Noname manuscript No.
(will be inserted by the editor)

A Novel Fusion Approach in the Extraction of Kernel Descriptor with Improved Effectiveness and Efficiency

Priyabrata Karmakar · Shyh Wei Teng · Guojun Lu · Dengsheng Zhang

Received: date / Accepted: date

Abstract Image representation using feature descriptors is crucial. A number of histogram-based descriptors are widely used for this purpose. However, histogram-based descriptors have certain limitations and kernel descriptors (KDES) are proven to overcome them. Moreover, the combination of more than one KDES performs better than an individual KDES. Conventionally, KDES fusion is performed by concatenating them after the gradient, colour and shape descriptors have been extracted. This approach has limitations in regard to the efficiency as well as the effectiveness. In this paper, we propose a novel approach to fuse different image features before the descriptor extraction, resulting in a compact descriptor which is efficient and effective. In addition, we have investigated the effect on the proposed descriptor when texture-based features are fused along with the conventionally used features. Our proposed descriptor is examined on two publicly available image databases and shown to provide outstanding performances.

Keywords Kernel descriptor · Tamura features · Descriptor fusion · Image classification · Image retrieval

1 Introduction

Effective and efficient image representation is essential in image processing applications, such as image classification and retrieval. With the increase in advanced camera technologies and cheaper storage devices, a lot of images are captured everyday. To process this huge amount of images by representing them effectively and efficiently is a challenging task. Images are commonly represented by various feature descriptors which capture the pixel information and represent images in a compact form. Image descriptors are broadly classified into global and local descriptors. The former one describes an image as a whole. In contrast, the latter

Piyabrata Karmakar
E-mail: p.karmakar@federation.edu.au
School of Engineering, IT and Physical Sciences, Federation University Australia, Gippsland Campus, Churchill, VIC-3842

describes small or local regions of images. Local descriptors are usually more robust and discriminative than the global ones and therefore, they are popular and widely used [45, 14].

Over the years, many local descriptors have been proposed. Among them, histogram-based descriptors, such as scale invariant feature transform (SIFT) [21], histogram of oriented gradients (HOG) [7], Fuzzy colour histogram [18, 6] are popular. Although they are easy to use, these descriptors suffer from information loss due to the coarse quantization of pixel attributes. To overcome these limitations, a set of kernel descriptors (KDES)[3] were proposed. KDES are specifically designed that no quantization is performed on the pixel attributes. Instead, each pixel equally participates in the matching between two image patches. Thus, kernel descriptors provide more distinct and accurate information compared to the histogram-based descriptors.

In addition, combination of descriptors instead of a single descriptor is more effective approach to represent images. It has been shown that the performance of combining all KDES is higher than the highest performing individual kernel descriptor [3]. However, the conventional fusion approach as per [3] has limitations with respect to the effectiveness, efficiency and storage. To overcome these issues, in this paper we propose a novel fusion approach to extract kernel descriptor. Specifically, our contribution in this paper is two-fold:

1. Fuse different image features (e.g. gradient, colour and shape) together before the descriptor extraction stage and compare the performance with the conventional approach in terms of effectiveness and efficiency.
2. Investigate the performance when additional texture-based features are incorporated in the proposed fusion model along with the conventionally used features (i.e. gradient, colour and shape).

Recently, different deep learning architectures have shown the higher effectiveness in various image processing applications compared to the traditional feature descriptors (e.g. histogram-based descriptors, kernel descriptors) [40, 11]. However, the traditional feature descriptors have their own advantages. First of all, the deep learning architectures are often huge with many layers of neurons with millions of parameters and require large set of training images for the appropriate tuning and weight adjustments. In contrast, traditional feature descriptors can perform satisfactorily with a small number of training images and fewer parameters need to be tuned [22, 46]. In addition, deep learning is generally treated as a black-box and therefore, it is still not clear what visual information is being represented in the complex features derived from a deep learning method. In some application domains, e.g. in crime investigation and presentation of evidence in the court, it is essential to explain how a computer algorithm or method derives its results. Traditional feature descriptors are easier to explain and visualize [15]. Due to the aforementioned reasons, further research and development of traditional feature descriptors are still important and essential. Therefore, in this paper, our focus is to improve an existing feature descriptor framework in terms of effectiveness and efficiency.

The rest of paper is organized as follows. Section 2 discusses the motivation behind our work followed by related works in Section 3. Proposed fusion approach is explained in Section 4. Section 5 demonstrates descriptor dimensionality. The

details of the experiments and results are given in Section 6. Finally, Section 7 concludes the paper.

2 Motivation

In this section we discuss our motivation behind the proposed descriptor. At first, we describe the limitations of histogram-based descriptors followed by how KDES overcome them. After that, we discuss the necessity of descriptor fusion and finally, we introduce the approach to extract the proposed kernel descriptor.

During the extraction of histogram-based descriptors, pixel attributes are approximated or quantized to a pre-defined value or range which corresponds to a bin of the histogram. Such a representation encounters two limitations as follows.

1. Histogram-based descriptors consist of several bins. The number of histogram bins are user-defined and the magnitude of each bins are obtained by approximating individual pixel attributes using a pre-defined rule. The most common approach of approximation is the division of the entire range of pixel attributes into small intervals which correspond to the individual bins of a histogram. A pixel attribute is approximated to a particular bin if it belongs to the corresponding interval. While approximating pixel attributes, it may happen that perceptually very similar pixels are falling into different bins and less similar pixels are falling into the same bin. Therefore, it leads to less informative image representation.
2. The pixel attributes of local image regions are approximated to fixed dimension histograms, but the structural or spatial information of the regions is ignored. Based on the concept of histogram-based descriptors, two spatially dissimilar image regions with similar pixel attributes may form similar descriptors. Therefore, these descriptors are not as effective as it could be with the spatial information of pixel attributes incorporated.

Due to the two above limitations, the image representation with histogram-based descriptors is less discriminative. To overcome this, the authors of [3] have proposed a set of KDES. In the KDES framework, each pixel equally participates in the process of similarity measurement between two images. Thus, the information loss due to coarse quantization is minimized. In addition, in the KDES framework, along with the pixel attributes, pixel positions are also considered while measuring the similarities between images. Therefore, the spatial and structural information of pixels in an image region is preserved.

Descriptor fusion is a standard approach to increase the effectiveness in image representation. Individual descriptors capture one type of features and may not be sufficient to represent images effectively. Therefore, to represent images with multiple features simultaneously, descriptor fusion is required. In addition, image representation can be made rotation-, translation- and occlusion-invariant at the same time by fusing individual descriptors which are rotation-, translation- and occlusion-invariant respectively. There are different fusion approaches exist in the literature. For example, in [38,39,20], multiple kernel learning frameworks are proposed for fusing the descriptors by a weighted summation of kernel matrices computed over individual candidate descriptors. To fuse different features, covariance matrices are also used for representing images effectively in [36,37,31].

Serial fusion or the concatenation of diffrernt candidate descriptors into a single robust descriptor is also a popular approach [30]. The descriptor fusion in [3] and in the subsequent works is performed by the concatenation of image-level descriptors of individual KDES. Although this conventional fusion approach shows higher effectiveness, it still has several limitations. First of all, it is not an efficient process. Secondly, It needs huge amount of storage. Finally, there is a risk of information loss. These limitations are discussed in Section 3.2. To overcome these limitations of conventional fusion, in this paper, we have proposed a novel approach where image features (i.e. gradient, colour and shape) are fused before the descriptor extraction stage, resulting in a more efficient and effective descriptor.

Conventionally, KDES are based on gradient, colour and shape features. However, the KDES framework can turn any kind of pixel attributes into a patch-based descriptor [44, 35, 13]. Tamura features represent important texture information. Originally [33], Tamura features are designed as global descriptors and a total of six features have been proposed: 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 in distinguishing different textures. To leverage the properties of Tamura features in representing local image regions, authors of [17] have proposed a set of texture-based kernel descriptors based on per-pixel Tamura features. In this paper, we have also investigated the performance of the proposed descriptor when per-pixel Tamura features are incorporated along with the conventionally used features into the proposed fusion model.

