DCT Inspired Feature Transform for Image Retrieval and Reconstruction

Citation for published version:
Wang, Y, Shi, M, You, S & Xu, C 2016, 'DCT Inspired Feature Transform for Image Retrieval and Reconstruction' IEEE Transactions on Image Processing, vol. 25, no. 9, pp. 4406-4420. DOI: 10.1109/TIP.2016.2590323

Digital Object Identifier (DOI):
10.1109/TIP.2016.2590323

Link:
Link to publication record in Edinburgh Research Explorer

Document Version:
Peer reviewed version

Published In:
IEEE Transactions on Image Processing

DCT Inspired Feature Transform for Image Retrieval and Reconstruction

Yunhe Wang, Miaojing Shi, Shan You, and Chao Xu

***Abstract*— Scale invariant feature transform (SIFT) is effective for representing images in computer vision tasks, as one of the most resistant feature descriptions to common image deformations. However, two issues should be addressed: first, feature description based on gradient accumulation is not compact and contains redundancies; second, multiple orientations are often extracted from one local region and therefore produce multiple descriptions, which is not good for memory efficiency. To resolve these two issues, this paper introduces a novel method to determine the dominant orientation for multiple-orientation cases, named discrete cosine transform (DCT) intrinsic orientation, and a new DCT inspired feature transform (DIFT). In each local region, it first computes a unique DCT intrinsic orientation via DCT matrix and rotates the region accordingly, and then describes the rotated region with partial DCT matrix coefficients to produce an optimized low-dimensional descriptor. We test the accuracy and robustness of DIFT on real image matching. Afterward, extensive applications performed on public benchmarks for visual retrieval show that using DCT intrinsic orientation achieves performance on a par with SIFT, but with only 60% of its features; replacing the SIFT description with DIFT reduces dimensions from 128 to 32 and improves precision. Image reconstruction resulting from DIFT is presented to show another of its advantages over SIFT.**

***Index Terms*— Image representation, DCT intrinsic orientation, DIFT, image matching, image retrieval, image reconstruction.**

I. Introduction

LOCAL features are widely adopted in many computer vision applications, *e.g.*, image retrieval [50] and classification [58]. One of the most successful local representations is the scale invariant feature transform (SIFT) presented by Lowe [32]. It is famous for its invariance to scale and rotation, but is not compact and precise. PCA is often applied to remove the correlation and redundancies among the SIFT components [28]. However, PCA-SIFT is a data-driven approach depending on the PCA training. The motivation of this paper is thus to find a data-independent transform that decorrelates the descriptor in all dimensions. To achieve this, we are motivated by the idea behind the JPEG image coding standard, the discrete cosine transform (DCT) [2], [40] – a local region can be linearly composed by a set of basic patterns. Owing to the fact that all patterns are orthogonal to each other, the DCT representation has no redundancy between components; the description can thus be very compact. Building upon this idea, we propose DCT inspired feature transform following the three steps (see Fig.1) in SIFT: local region (and interest point) detection, orientation assignment, and feature description.

In the local region detection, Difference of Gaussian (DoG) is utilized as an approximation of Laplace of Gaussian (LoG), but is much faster. Difference octaves in images are calculated to construct the Gaussian pyramid. The same features can be detected on different scales of the pyramid to gain the scale invariance. In [36] and [38], Mikolajczyk *et.al.* add affine invariance into the Hessian-Laplace detector and combine the Laplace scale with Hessian-affine. It outperforms others and is used to replace DoG in latest SIFT. We adopt the same manner to detect the local region.

As for the orientation assignment, the dominant orientation of each interest point is computed inside the local region and rotated accordingly for rotation invariance. According to [12] and [32], there might exist multiple dominant orientations in one region, and therefore produce multiple feature descriptions, which is very costly for memory efficiency. In this paper, we compute a DCT intrinsic orientation from the ratio of DCT coefficients $\mathcal{C}_{0,1}$ and $\mathcal{C}_{1,0}$; they correspond to the first horizontal and vertical component in the DCT basis in Fig.1. We formulate their ratio as a measurement of the region's intrinsic orientation, and show that it deals precisely with rotation invariance and disregards subtle details, unlike multiple orientations. Meanwhile, it significantly saves memory cost compared to multiple orientations at each interest point.

