as,

1) At each pixel (x, y), compute six averages for window of sizes $2^k \times 2^k, k = 0, 1, \cdots, 5$ around the pixel. $A_k(x,y) = \sum_{i=(x-2^{k-1})}^{(x+2^{k-1}-1)} \sum_{j=(y-2^{k-1})}^{(y+2^{k-1}-1)} \frac{g(i,j)}{2^{2k}}$, where $g(i,j)$ is the pixel intensity at (i,j).
2) At each pixel, compute absolute differences $E_k(x,y)$ between the pairs of non-overlapping averages in the horizontal (h) and vertical (v) directions.

$$E_{k,h}(x,y) = |A_k(x+2^{k-1},y) - A_k(x-2^{k-1},y)|$$

$$E_{k,v}(x,y) = |A_k(x,y+2^{k-1}) - A_k(x,y-2^{k-1})|$$

3) At each pixel, find the value of k that maximises the difference $E_k(x,y)$ in either direction and set the best size , $S_{best}(x,y) = 2^k$.
4) Compute the coarseness feature (f_{coarse}) by averaging $S_{best}(x,y)$ over the entire image.

Directionality: Directionality conveys the existence of any directional pattern (vertical, horizontal or diagonal) in an image. It describes globally how the texture in the image is distributed or concentrated along certain orientations. Tamura directionality is calculated from the edge histogram H_D. It includes the following steps,

1) The gray-scale image is convolved with horizontal and vertical edge (Prewitt) operators.
2) Say, for a particular pixel in the image, after applying the Prewitt operator, the outputs are ΔH and ΔV . Edge of a pixel is a vector and it has both magnitude ($|\Delta G|$) and direction (θ) which are calculated as, $|\Delta G| = \frac{|\Delta H| + |\Delta V|}{2}$, $\theta = tan^{-1}\frac{|\Delta V|}{|\Delta H|} + \frac{\pi}{2}$. H_D is calculated by quantizing θ $(0 \leq \theta < \pi)$ and counting the number of pixels with magnitude greater than a pre-defined threshold.
3) After calculating the H_D, all peaks and valleys in H_D are identified. If there are n peaks in the histogram, for each peak i, let w_i be the window of bins from the previous valley to the next valley (a window contains a peak in it) and ϕ_i be the angular position of the peak in w_i. Let, $H_D(\phi)$ be the height of a bin at angular position ϕ . The Tamura directionality (f_{dir}) from the sharpness of H_D is calculated as: $f_{dir} = 1 - r \times n \times \sum_{i=1}^{n} \sum_{w_i \in \phi} (\phi - \phi_i)^2 \times H_D(\phi)$, where r is the normalizing factor related to the quantizing levels of ϕ.

Contrast: Contrast measures how gray-level intensity of the pixels vary in the image and to what extent their distribution is biased to black or white. Contrast (f_{con}) is measured as, $f_{con} = \frac{\sigma}{(\alpha^4)^n}$, σ is the standard deviation of the gray-level histogram, α is the kurtosis of the gray-level histogram and n is set as 0.25.

B. Kernel descriptors

Authors of [14] have proposed three kernel descriptors based on three-pixel attributes: gradient, colour and LBP pattern. In this section, we will briefly describe the gradient kernel descriptor (GKDES) to present an overview about defining a match kernel on pixel attributes and how kernel descriptors are extracted from the match kernels. Gradient match kernel is defined by (1),

$$K_{grad}(A,B) = \sum_{z \in A} \sum_{z' \in B} \tilde{m}(z)\tilde{m}(z') k_o(\tilde{\theta}(z), \tilde{\theta}(z')) k_p(z,z') \quad (1)$$