3 Related Work

In this section, we discuss the basic idea of KDES extraction based on a match kernel, followed by description of a set of relevant literature related to the KDES. Thereafter, the conventional approach to fuse individual KDES is discussed.

3.1 Kernel Descriptors

In this section, the concept behind the extraction of KDES based on match kernels is briefly explained using gradient kernel descriptors (GKDES). The match kernel based on gradient is given by (1).

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

where A and B are two different image patches. $\tilde{m}(z)$ represents the normalized gradient magnitudes of pixels in Patch A. $\tilde{m}(z)\tilde{m}(z')$, a linear kernel which can also be represented as $k_{\tilde{m}}(z, z')$ 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. To estimate the difference between orientations at pixels z and z', the authors of [3] computed k_o with normalized gradient vectors which are basically 2-D data containing x- and y-directional gradients of individual pixels. $k_p(z, z') = \exp(-\gamma_p\|z - z'\|^2)$ is a

Gaussian kernel over the 2-D position of pixels inside a patch and z (or z') denotes the 2D position. k_p conveys how close two pixels are spatially.

Gradient kernel descriptor (GKDES) is based on K_{grad} which consists of three candidate kernels: k_m, k_o and k_p. In [3], GKDES is extracted using the feature maps of these three candidate kernels and it is given by (2).

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

where $\phi_{\tilde{m}}(.)$, $\phi_o(.)$ and $\phi_p(.)$ are the feature maps of $k_{\tilde{m}}$, k_o and k_p respectively. $\otimes$ represents Kronecker product.

F_{grad} is called the gradient kernel descriptor as it is derived from the gradient match kernel. Being a linear kernel, the feature map of $k_{\tilde{m}}$ is the normalized gradient magnitudes only, i.e. $\phi_{\tilde{m}}(z) = \tilde{m}(z)$. For the other two candidate kernels, feature maps cannot be extracted directly as they are non-linear (Gaussian) kernels. Therefore, the feature maps $\phi_o(.)$ and $\phi_p(.)$ are approximated using a set of basis vectors. GKDES extraction in [3] results in a very high-dimensional descriptor and kernel principal component analysis (KPCA) is applied to reduce the dimensionality. Similarly, colour and shape kernel descriptors can also be extracted.

After the proposal in [3], many researchers have improved the conventional KDES in terms of effectiveness and efficiency. Moreover, due to its simplicity, KDES have been successfully applied to different applications of image processing and computer vision. To build image-level descriptors from the local descriptors, hierarchical kernel descriptors (HKDES) are proposed in [2]. HKDES is extracted from the match kernels defined over the patch-based KDES and spatial information of image patches is incorporated within the extracted descriptors. To integrate the image label information and to extract lower dimensional descriptors, supervised kernel descriptors (SKDES) are proposed in [41]. Another supervised approach, supervised efficient KDES (SEKD) is proposed in [43]. Instead of using joint basis vectors, authors of [42] extracted efficient KDES (EKD) by automatically selecting a small number of pivot features. Context kernel descriptors (CKD) [26] which provide increased robustness are proposed by incorporating spatial context during the descriptor extraction stage. In [4], a set of kernel descriptors are designed on depth images to represent size, 3D shape and depth edge more effectively. In [28], depth kernel descriptors are used for indoor scene labelling of RGB-depth images by aggregating kernel descriptors over super-pixels.

In [3] and the subsequent modifications on it, the non-linear candidate kernels take multi-dimensional data as input. In other words, the candidate kernels are the Hadamard product of component kernels computed over 1-D data. Therefore, if any image irregularity or noise exists, its effect becomes multiplicative and degrades the effectiveness of the overall performance. In addition, the descriptors are originally extracted as high-dimensional and the dimensionality reduction process increases the computational complexity [12] as well as potential of losing discriminative information. To overcome these limitations, authors of [16] have proposed improved kernel descriptors by leveraging the kernel properties. Specifically, the candidate kernels are represented as the summation of component kernels which are built using 1-D data. For example in [3], k_o which is a candidate kernel of the gradient match kernel takes a two-dimensional input of gradient vectors. In contrast, the corresponding candidate kernel $k_o^{'}$ in [16] is the summation of k_{ox} and

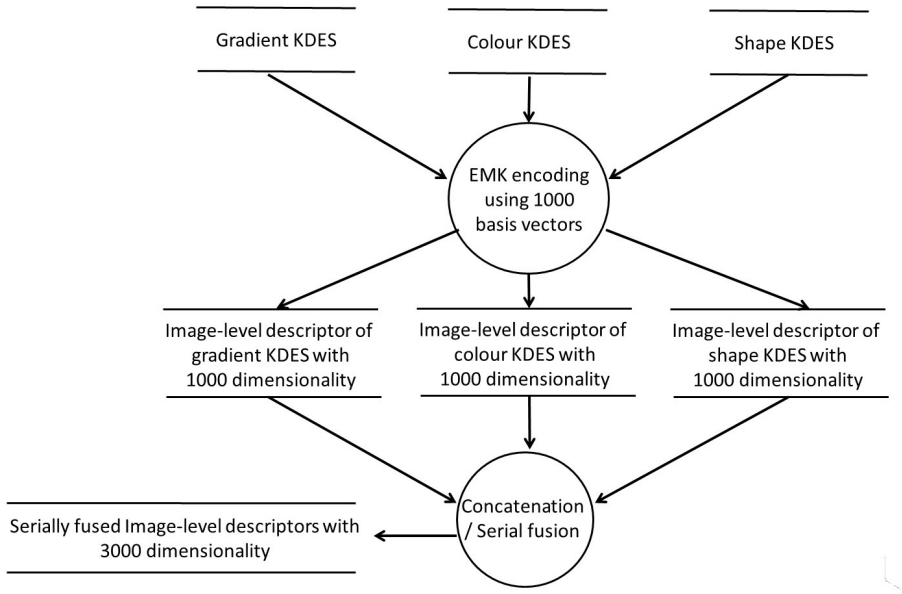

Fig. 1 Conventional fusion approach of kernel descriptors in [3]

k_{oy} which are the component kernels built using normalized x- and y- directional gradients respectively, such that $k_o^{'} = k_{ox} + k_{oy}$. In this way, all candidate kernels for each of the match kernels are modified and the resulting descriptors are extracted as lower dimensional and lesser noise-sensitive. The descriptors in [16] are originally extracted as low-dimensional. So, there is no dimensionality reduction needed and the corresponding risk of information-loss and higher computational complexity are avoided.

3.2 Conventional Fusion of KDES and its Limitations

Individual KDES are based on a particular pixel attribute or an image feature. They are more effective compared to their histogram-based counterparts. The combination of descriptors instead of a single descriptor is even more effective to represent images. In [3] authors have shown that the combined performance of all three KDES is higher than the highest performing individual kernel descriptor. We refer this fusion approach as the conventional fusion. It is shown by a flow diagram in Figure 1. The same fusion approach is also considered in the subsequent modifications on KDES. The conventional fusion approach has limitations with respect to the efficiency, effectiveness and storage as follows.

1. Conventional fusion in [3] and in the subsequent literature is achieved by concatenating the image-level descriptors of gradient, colour and shape-based kernel descriptors (i.e. serial fusion). This process includes local descriptor extraction, dimensionality reduction and image-level descriptor encoding of each KDES. This is a computationally expensive process.

2. Due to the dimensionality reduction, individual KDES loses some information and therefore, the effectiveness of the conventionally fused descriptor is also affected.
3. Concatenation of image-level descriptors lead to a high dimensional representation which needs higher memory requirements and it is highly time-consuming to process.