In the region description, SIFT first divides the local region into sixteen equal blocks, then gradients are calculated in each block and accumulated into eight bins according to their directions. By concatenating them together, the final descriptor is of 128 dimensions. Since gradients in different blocks can be very similar, there exist redundancies in these gradients-described approaches; accumulated gradients in the same bin are pooled as a histogram frequency, local information is lost in this pooling. Efficiency and precision seem to contradict each other. This paper proposes a DCT inspired feature transform description, DIFT, which successfully tackles two

Manuscript received October 14, 2015; revised April 12, 2016 and June 28, 2016; accepted June 28, 2016. Date of publication July 11, 2016; date of current version August 2, 2016. This work was supported by the National Natural Science Foundation of China under Grant NSFC 61375026 and 2015BAF15B00. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Xiaochun Cao. *(Corresponding author: Miaojing Shi.)*

Y. Wang, S. You, and C. Xu are with the Key Laboratory of Machine Perception, Ministry of Education, Cooperative Medianet Innovation Center, School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China (e-mail: wangyunhe@pku.edu.cn; youshan@pku.edu.cn; xuchao@cis.pku.edu.cn).

M. Shi is with the School of Informatics, Institute of Perception, Action and Behaviour, The University of Edinburgh, Edinburgh EH8 9AB, U.K. (e-mail: miaojing.shi@ed.ac.uk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2016.2590323

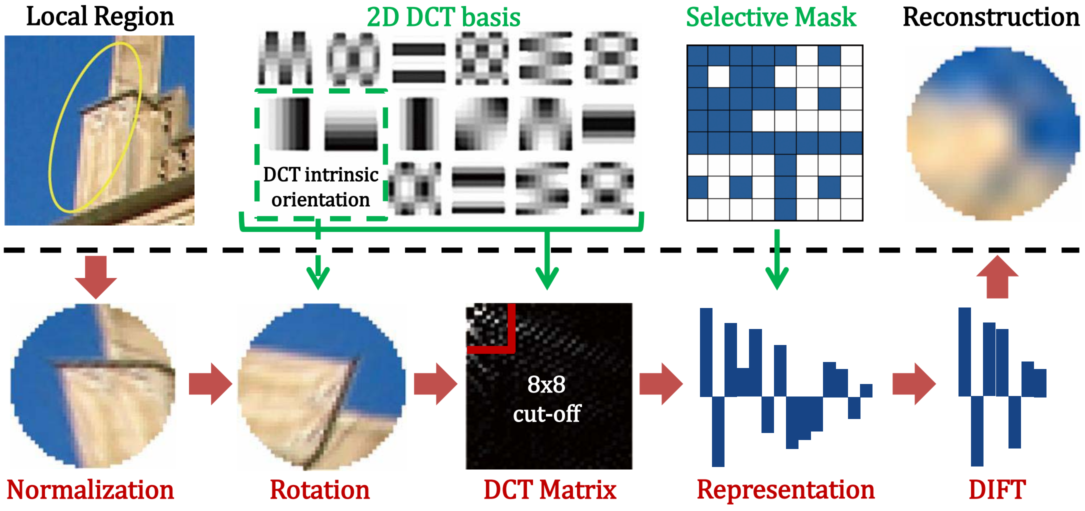


Fig. 1. The flow chart of DIFT: 1) a local region (yellow ellipse) is extracted from an image and normalized to a circle; 2) the local region is rotated by the DCT intrinsic orientation, which is computed via the first horizontal and vertical component of the 2D DCT basis; 3) the rotated region is described by its DCT matrix and followed by a 8×8 cut-off operation; a selective mask is applied to the representation to screen the optimum 32-dimensional DIFT descriptor. A reverted operation is illustrated in the end to show the reconstruction result using DIFT.

"contradictory" aspects. Each local region is described by its DCT coefficients, high-frequency components are firstly dropped as a simulation of the Gaussian processing; a distinctive and compact 32-dimensional DIFT is further selected by optimizing a gradient-based loss function; via inverse DCT, DIFT is able to restore the local region.

We test the accuracy and robustness of DIFT on a famous benchmark dataset [37] for image matching and present a systematic assessment for feature orientation and description. Afterwards, we apply DIFT in two real applications, image retrieval and reconstruction; we show significant improvement can be achieved over state of the art.