where A and B are two different patches. $\tilde{m}(z)$ represents the normalized gradient magnitudes of pixels in Patch A. $\tilde{m}(z)\tilde{m}(z')$, a linear kernel can also be represented as, $k_{\tilde{m}}(z,z')$. It provides a weight to the contribution of each pixel using gradient magnitudes to the overall match of K_{grad}. $k_o(\tilde{\theta}(z), \tilde{\theta}(z')) = \exp(-\gamma_o \|\tilde{\theta}(z) - \tilde{\theta}(z')\|^2)$ is a Gaussian kernel over gradient orientations. It computes the similarity of gradient orientations. $k_p(z,z') = \exp(-\gamma_p \|z - z'\|^2)$ is a Gaussian kernel over 2D position of pixels inside a

patch and z (or z') denotes the 2D position. k_p measures how close two pixels are spatially. Direct kernel computation over image patches is a computationally expensive task when the number of images is large. Therefore, the authors of [14] have proposed a low-dimensional feature representation approach from the match kernels. Candidate kernels of K_{grad} given by (1) can be written in terms of their inner products. Such as, $k_{\tilde{m}}(z,z') = \phi_{\tilde{m}}(z)^T\phi_{\tilde{m}}(z')$, $k_o(\tilde{\theta}(z),\tilde{\theta}(z')) = \phi_o(\tilde{\theta}(z))^T\phi_o(\tilde{\theta}(z'))$, $k_p(z,z') = \phi_p(z)^T\phi_p(z')$, where $\phi_{\tilde{m}}(.)$, $\phi_o(.)$ and $\phi_p(.)$ are the feature maps of $k_{\tilde{m}}$, k_o and k_p respectively. So the descriptor can be extracted as,

$$F_{grad}(A) = \sum_{z\in A} \phi_{\tilde{m}(z)}\phi_o(\tilde{\theta}(z)) \otimes \phi_p(z) \tag{2}$$

where $\otimes$ is Kronecker product. F_{grad} is called gradient kernel descriptor as it is derived from gradient match kernel.

Being a linear kernel, the feature map of $k_{\tilde{m}}$ is normalized gradient magnitudes only, i.e., $\phi_{\tilde{m}}(z) = \tilde{m}(z)$. For the other two candidate kernels, feature maps cannot be extracted implicitly as they are non-linear (Gaussian) kernels. Therefore, feature maps $\phi_o(.)$ and $\phi_p(.)$ are extracted explicitly by approximating them over a set of basis vectors. For example, if $k_o(\tilde{\theta}(z),\tilde{\theta}(z'))$ is considered, $\{x_i\}_{i=0}^d \in X$ are sampled normalized gradient vectors and $\{\phi_o(x_i)\}_{i=0}^d$ is the set of basis vectors which are obtained by mapping x_i over the same kernel space as k_o, then k_o can be re-written as,

$$\begin{aligned} k_o(\tilde{\theta}(z),\tilde{\theta}(z')) &= k_o(\tilde{\theta}(z),X)^T[K_o^{-1}]_{ij}k_o(\tilde{\theta}(z),X) \\ &= [Gk_o(\tilde{\theta}(z),X)]^T[Gk_o(\tilde{\theta}(z'),X)] \end{aligned} \tag{3}$$

where $k_o(\tilde{\theta}(z),X) = [k_o(\tilde{\theta}(z),x_1),\cdots\ k_o(\tilde{\theta}(z),x_d))]$ is a $d\times 1$ vector, K_o is a $d\times d$ matrix, $(i,j)^{th}$ element of K_o is $k_o(x_i,x_j)$ and $K_o^{-1} = G^TG$ obtained by applying Cholesky decomposition. So, the approximated feature map is $\tilde{\phi_o}(\tilde{\theta}(z)) = [Gk_o(\tilde{\theta}(z),X)]$. $\tilde{\phi_o}(\tilde{\theta}(z))$ is an explicit approximation of $\phi_o(\tilde{\theta}(z))$. Similarly, $\phi_p(z)$ is approximated as $\tilde{\phi_p}(z)$. So, (2) can be approximated as,

$$\tilde{F}_{grad}(A) = \sum_{z\in A} \phi_{\tilde{m}(z)}\tilde{\phi_o}(\tilde{\theta}(z)) \otimes \tilde{\phi_p}(z) \tag{4}$$

$\tilde{F}_{grad}(A)$ is the approximation of $F_{grad}(A)$. $\tilde{F}_{grad}(A)$ is originally of high dimension. Therefore, it is reduced to a acceptable dimension by applying kernel principal component analysis (KPCA).