4 Proposed Fusion Approach

To address the above mentioned limitations, a novel fusion approach is proposed in this section. The proposed approach will address the above limitations in the following aspects.

1. In the proposed approach, different image features (i.e. gradient, colour, shape) are fused before the descriptor extraction. Hence, a unique kernel descriptor is extracted which contains all the distinct information of different pixel attributes. To construct image-level descriptors, image-level encoding is performed only once. For this reason, time complexity is reduced by almost one-third compared to what is in [3].
2. There is no dimensionality reduction required in the extraction of the proposed descriptor. Therefore, the chance of information loss is lower. Thus, the proposed descriptor is more effective compared to the conventionally fused descriptors.
3. Only one descriptor is extracted. Therefore, there is no need of image-level descriptor concatenation which is needed for the serial fusion. Thus, each image in the proposed scenario is represented with a lower dimensional descriptor compared to the conventionally fused descriptors. Due to lower dimensionality, memory requirement is lower and the processing of each image in the proposed scenario is faster compared to what it is in [3].

The proposed approach can fuse any kind and any number of pixel attributes together. At first, we have fused three conventionally used features (i.e. gradient, colour and shape) to make a fair comparison with the baseline [3]. For that, we have designed fused match kernel (FMK) and the resulting descriptor is named as fused kernel descriptor (FKD). After that, we have investigated the performance when texture-based features (i.e. per-pixel Tamura features) are also integrated into the fusion model. Here, we have only considered first three Tamura features, i.e. coarseness, directionality and contrast. In this scenario, we named the descriptor as extended fused kernel descriptor (EFKD) extracted from the extended fused match kernel (EFMK).

4.1 Formation of Fused Match Kernel

Kernel descriptors are based on a match kernel that are computed using individual features (e.g.pixel attributes) over a dense patch (in [3], it is 16×16 pixels patch with 8 pixels of spacing) using a Gaussian function. For example, gradient kernel descriptor is extracted from the gradient match kernel where the main component is a candidate kernel computed using gradient vectors. By following the same

approach, if we can design a candidate kernel that captures multiple features from a dense patch, then the descriptor extracted from the corresponding match kernel will contain information related to all the features considered. Therefore, we have designed a kernel K_M which captures gradient, colour and shape features. Based on [32], the summation of valid kernels produces another valid kernel. We computed K_M as the summation of kernels computed using different features. Thereafter, a unique match kernel named as fused match kernel (FMK) is proposed using K_M as follows.

$$K_{Fused} = \sum_{z \in A} \sum_{z' \in B} K_M(z, z') k_p^{'}(z, z'), \tag{3}$$

where A and B are the image patches from two different images. z and z' represent the pixels of A and B patches respectively. $K_M(z, z') = K_G(z, z') + K_C(z, z') + K_S(z, z')$. By summing up K_G, K_C and K_S, the similarity scores obtained by them are fused in K_M. $k_p^{'}(z, z') = k_{px}(z, z') + k_{py}(z, z')$ measures the spatial proximity of pixels inside Patches A and B. $k_{px}(z, z') = \exp(-\gamma_p \|z_x - z'_x\|^2)$ is a Gaussian kernel over x position of pixels inside a patch and z_x (or z'_x) denotes the x pixel position. Similarly, k_{py} is the Gaussian kernel over y position of pixels.

Match kernels corresponding to individual KDES are composed mainly of two components. The first component is related to the corresponding pixel attribute. The second component is same for each case and it is related to the pixel positions. K_M used in (3) is the summation of K_G, K_C and K_S which are the unique components belonging to the gradient, colour and shape features respectively. The definitions of K_G, K_C and K_S are provided by (4), (5) and (6) respectively.

$$K_G(z, z') = \tilde{m}(z)\tilde{m}(z') k_o^{'}(\tilde{\theta}(z), \tilde{\theta}(z')), \tag{4}$$

where K_G measures the similarity between patches A and B in terms of pixel gradients. $\tilde{m}(z)$ and $\tilde{m}(z')$ are the normalized gradient magnitudes of Patches A and B respectively. $k_o^{'}(\tilde{\theta}(z), \tilde{\theta}(z')) = k_{ox}^{'}(\tilde{\theta}_x(z), \tilde{\theta}_x(z')) + k_{oy}^{'}(\tilde{\theta}_y(z), \tilde{\theta}_y(z'))$. $k_{ox}(\tilde{\theta}_x(z), \tilde{\theta}_x(z')) = \exp(-\gamma_o \|\tilde{\theta}_x(z) - \tilde{\theta}_x(z')\|^2)$ is a Gaussian kernel over normalized x- directional gradients. Similarly, k_{oy} is the Gaussian kernel over normalized y- directional gradients.

$$K_C(z, z') = k_{cR}(z_R, z'_R) + k_{cG}(z_G, z'_G) + k_{cB}(z_B, z'_B), \tag{5}$$

where K_C measures the colour similarity between patches A and B. $k_{cR}(z_R, z'_R) = \exp(-\gamma_c \|c(z_R) - c(z'_R)\|^2)$ is a Gaussian kernel over pixel intensity at red channel. Similarly, k_{cG} and k_{cB} are the Gaussian kernels over pixel intensities at green and blue channels respectively. If the image is grayscale, then K_C is the Gaussian kernel over pixel intensities.

$$K_S(z, z') = \tilde{s}(z)\tilde{s}(z') k_b^{'}(b(z), b(z')), \tag{6}$$

where K_S measures how two patches A and B are similar in terms of shape. $\tilde{s}(z)$ and $\tilde{s}(z')$ are the normalized standard deviations of pixel values in the 3×3 neighbourhood around each pixel (z or z') inside patches A and B respectively. $b(z)$ is a binary vector which is the local binary pattern (LBP) [25] of z around 3×3 neighbourhood. $k_b'(b(z), b(z')) = k_{b1}(b_1(z), b_1(z')) + \cdots + k_{b8}(b_8(z), b_8(z'))$. $k_{b1}, \cdots, k_{b8}$ are the Gaussian kernels computed over individual dimensions of 8-D LBP.

After designing FMK, we have extended it by incorporating per-pixel Tamura features [17]. Thus, resulting extended fused match kernel (EFMK) and it is given by 7.

$$K_{EFused}(A,B) = \sum_{z\in A}\sum_{z'\in B} K_{EM}(z,z')k_p^{'}(z,z'), \tag{7}$$

where $K_{EM}(z,z') = K_M(z,z')+K_T(z.z')$ and $K_T(z,z') = k_s(s(z),s(z'))+k_\psi(\psi(z),\psi(z'))+k_{con}(con(z),con(z'))$. K_T sums up the similarity scores obtained using K_s, K_ψ and K_{con} given by 8, 9 and 10 which are the Gaussian kernels computed over per-pixel coarseness, directionality and contrast respectively.

$$K_s(s(z),s(z')) = \exp(-\gamma_s\|s(z)-s(z')\|^2), \tag{8}$$

where $s(z)$ represents the S_{best} value of a pixel z. K_s finds the similarity between S_{best} values. S_{best} values for individual pixels are obtained by considering the same approach as in [33].

$$K_\psi(\psi(z),\psi(z')) = \exp(-\gamma_\psi\|\psi(z)-\psi(z')\|^2), \tag{9}$$

where K_ψ finds the similarity between the normalized (within [0,1]) ψ values of pixels z and z'. $\psi(z) = \theta(z)$ for the corresponding $|\Delta G|> t$, where t is fixed as '12' as per [33]. Otherwise $\psi(z) = \varepsilon(\varepsilon\to 0)$. The edge of a pixel is a vector and it has both magnitude ($|\Delta G|$) and direction (θ) which are calculated as $|\Delta G| = \frac{|\Delta H|+|\Delta V|}{2}$, $\theta = tan^{-1}\frac{|\Delta V|}{|\Delta H|}+\frac{\pi}{2}$. ΔH and ΔV are obtained by applying Prewitt operator to the corresponding pixel. Thresholding $|\Delta G|$ by t helps to reject unreliable directions which cannot be considered as edge points.

$$K_{con}(con(z),con(z')) = \exp(-\gamma_{con}\|con(z)-con(z')\|^2), \tag{10}$$

where K_{con} finds the similarity between normalized (with [0,1]) contrast values at z and z'. Contrast value per pixel is calculated as $con(z) = (I-mn)/std$. I is the intensity of pixel z. mn and std are the mean and standard deviation of the pixel intensities around the 3×3 neighbourhood of the pixel z.