Our work is inspired by [40], which employs DCT coefficients to represent the MSER regions in an image, but it is fundamentally different: 1) MSER detection is of redundancy and post-processing is usually required. In contrast, we plant our description on the basis of Hessian-affine detector [36], [38], which is more robust and faster; 2) to tackle the issue of multiple orientations in hessian-affine detector, we propose DCT intrinsic orientation to determine the dominant orientation of a local region. Experiments show that embedding it into the SIFT description will achieve the same retrieval performance, but with a significantly reduced memory overhead, typically by a factor of two; 3) instead of simply choosing top 10 DCT coefficients in [40], we carefully analyze and optimize the selection of feature dimensions in DIFT, and demonstrate that replacing the SIFT description with the DIFT description not only reduces the dimensions but also improves the retrieval precision; 4) in the end, we propose a simple yet effective way to reconstruct the image from DIFT. It helps us intuitively visualize the attribute of an image feature and suggests ways to improve the retrieval performance.

II. Related Works

This section surveys the literature of local feature extraction and its application in image retrieval and reconstruction: 1) local region detection and orientation estimation; 2) feature description; 3) image retrieval and reconstruction.

A. Local Region Detection and Orientation Estimation

Apart from the DoG detector [32], there exist a variety of local region and key point detection algorithms. In [19], Harris and Stephens utilize the Harris detector to detect the local region owing to its invariance to rotation. Mikolajczyk and Schmid [34] embed the scale space theory into Harris and propose a new scale-invariant detector named Harris-Laplace; similarly, they [35] also embed affine Gaussian scale space into Harris and propose an affine-invariant Harris. Hessian-Laplace [38] is another scale-invariant detection method, which makes use of Hessian matrix instead of Harris due to its superior accuracy; Mikolajczyk and Schmid further [36] improve this method by adding affine invariance to it and propose a Hessian-affine detector outperforming both Harris-affine and Hessian-Laplace.

Regarding the orientation assignment, SIFT [32] computes the dominant orientations by counting the peaks on the gradient histogram. Fan *et.al.* [12] point out SIFT orientations are not precise, and compute the orientations at every pixel inside the local region, which is a highly complex process. Rublee *et.al.* [45] propose to compute the orientation via the intensity centroid; Ahonen and He [3] accumulate the texture pattern LBP (local binary pattern) inside each local region, which can be seen as a serial of 0-1 sequence and the rotation of the local region is thereby expressed by the rotation of the sequence. These approaches either replace gradients with other patterns to implicitly compute the orientation or are of high complexity, computing the exact orientation. We are inspired by the work of Shen and Sethi [47], in which they suggest several possible manners of computing orientations from the DCT coefficients, but they don't give any proof or solid explanations. On the other hand, we propose a DCT intrinsic orientation and provide careful formulations and discussions.

B. Feature Description

Most representative feature description methods are built upon the accumulation and aggregation of local gradients as initially proposed in SIFT [12], [28], [37]. Ke and Sukthankar [28] design PCA-SIFT by calculating both horizontal and vertical gradients in a local region. A 128-dimensional descriptor is reduced to a 36-dimensional vector through a PCA rotation matrix. Because the matrix is pre-trained, the method is not scalable. SURF [7] divides the image into 4×4 blocks according to the dominant direction and applies the Haar templates in each block. Responses for these templates are accumulated to generate descriptors. It is a 64-dimensional descriptor. Some other works try to embed *e.g.*, color [1], orientation [20], [37], shape [8], steerable filter [13], into the local description to improve the robustness and scalability of the feature descriptor.

C. Image Retrieval and Reconstruction

Despite the fact that these varieties of feature descriptions show their superiority in a lot of real applications, SIFT is still the most popular and reliable local descriptor and is widely used in image matching [37] and retrieval [25], [43], [51], [52]. Arandjelović and Zisserman [4] propose rootSIFT to reduce the larger bin values relative to smaller bin values in SIFT and improve retrieval precision. Neither SIFT nor rootSIFT is compact and can be compressed by applying PCA [21], [28]. We propose DIFT description which produces a lower-dimensional descriptor compared to SIFT. Local features are usually fed into an inverted file structure *e.g.*, BOW [15], [43] and SMK [55], or aggregated into a global representation *e.g.*, fisher vector [42] and VLAD [25], [26], [49] for image retrieval.