III. Proposed Work

A. Proposed Tamura Kernel descriptors

In this section, we present our proposed kernel descriptor approach based on Tamura features. We propose three kernel descriptors on Coarseness, Directionality and Contrast. At first, we define Coarseness match kernel given by (5),

$$K_{coarse}(A,B) = \sum_{z\in A}\sum_{z'\in B} k_s(s(z),s(z'))k_p(z,z') \tag{5}$$

where $k_s(s(z),s(z')) = \exp(-\gamma_s\|s(z)-s(z')\|^2)$, a Guassian kernel finds the similarity between S_{best} values. $s(z)$ represents the S_{best} value of a pixel at z. S_{best} values for individual pixels are obtained by considering up to the first three stages of Tamura Coarseness computation. $k_p(z,z') = \exp(-\gamma_p\|z-z'\|^2)$ is a Gaussian kernel over 2D position of pixels inside a patch and z (or z') denotes the 2D position. It measures how close two pixels are spatially. Both S_{best} and 2D pixel position values are normalized in the range [0,1].

To extract descriptor from K_{coarse}, we have considered the same approach as in [14] and the Coarseness kernel descriptor (CorKDES) is given by

$$F_{coarse}(A) = \sum_{z\in A} \phi_s(s(z)) \otimes \phi_p(z) \tag{6}$$

where $\phi_s(.)$ and $\phi_p(.)$ are the feature maps of k_s and k_p respectively. As mentioned before, feature maps of k_s and k_p can not be extracted implicitly as they are non-linear (Gaussian) kernels. Therefore, feature maps are extracted explicitly by approximating them over a set of basis vectors. We have considered the same approach as in [14] to approximate the feature maps of k_s and k_p as $\tilde{\phi_s}(s(z))$ and $\tilde{\phi_p}(z)$. So, (6) is approximated as,

$$\tilde{F}_{coarse}(A) = \sum_{z\in A} \tilde{\phi_s}(s(z)) \otimes \tilde{\phi_p}(z) \tag{7}$$

We chose the size of basis vectors on kernel k_s as 5. The basis vectors are the normalized (within $[0,1]$) '2^k' $(k = 0,1,\cdots,5)$ values. The size of basis vectors on kernel k_p is chosen as 5×5 which is same as in [14]. To compute $\tilde{F}_{coarse}$, (7) involves with Kronecker product. So, the dimension of $\tilde{F}_{coarse}$ is $5\times 25 = 125$ which is an acceptable dimension to process the descriptor. Therefore, there is no need of dimension reduction of the descriptors. Now, we define the match kernels related to Directionality and Contrast by (8) and (9) respectively.

$$K_{dir}(A,B) = \sum_{z\in A}\sum_{z'\in B} k_\psi(\psi(z),\psi(z'))k_p(z,z') \tag{8}$$

where $k_\psi(\psi(z),\psi(z')) = \exp(-\gamma_\psi\|\psi(z)-\psi(z')\|^2)$, a Gaussian kernel which finds the similarity between the normalized (within [0,1]) ψ values of pixels at z and z'. For each pixel, edge magnitude ($|\Delta G|$) and direction (θ) are calculated in the same way as in Tamura directionality. $\psi(z) = \theta(z)$ for the corresponding $|\Delta G| > t$, where t is fixed as 12 as per [4]. Otherwise $\psi(z) = \varepsilon(\varepsilon\to 0)$. Thresholding $|\Delta G|$ by t helps to reject unreliable directions which cannot be considered as edge points.