In the research area of feature extraction, the main kernels used in the literature are linear, polynomial and Gaussian. However, Gaussian kernel is the most suitable to represent the complex relationship between data [27]. Hence, it is highly popular among the researchers. The Gaussian kernel has been used in [3] and in the consequent research done on KDES. Therefore, in our work, Gaussian kernels are also used as the base kernels to match pixel attributes in the corresponding match kernels to make a fair comparison with the existing methods.

4.2 Descriptor Extraction from Fused Match Kernel

To extract the descriptor from FMK, the same approach that has been taken in [16] to extract descriptors from the match kernels is considered. At first, the candidate kernels of K_{Fused} are represented in terms of their inner products as, $K_M(z,z') = \phi_M(z)^T\phi_M(z')$ and $k_p^{'}(z,z') = \phi_p^{'}(z)^T\phi_p^{'}(z')$, where $\phi_M(.)$ and $\phi_p^{'}(.)$ are the feature maps of k_M and $k_p^{'}$ respectively. The descriptor extraction using the feature maps is given by (11).

$$F_{Fused}(A) = \sum_{z \in A} \phi_M(z) \otimes \phi_p^{'}(z), \tag{11}$$

K_M is the linear summation of three kernels K_G, K_C and K_S. Therefore, ϕ_M is the concatenation of feature maps of these three kernels [32] and it is given by (12).

$$\begin{aligned} K_M(z, z') &= K_G(z, z') + K_C(z, z') + K_S(z, z') \\ &= \phi_G(z)^T \phi_G(z') + \phi_C(z)^T \phi_C(z') + \phi_S(z)^T \phi_S(z') \\ &= \begin{bmatrix} \phi_G(z) \\ \phi_C(z) \\ \phi_S(z) \end{bmatrix}^T \begin{bmatrix} \phi_G(z') \\ \phi_C(z') \\ \phi_S(z') \end{bmatrix}, \end{aligned} \tag{12}$$

from (12), it can be easily concluded that $\phi_M(.) = \begin{bmatrix} \phi_G(.) \\ \phi_C(.) \\ \phi_S(.) \end{bmatrix} = \begin{bmatrix} \phi_G(.) & \phi_C(.) & \phi_S(.) \end{bmatrix}^T$.

The feature maps of non-linear (Gaussian) kernels cannot be extracted directly. Therefore, $\phi_G(.)$, $\phi_C(.)$, $\phi_S(.)$ and $\phi_p(.)$ are approximated using individual sets of basis vectors as per [3,16]. Thereafter, $\phi_M(.)$ is approximated as $\tilde{\phi}_M(.)$ by concatenating the approximated feature maps of K_G, K_C and K_S. Also, $\phi_p^{'}(.)$ is approximated as $\tilde{\phi_p'}(.)$. So, (11) is approximated and given by (13).

$$\tilde{F}_{Fused}(A) = \sum_{z \in A} \tilde{\phi}_M(z) \otimes \tilde{\phi_p'}(z). \tag{13}$$

The descriptor given by equation (13) is the fused kernel descriptor (FKD). A flow diagram to show how FKD is extracted and evaluated is given in Figure 2. The dimensionality of FKD is significantly lower compared to the original dimensions of individual KDES in the baseline [3]. Therefore, no dimensionality reduction is required for FKD. In contrast, to perform serial fusion in the baseline, dimensionality reduction is performed for each participating KDES and it is time-consuming process. In addition, to represent images using FKD, descriptor extraction and image-level encoding done only once compared to n times in the serial fusion (considering n different types of KDES take part in the corresponding serial fusion). Therefore, FKD is more efficient in representing images compared to the serial fusion in baseline.

The proposed fusion approach is not restricted to fuse only gradient, colour and shape features. Due to it's simplicity, any kind and any number of features can be fused together to obtain a unique kernel descriptor. Thus, we have also extracted a descriptor based on EFMK which is given by 7 and we named it extended fused kernel descriptor (EFKD). To extract EFKD from EFMK, the same approach of FKD extraction is considered. Feature maps of candidate kernels of EFMK are approximated using individual sets of basis vectors. Subsequently, using all the feature maps together, EFKD is extracted and it is given by 14.

$$F_{EFused}(A) = \sum_{z \in A} \phi_{EM}(z) \otimes \phi_p^{'}(z), \tag{14}$$

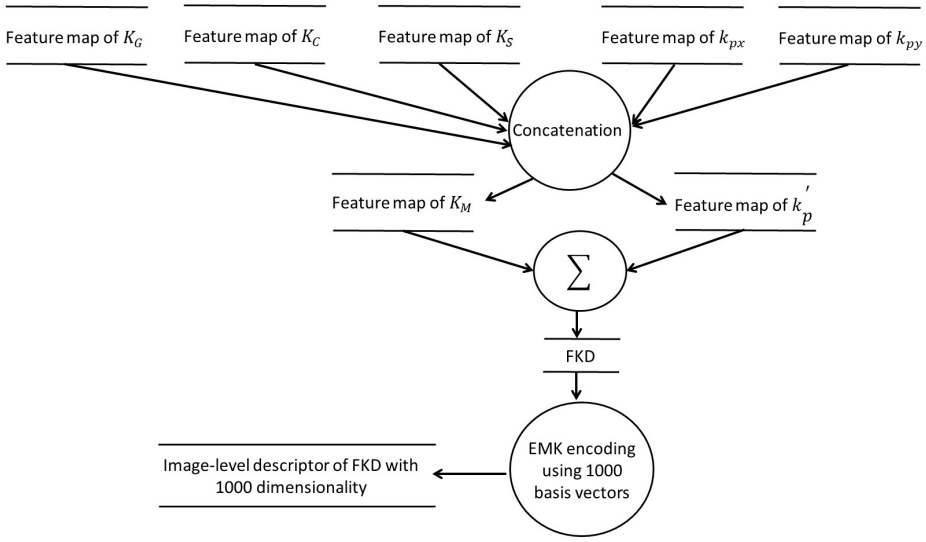

Fig. 2 Extraction and evaluation of FKD

where $\phi_{EM}(.) = \left[\begin{array}{cc} \phi_M(.) & \phi_T(.) \end{array} \right]^T$. K_{EM} being a non-linear kernel, $\phi_{EM}(.)$ is approximated as $\tilde{\phi}_{EM}(.)$ by concatenating the approximated feature maps of K_M and K_T. $\phi_T(.)$, the feature map of K_T is approximated as $\tilde{\phi}_T(.)$ by concatenating the approximated feature maps of K_S, K_ψ and K_{con}. Finally, (14) is approximated and given by (15).

$$\tilde{F}_{EFused}(A) = \sum_{z \in A} \tilde{\phi}_{EM}(z) \otimes \tilde{\phi}_p^{'}(z). \tag{15}$$

EFKD extracted using proposed fusion approach is also lower in dimensionality. Therefore, no dimensionality reduction needed. A detailed discussion on the descriptor dimensionality is provided in the following section.

5 Descriptor Dimensionality

In this section, the dimensionality associated with the conventional and the proposed fusion approaches will be compared for the local descriptor extraction as well as for the EMK encoding stage.