Reconstructing an image from local features is another interesting topic in computer vision. It raises issues such as privacy [62], visualization [59], and overhead storage [11], [27]. There are typically two ways to carry out the reconstruction: one is to directly search patches matching the given descriptors in the corpus [11], [27], [59]; the other is to search for the local patch from an additional set [62]. Due to the nonlinear transform of local features, a typical searching process is rather inefficient, and the reconstruction is vague and biased, sometimes terribly blurred and hard to understand [27]. Due to the inverse attribute of DCT, we are able to reconstruct the local patch and restore the whole image to a remarkable degree.

III. DCT Inspired Feature Transform

This section reviews the basic knowledge of DCT and then presents the DCT inspired feature transform in three steps: local region detection, DCT intrinsic orientation assignment, and DIFT description.

A. Discrete Cosine Transform, DCT

DCT is widely applied in signal and image processing [2], [10]. It expresses a signal/an image in terms of cosine functions oscillating at different frequencies. Specifically, for a given image patch $f(x, y)$, its DCT coefficients are defined as follows:

$$\mathcal{C}_{i,j} = \alpha_i \alpha_j \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) \times \cos\left(\frac{\pi(2x+1)i}{2N}\right) \cos\left(\frac{\pi(2y+1)j}{2N}\right), \quad (1)$$

where

$$\alpha_i = \begin{cases} \sqrt{\frac{1}{N}}, & if \quad i = 0, \\ \sqrt{\frac{2}{N}}, & otherwise. \end{cases} \quad (2)$$

A local patch is usually rescaled to N^2 to generate visual descriptor, *e.g.*, a 41 × 41 patch.

B. Local Region Detection

In order to guarantee the scale and affine invariance, the Hessian-affine detector first computes the scale parameter of each key point to select the scale-invariant local region; then estimates the affine-invariant shape using the Hessian matrix; for its SIFT description, the affine region is further normalized to a circle [37], [38]. For the sake of scale and affine invariance in DIFT, we adopt the same manner to detect local regions. To apply this scale and affine invariant region to DCT, we use the circumscribed square of the circle (41 × 41) and fill the outside pixels with zeros [12].

C. DCT Intrinsic Orientation Assignment

For some specific urban scenes, it is showed that a single up-right orientation gives the best feature representation [22], [56]; nonetheless, for natural scenes or man-made objects, dominant orientations are usually required. SIFT [32] calculates the dominant orientations of each local region by counting the peaks in the gradient histogram. In [12] and [32], it is pointed out that for the local region with multiple peaks of similar magnitudes, there will be multiple orientations created at the same location. This multiple-peak phenomenon may produce bursty features and corrupt the similarity measure [23], [48], and also bias the matching between similar patches [12]. In this section, we present a novel way to deal with the rotation invariance. Instead of counting the peaks, we compute an intrinsic orientation from DCT coefficients $\mathcal{C}_{1,0}$ and $\mathcal{C}_{0,1}$ in Fig.1 and rotate each local region accordingly. We prove this orientation is unique by showing that similar local regions are always rotated into the same intrinsic position. In this sense, we dramatically reduce the feature number in each image. Note that the proposed DCT intrinsic orientation assignment does not apply to the single up-right orientation case.

Intuitively, looking at the dashed green block in Fig.1, $\mathcal{C}_{0,1}$ and $\mathcal{C}_{1,0}$ reflect the general horizontal and vertical information in the frequency domain, and their ratio $\frac{\mathcal{C}_{0,1}}{\mathcal{C}_{1,0}}$ therefore discloses the orientation measure in it. It resembles the global smoothed gradient orientation in [16]; however, it is a different measure in the frequency domain, which is more robust as a global measure whilst more responsive to local details. In the following section, we prove this observation mathematically, and demonstrate its superiority over the multiple orientations in SIFT.