$$K_{contrast}(A,B) = \sum_{z\in A}\sum_{z'\in B} k_{con}(con(z),con(z'))k_p(z,z') \tag{9}$$

where $k_{con}(con(z),con(z')) = \exp(-\gamma_{con}\|con(z) - con(z')\|^2)$, a Gaussian kernel finds the similarity between normalized (with [0,1]) contrast values at z and z'. Contrast

(1,1)	(1,2)	(1,3)	(1,4)	(1,5)	(1,6)	(1,7)	(1,8)
(2,1)	(2,2)	(2,3)	(2,4)	(2,5)	(2,6)	(2,7)	(2,8)
(3,1)	(3,2)	(3,3)	(3,4)	(3,5)	(3,6)	(3,7)	(3,8)
(4,1)	(4,2)	(4,3)	(4,4)	(4,5)	(4,6)	(4,7)	(4,8)
(5,1)	(5,2)	(5,3)	(5,4)	(5,5)	(5,6)	(5,7)	(5,8)
(6,1)	(6,2)	(6,3)	(6,4)	(6,5)	(6,6)	(6,7)	(6,8)
(7,1)	(7,2)	(7,3)	(7,4)	(7,5)	(7,6)	(7,7)	(7,8)
(8,1)	(8,2)	(8,3)	(8,4)	(8,5)	(8,6)	(8,7)	(8,8)

Fig. 1: 2D pixel position

4	4	4	4	4	4	4	4
4	3	3	3	3	3	3	4
4	3	2	2	2	2	3	4
4	3	2	1	1	2	3	4
4	3	2	1	1	2	3	4
4	3	2	2	2	2	3	4
4	3	3	3	3	3	3	4
4	4	4	4	4	4	4	4

Fig. 2: Rotation Invariant pixel position

values per pixel is calculated as, $con(z) = (I - mn)/std.$ where I is the intensity of pixel at z, mn and std are the mean and standard deviation of the pixel intensities around the 3×3 neighbourhood of the pixel at z. Kernel descriptors $\tilde{F}_{dir}$ (DirKDES) and $\tilde{F}_{contrast}$ (ConKDES) are extracted from K_{dir} and $K_{contrast}$ respectively in the same way as $\tilde{F}_{coarse}$ extracted from K_{coarse}. We chose the size of basis vectors on both k_ψ and k_{con} as 10. This reflects that for both $\tilde{F}_{dir}$ and $\tilde{F}_{contrast}$, descriptor dimension is 250 ($10 \times 25 = 250$).

B. Rotation Invariant position kernel

To find the spatial closeness between two pixels belong to two different patches, authors of [14] have proposed k_p, a Gaussian position kernel. k_p takes the input as normalized 2D position of pixels inside a patch and outputs the similarity scores which convey how a pixel of a patch is spatially close to all the other pixels of another patch. We have shown how 2D pixel positions are considered with the help of a 8×8 image patch in Fig. 1. In practice, 16×16 patches are used in our case. However, 2D pixel positions are not invariant to local rotation.

A texture image is made of patterns which are small visual elements. By repeated occurrence of these patterns, texture images are formed. It is often found that these patterns are affected by rotation. The patterns inside a texture image are expected to be oriented in a particular fashion. However, due to the rotation occurred, these patterns inside the texture image do not follow any particular orientation. In other words, we can say that the texture images are affected by local rotations. For this reason, it might cause a mismatch between two images even if they belong to the same class and consist of similar patterns. To elaborate it, we suppose, there are two texture images from same class and having similar patterns. However, patterns in the two images are in different orientations. Now, consider a 8×8 patch from both the images. Both the patches have similar patterns inside. However, the patterns inside both the patches are affected by rotation. Therefore, a pixel attribute at a particular position of one patch may not provide the optimum match with the pixel attribute at the corresponding position of another patch when 2D pixel positions are considered. We overcome this limitation by using rotation invariant (RI) pixel positions given by Fig. 2 where pixel positions are shown based on a 8×8 patch. As the rotation generally occurs based on a centre point, therefore, all the pixels are indexed with the same position around the centre of a patch in a rectangular ring fashion in our proposed RI-pixel positions approach. So, when the local rotation occurs and pixel attributes move to another location, their position will stay the same and the matching score between two patches will be invariant to the rotation.