It is already known that KDES are extracted using the feature maps of candidate kernels of the corresponding match kernels. To approximate the feature maps, a set of basis vectors is needed. The basis vectors are constructed in the similar way as in [16]. The size of the basis vectors chosen on kernel k_o, k_c, k_b and k_p are 10×10, $5 \times 5 \times 5$, $2^8 = 256$ and 5×5 respectively. Therefore, the original dimensionality of gradient, colour and shape KDES used in conventional fusion [3] are $(10 \times 10) \times (5 \times 5) = 2500$, $(5 \times 5 \times 5) \times (5 \times 5) = 3125$ and $256 \times (5 \times 5) = 6400$ respectively. In contrast, the size of basis vectors on kernel $k_o{}'$ is $10 + 10 = 20$. The size of basis vectors on kernel $k_c^{'}$ is $5 + 5 + 5 = 15$ for RGB images or 5 for

Table 1 Comparison of dimensionalities between conventional fusion and FKD

Descriptor type	Conventional fusion	FKD
Local descriptor	Gradient- 2500 Colour- 3125 Shape - 6400	510 (RGB) 410 (Grayscale)
Image-level descriptor	3000	1000

grayscale images. The size of basis vectors on $k_b^{'}$ is $(2+2+2+2+2+2+2+2) = 16$ and the size of basis vectors on kernel $k_p{}'$ is $5 + 5 = 10$. Therefore, descriptor dimensionality of the proposed FKD is $(20 + 15 + 16) \times 10 = 510$ for RGB images or $(20 + 5 + 16) \times 10 = 410$ for grayscale images.

For the image-level representation, EMK encoding is done for the descriptors used in conventional fusion as well as for the FKD. For the EMK encoding of each local descriptor, the size of basis vectors chosen is 1000. i.e. the dimensionality of feature maps for each descriptor for EMK encoding is 1000. Therefore, the image-level descriptor dimensions of individual descriptors in the conventional fusion and FKD are 1000. As the conventional fusion approach is the concatenation of image-level descriptors of participating local descriptors. Therefore, the final dimensionality of image-level descriptors of each image using conventional fusion approach is 3000 (in the baseline [3], three local descriptors are concatenated). In contrast, the final dimensionality of image-level descriptors to represent an image using FKD is only 1000. For the better representation, dimensionality comparisons are given in Table 1 which states that irrespective of the local descriptor extraction or EMK encoding stage, FKD always deal with less dimensionality of descriptors compared to the conventional fusion approach.

Similarly, we have also computed the dimensionalities of EFKD extraction and encoding. All the feature maps required to extract FKD are also needed to extract EFKD. In addition, feature maps belonging to k_s, k_ψ, k_{con} are also needed. We have chosen the sizes of basis vector sets on the candidate kernels belonging to per-pixel Tamura coarseness (k_s), Tamura directionality (k_ψ) and Tamura contrast (k_{con}) are 5, 10 and 10 respectively. Therefore, the dimensionality of EFKD is $(20 + 15 + 16 + 5 + 10 + 10) \times (5 + 5) = 760$ for RGB images or $(20 + 5 + 16 + 5 + 10 + 10) \times (5 + 5) = 660$ for grayscale images. The dimensionality of EFKD is within the accepted range and efficient to process. Therefore, there is no need for dimensionality reduction. In this case also, the size of basis vectors chosen for EMK encoding of EFKD is 1000. Therefore, the image-level descriptor dimension belonging to EFKD is 1000.

6 Experiments and Results

In this section, the experimental details and the results are provided. At first, the test databases which are used to evaluate the classification and retrieval performances are discussed. After that, experiment settings and computation time are provided. Image classification and retrieval results are discussed thereafter, followed by a detailed qualitative and quantitative analysis.

Fig. 3 Sample images from Scene categories database

6.1 Test Databases

For the experiments, the following test databases are used. The motivation is to compare the performance of FKD with the conventional fusion approach in terms of both image classification and retrieval applications. In addition, we also examine the performance of EFKD to investigate the effect of incorporating per-pixel Tamura features into the proposed fusion model.

Scene categories database: The Scene categories database [19] contains 15 different classes of grayscale images. A set of sample images is shown in Figure 3. The number of images in the classes varies from 200 to 400 and in total, the database consists of 4485 images. The average size of the images in this database is 300×250 pixels. The different classes of this database are (1) Bedroom, (2) Coast, (3) Forest, (4) Highway, (5) Industrial, (6) Inside city, (7) Kitchen, (8) Living room, (9) Mountain, (10) Office, (11) Open country, (12) Store, (13) Street, (14) Suburb and (15) Tall building.

Caltech 101 database: The Caltech 101 database [10] contains 101 object categories and Google background category. The number of images in each category varies from 31 to 800 and in total, the database consists of 9146 images. The resolutions of images in this database vary from very low to very high. However, most of the images are of 300×300 pixels on average. The different classes of this database include animals, vehicles, flowers to architecture, musical instruments and tools. A set of sample images from the database is given in Figure 4.

6.2 Experimental Settings

To extract kernel descriptors, the same experiment settings are considered as in the baseline [3]. Individual images are resized to no larger than 300×300 pixels. Local descriptors (descriptors for conventional fusion as well FKD and EFKD) are extracted over patch sizes of 16×16 pixels with a spacing of 8 pixels. Image-level descriptors are obtained by EMK encoding on local descriptors. Spatial information in the image-level descriptors is incorporated using spatial pyramid grids (1×1, 2×2 and 4×4) during EMK encoding stage. All computations are performed in the Matlab environment.

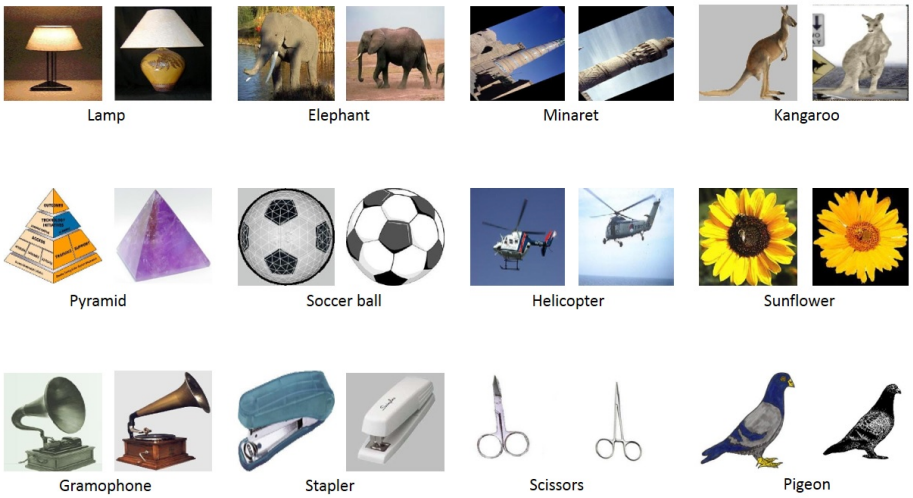

Fig. 4 Sample images from Caltech 101 database

Table 2 Comparison of computation time (Sec) between conventional and proposed fusion approaches

Descriptor type	Conventional fusion	FKD	EFKD
Local descriptor	2.2666	1.2592	1.3275
Image-level descriptor	1.1831	0.2037	0.2212

6.3 Computation Time

While the classification and retrieval results are provided in the separate subsections, comparison of computation time between conventional fusion and the proposed fusion approaches is given in Table 2. Computation times are calculated based on individual images and not for the whole image database. Computation time to extract local descriptors in the conventional fusion approach is the total time to extract three KDES for a single image. In contrast, in the proposed approach, it is the time to extract FKD or EFKD only. Computation time to obtain the image-level descriptors in the conventional fusion approach is the total computation time taken by EMK encoding of three local descriptors along with the time taken to concatenate them. In contrast, in the proposed approach, it is only EMK encoding of FKD or EFKD. From Table 2, it is clear that in both stages, descriptors associated using the proposed fusion approach are more efficient than the conventional one.