Given a local region $f(x, y)$, we assume it is rotated from an initial patch $f_0(x_0, y_0)$ by clockwise angle θ, $[x_0, y_0]$ denotes the initial position of any pixel of f_0. $[x, y]$ is thereby obtained:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} + \begin{bmatrix} -0.5W_0\cos\theta + 0.5H_0\sin\theta + 0.5W \\ -0.5W_0\sin\theta - 0.5H_0\cos\theta + 0.5H \end{bmatrix}, \quad (3)$$

where W_0 and H_0 are width and height coordinates of the original square, $W_0 = H_0 = N - 1$, where N denotes the square size. W and H are the current corresponding coordinates. Since each local region is indeed a circle located inside its circumscribed square, the rotation of the square can be seen as the rotation of its internal circle. In this context, $W = H = N - 1$, which corresponds to the circle diameter. The circle center is $\left[\frac{N-1}{2}, \frac{N-1}{2}\right]$. We denote by $\widehat{\mathcal{C}}_{1,0}$ and $\widehat{\mathcal{C}}_{0,1}$ the initial DCT coefficients (1) in f_0,

$$\widehat{\mathcal{C}}_{1,0} = \frac{\sqrt{2}}{N}\sum_{x_0=0}^{N-1}\sum_{y_0=0}^{N-1} f_0(x_0, y_0)\cos\left(\frac{\pi\ (2x_0+1)}{2N}\right),$$
$$\widehat{\mathcal{C}}_{0,1} = \frac{\sqrt{2}}{N}\sum_{x_0=0}^{N-1}\sum_{y_0=0}^{N-1} f_0(x_0, y_0)\cos\left(\frac{\pi\,(2y_0+1)}{2N}\right). \quad (4)$$

Since patch pixel value $f(x, y)$ remain the same with $f_0(x_0, y_0)$ after the rotation, DCT coefficients $\mathcal{C}_{1,0}$ and $\mathcal{C}_{0,1}$ in f can be written as

$$\mathcal{C}_{1,0} = \frac{\sqrt{2}}{N}\sum_{x,y} f(x, y)\cos\left(\frac{\pi\ (2x+1)}{2N}\right) \quad (5)$$
$$= \frac{\sqrt{2}}{N}\sum_{x_0,y_0} f_0(x_0, y_0)\cos\left(\frac{\pi\ (2x+1)}{2N}\right),$$
$$\mathcal{C}_{0,1} = \frac{\sqrt{2}}{N}\sum_{x_0,y_0} f_0(x_0, y_0)\cos\left(\frac{\pi\ (2y+1)}{2N}\right). \quad (6)$$

Let $p_{x_0,y_0}(\theta) = \cos\left(\frac{\pi(2x+1)}{2N}\right)$ and $q_{x_0,y_0}(\theta) = \cos\left(\frac{\pi(2y+1)}{2N}\right)$ and substitute (3) into them, we have

$$\begin{aligned} p_{x_0,y_0}(\theta) &= \cos\left(\frac{\pi}{N}\left(x_0 - \frac{N-1}{2}\right)\cos\theta - \frac{\pi}{N}\left(y_0 - \frac{N-1}{2}\right)\sin\theta + \frac{\pi}{2}\right) \\ &= \cos\left(\frac{r\pi}{N}\cos(\theta+\theta_0) + \frac{\pi}{2}\right) \\ &= -\sin\left(\frac{r\pi}{N}\cos(\theta+\theta_0)\right) \\ &\approx -\frac{r\pi}{N}\cos(\theta+\theta_0) \triangleq p_{r,\theta_0}(\theta), \end{aligned} \quad (7)$$

where $r = \sqrt{(x_0 - \frac{N-1}{2})^2 + (y_0 - \frac{N-1}{2})^2}$, and $x_0 = \frac{N-1}{2} + r\cos\theta_0$, $y_0 = \frac{N-1}{2} + r\sin\theta_0$, θ_0 denotes the initial angle from certain pixel $[x_0, y_0]$ to the center $\left[\frac{N-1}{2}, \frac{N-1}{2}\right]$ in the polar coordinate system. Similarly, we have

$$\begin{aligned} q_{x_0,y_0}(\theta) &= -\sin\left(\frac{r\pi}{N}\sin(\theta+\theta_0)\right) \\ &\approx -\frac{r\pi}{N}\sin(\theta+\theta_0) \triangleq q_{r,\theta_0}(\theta). \end{aligned} \quad (8)$$