When RI-pixel positions are incorporated in the position kernel which is a candidate kernel of all the three proposed match kernels in this paper, there is a change which occurs in the descriptors dimensions as well as in the complexity of descriptor extraction. As the RI-pixel positions are considered as 1D data, we chose the size of basis vectors on k_p as 5. Whereas, in case of 2D pixel positions, the size of basis vectors chosen on k_p is 5×5. Therefore using RI-pixel positions, the dimensions of $\tilde{F}_{coarse}$ becomes 25 ($5 \times 5 = 25$) and the dimensions of both $\tilde{F}_{dir}$ and $\tilde{F}_{contrast}$ becomes 50 ($10 \times 5 = 50$). The time complexity to extract the proposed descriptors is $O(nd^2 + nd_p^2 + n^2 dd_p)$, n is the number of pixels inside a patch. d represents the basis vector size on any of k_s, k_ψ or k_{con}. d_p represents the basis vector size on k_p. While using RI-pixel positions, d_p is significantly lower than what it is in case of 2D pixel positions. Therefore, using RI-pixel positions, decreases the time complexity of descriptor extraction phase compared to the case where 2D pixel positions are used.

IV. Experiment and Results

In this section, we will discuss the experiment details and results obtained. All the proposed descriptors are extracted over grayscale images using a patch size of 16×16 pixels with a spacing of 8 pixels. Kernel parameters involved in individual match kernels are chosen as $\gamma_s = 1$, $\gamma_\psi = 5$, $\gamma_{con} = 1$ and $\gamma_p = 3$. We have shown the performances of $\tilde{F}_{coarse}$, $\tilde{F}_{dir}$ and $\tilde{F}_{contrast}$ together as TamuraKDES on both texture and natural images. We consider the performance of TamuraKDES instead of individual descriptors, as we are improving conventional Tamura features (Coarseness, Directionality and Contrast)

which are generally used together to represent images. At first, image-level features of individual descriptors are extracted using efficient match kernel (EMK) [19] approach with 1000 visual words and three-level spatial pyramid grids [20]. To obtain TamuraKDES, image-level features of individual proposed descriptors are concatenated. For rotation invariant texture classification, we have used Outex_TC12_000 [21] as Test database 1. For real image classification, we have considered natural scene images from [22] as Test database 2. For Test database 3, we have used some classes from Caltech 101 database [23].

In this paper our goal is to show the effectiveness of Tamura features when it is being incorporated in the kernel descriptors framework. We do not aim to compare the performance of proposed descriptors in this paper with the conventional kernel descriptors [14]. Therefore, we compare the performance of our proposed descriptors with some existing works to achieve two specific goals: (a) to test the rotation invariance of proposed descriptors, Test database 1 is used on which the performances of TamuraKDES, conventional Tamura features along with a relevant literature are compared. (b) to test how effective the proposed descriptors are in representing the real images, we have used Test databases 2 and 3 and compared the performance of TamuraKDES with the performances of conventional Tamura features and some relevant literature where the texture features are used for real image classification. While extracting proposed descriptors, we have considered both 2D pixel positions and RI-pixel positions in k_p. However, from the experiment, it is observed that incorporating RI-pixel positions has achieved almost 2% higher accuracy in case of texture images having local rotation involved. Whereas, in the case of real images, using 2D pixel positions has achieved 1-1.5% higher accuracy compared to using RI-pixel positions. This is because, when using RI-pixel positions, all the pixels fall in same rectangular ring have same position index which is suitable to overcome local rotation. However, 2D pixel positions provide a unique position index to each pixel inside a patch and it is extracting more discriminative information in case of real images. Therefore, we have shown the results for Test database 1 when RI-pixel positions are considered for descriptor extraction. For the Test databases 2 and 3, we have shown the results when 2D pixel positions are considered for descriptor extraction.