6.4 Image Classification

The comparison of classification accuracy of FKD with the conventional fusion approach is given in Table 3 for the two databases considered here. Image classification is performed using support vector machine (SVM)-based classifiers with Laplacian kernel as per the baseline [3]. LIBSVM [5] is used as the classifica-

Table 3 Comparison of Classification accuracies (%) between conventional and proposed fusion approaches

Database	Conventional fusion[3]	FKD	EFKD
Scene categories	86.74	88.32	90.14
Caltech 101	74.58	77.84	80.61

Table 4 Comparison of Classification accuracies (%) of different methods

Method	Scene categories	Caltech 101
EFKD	90.14	80.61
EKD-All [42]	86.34	76.95
SKDES-All [41]	88.77	79.26
SEKD-All [43]	89.23	79.33

tion tool. A 10-fold cross-validation is performed on each database by randomly splitting the individual databases to 10 training and test sets and the average of 10 iterations is reported here. The classification results show that the effectiveness of FKD to classify images of both databases is higher than the conventional fusion approach. The reasons for FKD is more effective than the conventional fusion approach are threefold: first, the candidate kernels of the fused match kernel from which FKD is extracted are constructed by summation instead of Hadamard product of component kernels. Therefore, due to the usage of summation, FKD captures more discriminative information [9] and it is less noise sensitive [16,17]. Second, to extract FKD, information fusion is done at patch level. In contrast, in the conventional approach, information fusion is performed globally by concatenating image-level descriptors. As information fusion at patch level is performed over the raw pixel attributes of local image regions, it can capture the variation and correlation of pixel attributes more effectively than the global fusion of conventional approach. Finally, individual descriptors in the baseline [3] undergo a dimensionality reduction stage and it causes a risk of information loss. However, there is no dimensionality reduction associated to FKD and therefore, there is a minimum chance of information loss.

In Table 3, classification accuracies of EFKD for the databases considered here are also provided. By looking at the performance of EFKD, we can easily conclude that incorporation of per-pixel Tamura features into the proposed fusion model enhanced the classification accuracy. Although, the computation time and dimensionalities of EFKD and FKD are comparable.

Table 4 shows a performance comparison of our proposed approach (EFKD) with the existing kernel descriptor-based methods in terms of classification accuracy. For a fair comparison, we have only considered the combined or serially fused performance of kernel descriptors from these methods to compare the effectiveness of EFKD. From Table 4, it is clear that our proposed fusion approach outperforms the compared methods.

6.5 Image Retrieval

The task of image retrieval is to serach a database based on a query either to find a particular image or an image from a class [23,34]. In this paper, the image retrieval performances are measured by MAP values and using recall-precision

Table 5 Comparison of MAP (%) based on top k retrieved images

Database	k	Conventional fusion [3]	FKD	EFKD
Scene categories	100	60.26	62.19	64.15
Caltech 101	30	58.38	60.84	61.94

curves. There are various similarity and dissimilarity measures between a query image and the database images are available in the literature [29]. We have used Laplacian kernel to obtain the similarity scores. Each image from the databases is used as a query to retrieve rest of the images from the corresponding database. For each query, based on the top 'k' retrieved images, MAP values are obtained and recall-precision curves are plotted.

Precision measures how relevant the retrieved result is and recall measures how many relevant results are retrieved. The mathematical forms of precision and recall are given by (16) and (17) respectively [24, 1].

$$\text{Precision} = \frac{\text{Number of relevant images retrieved}}{\text{Total number of retrieved images}}, \tag{16}$$

$$\text{Recall} = \frac{\text{Number of relevant images retrieved}}{\text{Total number of relevant images in the database}}. \tag{17}$$

Mean average precision (MAP) [8] summarises the precision and recall values to a single value. To calculate MAP, average precision (AP) given by (18) needs to be calculated. The AP for a single query 'q' is the mean over the precision scores after each relevant retrieved item.

$$AP = \sum_{k=1}^{n} (P_k \times r(k))/N_R, \tag{18}$$

where P_k is the Precision value at the k^{th} retrieved image, $r(k)$ is an indicator function which equals 1 if the image at k^{th} rank is relevant, otherwise 0, and N_R is the total number of relevant images retrieved. MAP is the mean of the average precision values over the all sets of queries and it is given by (19).

$$MAP = 1/N_q \sum_{q=1}^{N_q} AP(q), \tag{19}$$

where N_Q is the total number of query images and $AP(q)$ is the average precision of query q.

A comparison of MAP values obtained using conventional fusion, FKD and EFKD is given in Table 5. The corresponding recall-precision curves are given in the Figures 5 and 6 for both databases respectively. From the retrieval results as well, it is clear that irrespective of the databases considered here, FKD outperforms in retrieving images compared to the conventional fusion as MAP values belonging to FKD are always higher compared to the conventional fusion and FKD possesses higher precision compared to the conventional fusion over the same recall rate. In addition, it is clear that combining Tamura features with the gradient, colour and shape features using the proposed fusion approach boosts the overall retrieval performance.