Replace x_0 and y_0 with their polar forms in $f_0(x_0, y_0)$, we have $f_0(x_0, y_0) = f_0(r, \theta_0)$, *i.e.*,

$$f_0(r, \theta_0) = f_0(\frac{N-1}{2} + r\cos\theta_0, \frac{N-1}{2} + r\sin\theta_0). \quad (9)$$

Substitute it into (6) and (6), we have:

$$\mathcal{C}_{1,0} = \frac{\sqrt{2}}{N}\sum_{r,\theta_0} f_0(r, \theta_0)p_{r,\theta_0}(\theta),$$
$$\mathcal{C}_{0,1} = \frac{\sqrt{2}}{N}\sum_{r,\theta_0} f_0(r, \theta_0)q_{r,\theta_0}(\theta). \quad (10)$$

Divide them and make use of (7) and (8), we have

$$\begin{aligned} \frac{\mathcal{C}_{0,1}}{\mathcal{C}_{1,0}} &\approx \frac{\sum_{\theta_0} g(\theta_0)\sin(\theta+\theta_0)}{\sum_{\theta_0} g(\theta_0)\cos(\theta+\theta_0)} \\ &\triangleq \frac{h_1(\theta)}{h_2(\theta)} \triangleq h(\theta), \end{aligned} \quad (11)$$

where $g(\theta_0) = \frac{\pi}{N}\sum_r r f_0(r, \theta_0) \geq 0$. Obviously,

$$h_1'(\theta) = h_2(\theta), h_2'(\theta) = -h_1(\theta). \quad (12)$$

Hence, we can construct a differential equation of DCT intrinsic orientation θ,

$$\begin{aligned} h'(\theta) &= \frac{h_1'(\theta)h_2(\theta) - h_2'(\theta)h_1(\theta)}{h_2^2(\theta)} \\ &= \frac{h_2^2(\theta) + h_1^2(\theta)}{h_2^2(\theta)} = 1 + h^2(\theta). \end{aligned} \quad (13)$$

Its solution is

$$h(\theta) = \tan(\theta + \varepsilon), \quad (14)$$

with the initial value condition,

$$\tan(\varepsilon) = h(0) = \frac{\widehat{\mathcal{C}}_{0,1}}{\widehat{\mathcal{C}}_{1,0}}. \quad (15)$$

Considering the fact the we have no idea of the initial patch f_0 and its $\hat{\mathcal{C}}_{0,1}$ and $\hat{\mathcal{C}}_{1,0}$, we can only obtain

$$\varphi = \theta + \varepsilon = \begin{cases} \arctan\left(\frac{\mathcal{C}_{0,1}}{\mathcal{C}_{1,0}}\right), & if \quad \mathcal{C}_{1,0} \geq 0 \\ \arctan\left(\frac{\mathcal{C}_{0,1}}{\mathcal{C}_{1,0}}\right) + \pi, & if \quad \mathcal{C}_{1,0} < 0 \end{cases} \quad (16)$$

with current $\mathcal{C}_{0,1}$ and $\mathcal{C}_{1,0}$. Each pixel in the current local patch f can thus be rotated with counterclockwise angle $\theta + \varepsilon$.

We call $\theta + \varepsilon$ DCT intrinsic orientation φ. All the local regions are rotated to certain positions where $\mathcal{C}_{0,1} \simeq 0$, meaning that the coefficient of the second DCT basis (row-wise) is nearly zero. We call this position the local region's intrinsic position. After rotating each local region back to its intrinsic position, we compute $\mathcal{C}$ for DIFT description (Sec.III-D).

Fig.2 shows the rotation invariance of DIFT compared to SIFT: given an initial local region, we rotate it manually to