Test database 1: Outex_TC12_000 consists of 24 different classes of texture images which are captured under three illuminations and nine rotation angles. The illuminations are $inca$, $tl84$ and $horizon$. The rotation angles are $0, 5, 10, 15, 30, 45, 60, 75$ and 90 degrees. Individual images are of the size 128×128 pixels. For each rotation angle, there are 20 images under a specific illumination condition. 20 images of $inca$ illumination and 0 degree angle, in total 480 images are used for training the classifier. To test the classifier, all the images captured under $tl84$ and $horizon$ illumination condition, i.e, 4320 images are used. A set of sample images from this database is shown in Fig. 3. We have used the performance of BRINT descriptor [24] to compare and show the effectiveness of TamuraKDES. BRINT descriptors are used for both rotation invariance and noise tolerance. However, we have only considered the rotation invariance case. Additionally, BRINT is already compared with all the existing popular methods for effectiveness in rotation invariance. So, by comparing the performance of TamuraKDES with BRINT, we can establish the effectiveness of TamuraKDES compared to the existing popular methods. For better comparison, experiment settings are kept same as in [24]. The work is carried out in Matlab interface and LIBSVM [25] is used as the classification tool. Results on this database are given in Table I where we can see that TamuraKDES outperforms the other two methods.

TABLE I: Comparison of image classification accuracy on Test database 1

Descriptor	Accuracy(%)
Conventional Tamura	29.85
BRINT [24]	98.13
TamuraKDES	**99.64**

TABLE II: Comparison of image classification accuracy on Test database 2

Descriptor	Accuracy(%)
Conventional Tamura	54.34
Human inspired feature [26]	91.93
TamuraKDES	**94.22**

Test database 2: Test database 2 is the natural scene images from [22]. In total, it consists of 1472 images which belong to four classes: (a) Coast, (b) Forest, (c) Mountain and (d) Open country, with a total number of 1472 images. A set of sample images is shown in Fig. 4. As Tamura features are human inspired, to investigate the effectiveness of TamuraKDES on this database, we have taken the reference of [26] where a set of human inspired features (including conventional Tamura features) are used to classify the images. For better comparison, we have kept the same experiment settings as in [26]. To test the classification accuracy, a feed-forward neural network is used with hidden and output layer consist of 120 and 4 neurons respectively. The entire database is tested with 20-fold cross-validation. Features are extracted over Matlab interface and for neural network classification, Scikit-learn toolbox [27] is used over Python environment. A comparison of classification accuracies on this database is given in Table II. From the results, we can conclude that TamuraKDES performs better compared to the other features to classify images in Test database 2.

Test database 3: Test database 3 consists of three classes from Caltech 101 database [23]. The three classes are: (a) Airplanes, (b) Faces and (c) Motorbikes. In total, the database consists of 2033 images. A set of sample images from the database is shown in Fig. 5. To compare the effectiveness of TamuraKDES on this database, we have considered [12] as the reference point. In [12], conventional Tamura features in the form of histogram and a combination of features including Tamura and invariant feature histograms are used to evaluate