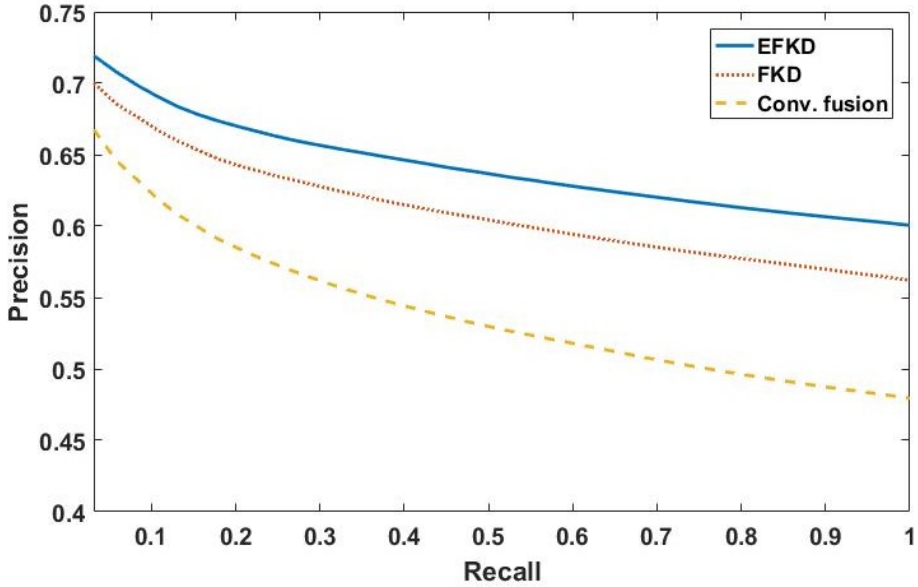

Fig. 5 Recall-Precision curve on Scene categories database

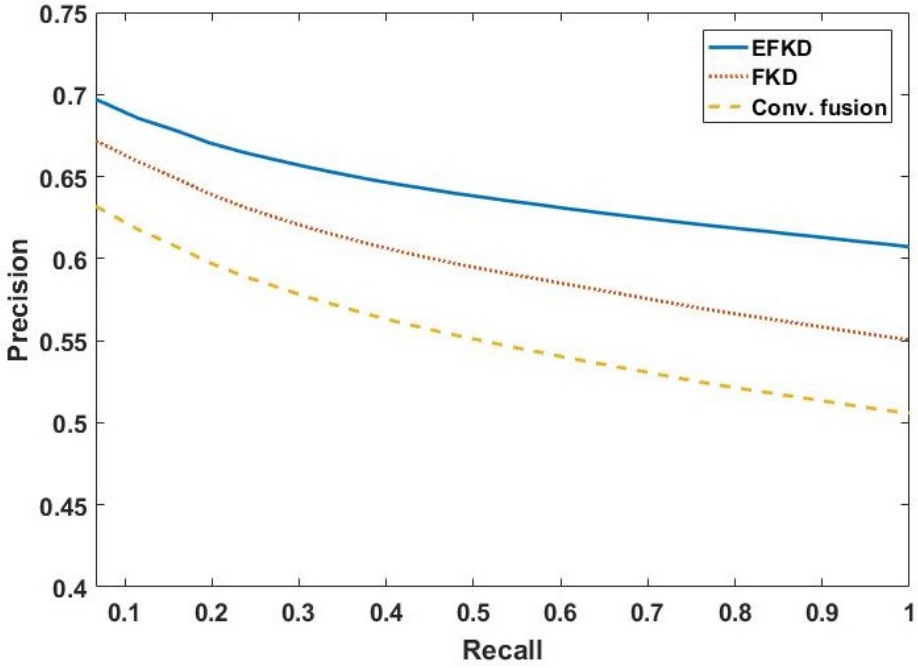

Fig. 6 Recall-Precision curve on Caltech 101 database

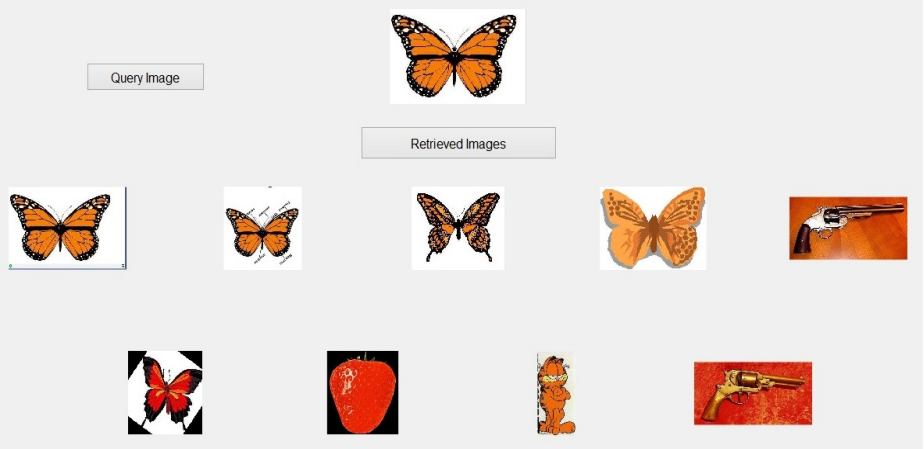

Fig. 7 Retrieval example using conventional fusion

We have compared the performance of our proposed approach with the existing KDES-based methods in terms of classification accuracy in Section 6.4. KDES-based methods in the literature are used for classification task only. In this paper, we aim to investigate how our proposed approach perform for image retrieval task and compared the retrieval result of our proposed approach with the conventional one. By comparing the trend of classification and retrieval results, it is theoretically evident that our proposed approach will outperform the existing KDES-based methods [42,41,43] for retrieval performance as well.

6.6 Qualitative and Quantitative Analysis

In this section, we provide further analysis on the effectiveness of the proposed fusion approach compared to the conventional one. For simplicity, only FKD is used here to represent the proposed fusion approach as FKD and the conventional fusion contain equivalent information. In Figures 7 and 8, two retrieval examples are provided using conventional fusion approach and FKD. The retrieval examples are based on a Matlab-based interface which takes input as a query image and outputs top 9 retrieved images. In both cases (i.e. Figures 7 and 8), same query image (i.e. image of a butterfly) is used to find the similarities with rest of the database images using a Laplacian kernel. It can be clearly observed that using FKD, all top 9 retrieved images are relevant, but using conventional fusion approach, only top 4 and 6-th ranked images are relevant. This is because, using FKD, there is least information loss and less noise corruption occurs. In contrast, using conventional fusion approach, information loss occurs due to dimensionality reduction and it is more noise-sensitive.

To further investigate the above retrieval results, we have analysed the insight of the data. Specifically, we examine how similarity scores of top 9 retrieved images vary with respect to the query. The similarity scores using both approaches are provided in Table 6. As per our experiment, when an image is matched with itself, a similarity score of 1 will be obtained. This indicates, the higher the similarity score

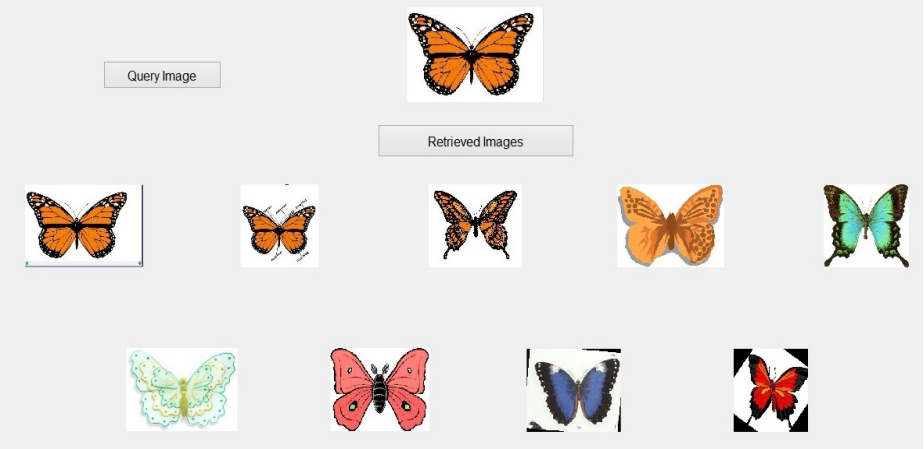

Fig. 8 Retrieval example using FKD

Table 6 Similarity scores of top 9 retrieved images based on a query

Retrieved images	FKD	Conventional fusion
Rank 1	0.9927	0.9914
Rank 2	0.9883	0.9866
Rank 3	0.9824	0.9807
Rank 4	0.9811	0.9785
Rank 5	0.9810	0.9743
Rank 6	0.9809	0.9735
Rank 7	0.9804	0.9711
Rank 8	0.9801	0.9695
Rank 9	0.9798	0.9691

of an image is, it will be retrieved as the higher ranked image. Now, if we see the top 4 images using both approaches, they are similar. However, if we look into the corresponding similarity scores obtained using both approaches, they are different. More precisely, the similarity scores of top 4 images obtained using FKD are higher compared to the scores obtained using the conventional fusion. Moreover, using FKD among the top 9 retrieved images, no irrelevant images scored higher than the relevant images. However, using the conventional fusion approach, 5-th ranked image (irrelevant to the query) obtained higher score compared to the relevant 6-th ranked image. Also, using the conventional fusion approach, 7-,8- and 9-th ranked images obtained higher scores in spite of being irrelevant to the query.

7 Conclusion

In this paper, we propose a novel fusion approach to extract kernel descriptors. KDES framework can turn any kind of pixel attributes to a patch-based descriptor. Conventionally, different types of KDES are fused using serial fusion approach which is less-efficient and less-effective. Therefore, in this paper we have proposed a novel approach that can be used to fuse any kind and any number of pixel

attributes before the descriptor extraction in an efficient and effective way. In addition, we have also shown that incorporation of Tamura features into the proposed fusion approach enhances the overall performance without significantly increasing the time consumption and the descriptor dimensionality.

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

References

1. Bakar, S.A., Hitam, M.S., Yussof, W.N.J.H.W.: Content-based image retrieval using sift for binary and greyscale images. In: Signal and Image Processing Applications (ICSIPA), 2013 IEEE International Conference on, pp. 83–88. IEEE (2013)
2. Bo, L., Lai, K., Ren, X., Fox, D.: Object recognition with hierarchical kernel descriptors. In: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, pp. 1729–1736. IEEE (2011)
3. Bo, L., Ren, X., Fox, D.: Kernel descriptors for visual recognition. In: Advances in Neural Information Processing Systems, pp. 244–252 (2010)
4. Bo, L., Ren, X., Fox, D.: Depth kernel descriptors for object recognition. In: Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on, pp. 821–826. IEEE (2011)
5. Chang, C.C., Lin, C.J.: Libsvm: a library for support vector machines. ACM Transactions on Intelligent Systems and Technology (TIST) **2**(3), 27 (2011)
6. Chatzichristofis, S.A., Boutalis, Y.S.: Fcth: Fuzzy color and texture histogram-a low level feature for accurate image retrieval. In: 2008 Ninth International Workshop on Image Analysis for Multimedia Interactive Services, pp. 191–196. IEEE (2008)
7. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, vol. 1, pp. 886–893. IEEE (2005)
8. Deselaers, T., Keysers, D., Ney, H.: Features for image retrieval: an experimental comparison. Information Retrieval **11**(2), 77–107 (2008)
9. Dov, D., Talmon, R., Cohen, I.: Kernel-based sensor fusion with application to audio-visual voice activity detection. IEEE Transactions on Signal Processing **64**(24), 6406–6416 (2016)
10. Fei-Fei, L., Fergus, R., Perona, P.: Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. Computer Vision and Image Understanding **106**(1), 59–70 (2007)
11. Garcia-Garcia, A., Orts-Escolano, S., Oprea, S., Villena-Martinez, V., Garcia-Rodriguez, J.: A review on deep learning techniques applied to semantic segmentation. arXiv preprint arXiv:1704.06857 (2017)
12. Günter, S., Schraudolph, N.N., Vishwanathan, S.: Fast iterative kernel principal component analysis. Journal of Machine Learning Research **8**(Aug), 1893–1918 (2007)
13. Hu, D., Bo, L., Ren, X.: Toward robust material recognition for everyday objects. In: BMVC, vol. 2, pp. 1–6 (2011)
14. Ju Han, Kai-Kuang Ma: Fuzzy color histogram and its use in color image retrieval. IEEE Transactions on Image Processing **11**(8), 944–952 (2002)
15. Karmakar, P., Teng, S.W., Lu, G., Zhang, D.: A kernel-based approach for content-based image retrieval. In: 2018 International Conference on Image and Vision Computing New Zealand (IVCNZ), pp. 1–6. IEEE (2018)
16. Karmakar, P., Teng, S.W., Zhang, D., Liu, Y., Lu, G.: Improved kernel descriptors for effective and efficient image classification. In: Digital Image Computing: Techniques and Applications (DICTA), 2017 International Conference on, pp. 1–8. IEEE (2017)
17. Karmakar, P., Teng, S.W., Zhang, D., Liu, Y., Lu, G.: Improved tamura features for image classification using kernel based descriptors. In: Digital Image Computing: Techniques and Applications (DICTA), 2017 International Conference on, pp. 1–7. IEEE (2017)
18. Konstantinidis, K., Gasteratos, A., Andreadis, I.: Image retrieval based on fuzzy color histogram processing. Optics Communications **248**(4-6), 375–386 (2005)
19. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: CVPR, vol. 2, pp. 2169–2178. IEEE (2006)

20. Liu, X., Wang, L., Zhang, J., Yin, J.: Sample-adaptive multiple kernel learning. In: Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, AAAI'14, pp. 1975–1981. AAAI Press (2014). URL http://dl.acm.org/citation.cfm?id=2892753.2892827
21. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision **60**(2), 91–110 (2004)
22. Luan, S., Chen, C., Zhang, B., Han, J., Liu, J.: Gabor convolutional networks. IEEE Transactions on Image Processing (2018)
23. Makantasis, K., Doulamis, A., Doulamis, N., Ioannides, M.: In the wild image retrieval and clustering for 3d cultural heritage landmarks reconstruction. Multimedia Tools and Applications **75**(7), 3593–3629 (2016)
24. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press, New York, NY, USA (2008)
25. Ojala, T., Pietikäinen, M., Harwood, D.: A comparative study of texture measures with classification based on featured distributions. Pattern Recognition **29**(1), 51–59 (1996)
26. Pan, H., Olsen, S.I., Zhu, Y.: Feature extraction and learning using context cue and rényi entropy based mutual information. In: International Conference on Pattern Recognition Applications and Methods, pp. 69–88. Springer (2015)
27. Pilario, K.E., Shafiee, M., Cao, Y., Lao, L., Yang, S.H.: A review of kernel methods for feature extraction in nonlinear process monitoring. Processes **8**(1), 24 (2020)
28. Ren, X., Bo, L., Fox, D.: Rgb-(d) scene labeling: Features and algorithms. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, pp. 2759–2766. IEEE (2012)
29. Rubner, Y., Tomasi, C., Guibas, L.J.: The earth mover's distance as a metric for image retrieval. International journal of computer vision **40**(2), 99–121 (2000)
30. Sajjad, M., Ullah, A., Ahmad, J., Abbas, N., Rho, S., Baik, S.W.: Integrating salient colors with rotational invariant texture features for image representation in retrieval systems. Multimedia Tools and Applications **77**(4), 4769–4789 (2018)
31. Serra, G., Grana, C., Manfredi, M., Cucchiara, R.: Covariance of covariance features for image classification. In: Proceedings of International Conference on Multimedia Retrieval, p. 411. ACM (2014)
32. Shawe-Taylor, J., Cristianini, N.: Kernel methods for pattern analysis. Cambridge university press (2004)
33. Tamura, H., Mori, S., Yamawaki, T.: Textural features corresponding to visual perception. IEEE Transactions on Systems, Man, and Cybernetics **8**(6), 460–473 (1978)
34. Tieu, K., Viola, P.: Boosting image retrieval. International Journal of Computer Vision **56**(1-2), 17–36 (2004)
35. Tran, T.H., Nguyen, V.T.: How good is kernel descriptor on depth motion map for action recognition. In: International Conference on Computer Vision Systems, pp. 137–146. Springer (2015)
36. Tuzel, O., Porikli, F., Meer, P.: Region covariance: A fast descriptor for detection and classification. Computer Vision–ECCV 2006 pp. 589–600 (2006)
37. Tuzel, O., Porikli, F., Meer, P.: Pedestrian detection via classification on riemannian manifolds. IEEE transactions on pattern analysis and machine intelligence **30**(10), 1713–1727 (2008)
38. Varma, M., Ray, D.: Learning the discriminative power-invariance trade-off. In: Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on, pp. 1–8. IEEE (2007)
39. Vedaldi, A., Gulshan, V., Varma, M., Zisserman, A.: Multiple kernels for object detection. In: Computer Vision, 2009 IEEE 12th International Conference on, pp. 606–613. IEEE (2009)
40. Voulodimos, A., Doulamis, N., Doulamis, A., Protopapadakis, E.: Deep learning for computer vision: A brief review. Computational Intelligence and Neuroscience **2018**, 7068349–7068349 (2018)
41. Wang, P., Wang, J., Zeng, G., Xu, W., Zha, H., Li, S.: Supervised kernel descriptors for visual recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2858–2865 (2013)
42. Xie, B., Liu, Y., Zhang, H., Yu, J.: Efficient kernel descriptor for image categorization via pivots selection. In: Image Processing (ICIP), 2013 20th IEEE International Conference on, pp. 3479–3483. IEEE (2013)
43. Xie, B., Liu, Y., Zhang, H., Yu, J.: A novel supervised approach to learning efficient kernel descriptors for high accuracy object recognition. Neurocomputing **182**, 94–101 (2016)

44. Yang, S., Bo, L., Wang, J., Shapiro, L.G.: Unsupervised template learning for fine-grained object recognition. In: Advances in Neural Information Processing Systems, pp. 3122–3130 (2012)
45. Zhang, D., Islam, M.M., Lu, G.: A review on automatic image annotation techniques. Pattern Recognition **45**(1), 346–362 (2012)
46. Zhou, Y., Ye, Q., Qiu, Q., Jiao, J.: Oriented response networks. In: Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on, pp. 4961–4970. IEEE (2017)