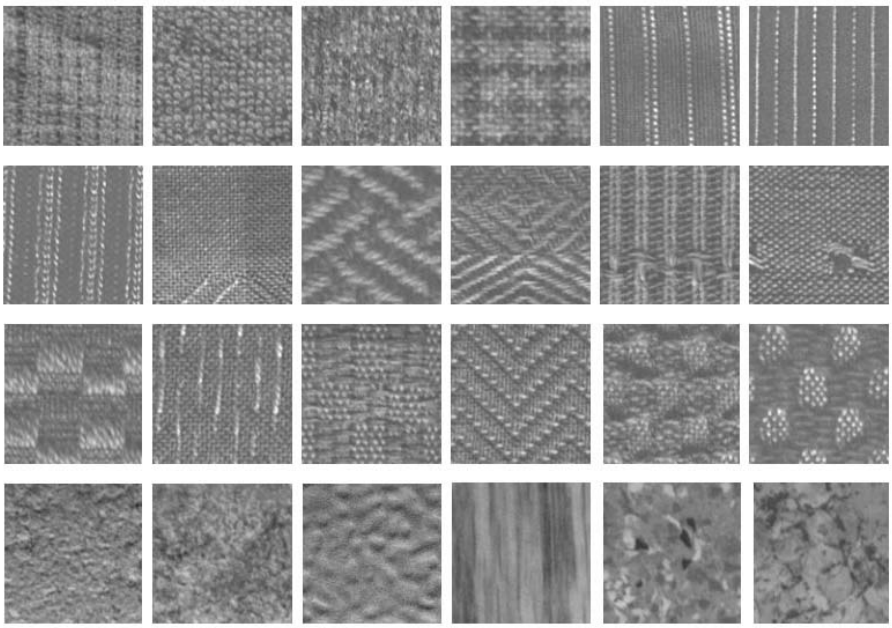

Fig. 3: Sample images from Test database 1

Fig. 4: Sample images from Test database 2

this database. To obtain the performance, nearest neighbour (NN) classifier is used to calculate the error-rate (Err) as, $Err = 1 - P(1)$, where $P(1)$ is the average precision for the first rank results over all the queries involved. In our experiment, we have used the same settings as in [12]. Features are extracted in Matlab interface and nearest neighbor classification is implemented using Scikit-learn toolbox over Python environment. Table III shows the comparison of the performances on this database and TamuraKDES is clearly the winner as the least error-rates are obtained using TamuraKDES.

TABLE III: Comparison of error-rates (%) on Test database 3

Descriptor	Airplanes	Faces	Motorbikes
Conventional Tamura	1.6	3.9	7.4
Combination of features [12]	0.8	1.6	8.5
TamuraKDES	**0.2**	**0.8**	**4.5**

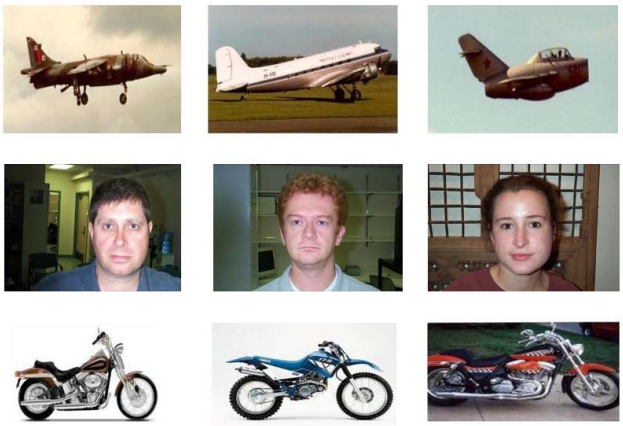

Fig. 5: Sample images from Test database 3

V. Conclusion

In this paper improved Tamura features are proposed using kernel descriptor framework. In the literature, most of the improvements on Tamura features are made based on histogram based approach. To construct histogram features, per pixel based Tamura features are quantized which may cause a significant amount of information loss. Whereas, in kernel descriptor approach, Tamura features per pixel equally participates in matching between two patches. Therefore, our proposed descriptors outperform conventional Tamura features significantly.

REFERENCES

[1] B. S. Manjunath, J.-R. Ohm, V. V. Vasudevan, and A. Yamada, "Color and texture descriptors," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 6, pp. 703–715, 2001.

[2] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[3] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 886–893.

[4] H. Tamura, S. Mori, and T. Yamawaki, "Textural features corresponding to visual perception," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 8, no. 6, pp. 460–473, 1978.

[5] V. Castelli and L. D. Bergman, "Digital imagery: fundamentals," *Image Databases: Search and Retrieval of Digital Imagery*, pp. 1–10, 2002.

[6] W.-Y. Ma and H. J. Zhang, "Benchmarking of image features for content-based retrieval," in *Signals, Systems & Computers, 1998. Conference Record of the Thirty-Second Asilomar Conference on*, vol. 1. IEEE, 1998, pp. 253–257.

[7] T. Majtner and D. Svoboda, "Extension of tamura texture features for 3d fluorescence microscopy," in *3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT), 2012 Second International Conference on*. IEEE, 2012, pp. 301–307.

[8] M. M. Islam, D. Zhang, and G. Lu, "A geometric method to compute directionality features for texture images," in *Multimedia and Expo, 2008 IEEE International Conference on*. IEEE, 2008, pp. 1521–1524.

[9] Y. Liu, Z. Li, and Z.-M. Gao, "An improved texture feature extraction method for tyre tread patterns," in *International Conference on Intelligent Science and Big Data Engineering*. Springer, 2013, pp. 705–713.

[10] M. M. Ali, M. B. Fayek, and E. E. Hemayed, "Human-inspired features for natural scene classification," *Pattern Recognition Letters*, vol. 34, no. 13, pp. 1525–1530, 2013.

[11] P. Howarth and S. Rüger, "Robust texture features for still-image retrieval," *IEE Proceedings-Vision, Image and Signal Processing*, vol. 152, no. 6, pp. 868–874, 2005.

[12] T. Deselaers, D. Keysers, and H. Ney, "Features for image retrieval: an experimental comparison," *Information Retrieval*, vol. 11, no. 2, pp. 77–107, 2008.

[13] T. Majtner and D. Svoboda, "Extension of tamura texture features for 3d fluorescence microscopy," in *3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT), 2012 Second International Conference on*. IEEE, 2012, pp. 301–307.

[14] L. Bo, X. Ren, and D. Fox, "Kernel descriptors for visual recognition," in *Advances in Neural Information Processing Systems*, 2010, pp. 244–252.

[15] ——, "Depth kernel descriptors for object recognition," in *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*. IEEE, 2011, pp. 821–826.

[16] X. Ren, L. Bo, and D. Fox, "Rgb-(d) scene labeling: Features and algorithms," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 2759–2766.

[17] T.-H. Tran and V.-T. Nguyen, "How good is kernel descriptor on depth motion map for action recognition," in *International Conference on Computer Vision Systems*. Springer, 2015, pp. 137–146.

[18] D. Hu, L. Bo, and X. Ren, "Toward robust material recognition for everyday objects." in *BMVC*, vol. 2. Citeseer, 2011, p. 6.

[19] L. Bo and C. Sminchisescu, "Efficient match kernel between sets of features for visual recognition," in *Advances in neural information processing systems*, 2009, pp. 135–143.

[20] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Computer vision and pattern recognition, 2006 IEEE Computer Society Conference on*, vol. 2. IEEE, 2006, pp. 2169–2178.

[21] 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.

[22] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International Journal of Computer Vision*, vol. 42, no. 3, pp. 145–175, 2001.

[23] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories," *Computer Vision and Image Understanding*, vol. 106, no. 1, pp. 59–70, 2007.

[24] L. Liu, Y. Long, P. W. Fieguth, S. Lao, and G. Zhao, "Brint: binary rotation invariant and noise tolerant texture classification," *IEEE Transactions on Image Processing*, vol. 23, no. 7, pp. 3071–3084, 2014.

[25] C.-C. Chang and C.-J. Lin, "Libsvm: a library for support vector machines," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, no. 3, p. 27, 2011.

[26] M. M. Ali, M. B. Fayek, and E. E. Hemayed, "Human-inspired features for natural scene classification," *Pattern Recognition Letters*, vol. 34, no. 13, pp. 1525–1530, 2013.

[27] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, "Scikit-learn: Machine learning in python," *Journal of Machine Learning Research*, vol. 12, no. Oct, pp. 2825–2830, 2011.