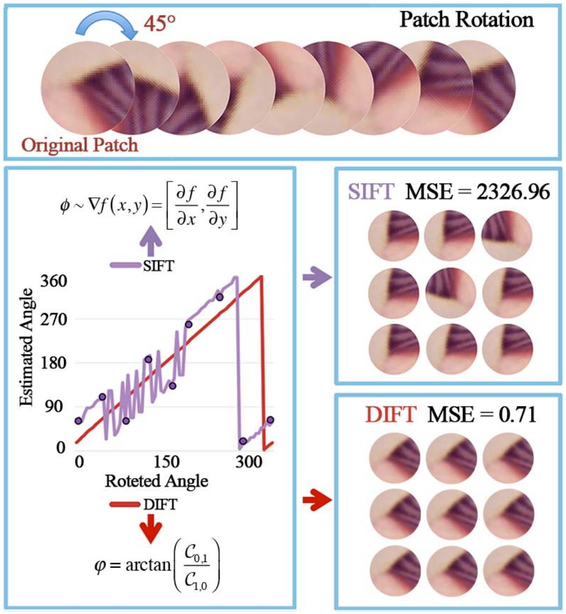

45°
Patch Rotation
Original Patch
$\phi \sim \nabla f(x,y) = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right]$
SIFT
Estimated Angle
360
270
180
90
0
0
150
300
Roteted Angle
DIFT
$\varphi = \arctan\left(\frac{\mathcal{C}_{0,1}}{\mathcal{C}_{1,0}}\right)$
SIFT MSE = 2326.96
DIFT MSE = 0.71

Fig. 2. A comparison between SIFT dominant orientation and DIFT intrinsic orientation. Given an initial image clockwise rotated in the top block, the dominant angle is estimated in SIFT and DIFT in the bottom left block; each local patch is rotated back using the estimated angle from SIFT and DIFT in the two bottom right blocks, respectively. The mean squared error (MSE) of the angle difference (0°-360°) is employed to evaluate the estimation performance.

generate a sequence of rotated local regions, *e.g.*, every 45° in the top block, which is the true rotated angle. We give the estimated orientations computed in SIFT and DIFT, respectively, in the bottom left block. It can be seen that, the estimation by DIFT is exact and consistent, which is denoted by a straight red line in the Descartes coordinate system; the denotation of SIFT is the purple line. We chose nine points (dotted on the line) corresponding to the top block and rotate the regions as shown in the bottom right blocks. Clearly, all local regions can be rotated into their intrinsic positions according to their φ. However, SIFT ends up with a number of dominant orientations due to too many subtle details.

D. DIFT Description

This section presents DIFT description. Owing to the fact that all DCT patterns are orthogonal to each other, the DCT representation has no redundancies between components; it keeps both color and texture information in its description, which makes it possible for local region reconstruction.

Illumination change is considered not to be critical for SIFT [37]. It can be expressed in an image patch f (we use f to denote the rotated region by φ) as $f \times e + c$, where e and c are constants [37]. Slope e can be simply tackled by normalization as long as the offset c is removed, while in DIFT c is captured by the DC component $\mathcal{C}_{0,0}$, the remaining AC components in DCT are independent of this offset. To remove the change in $\mathcal{C}_{0,0}$, we propose to subtract each pixel intensity f_i by the minimum value f_m inside the patch, $f_i^* = f_i - f_m$. The subtraction does not affect the AC components.

For a fixed image patch after the preprocessing (Sec.III-B andIII-C), we have the basic DCT patterns

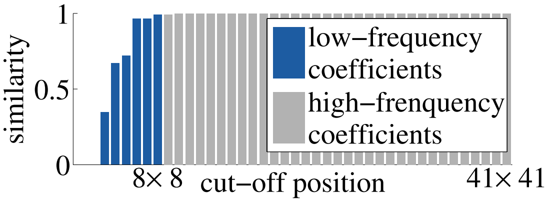


Fig. 3. SIFT similarities at different cutoff positions. Each bar denotes the cosine similarity between two SIFTs extracted from the original and restored local regions. Top-left 8×8 cut-off position produces the similarity 99.8%.

as showed in Fig.1 to represent it. Using (1), we obtain the corresponding DCT coefficient matrix $\mathcal{C}$ whose elements denote the weights of basic DCT patterns. The descriptor is not fully compact yet, thus we propose two steps to achieve optimum compactness and representativeness: one is the high-frequency cutoff and the next is dimensionality selection to choose the most distinctive components. The motivation behind this is: when we restore the local region from DIFT, we want it to be able to retain the description of SIFT; keep the gradient information as good as SIFT; capture the texture and brightness information beyond SIFT.

1) High-Frequency Cutoff: Considering the extraction of SIFT, a Gaussian filter is first applied to smooth the local region. It can be seen as a space domain low-pass filter to remove white noise [54]. We adopt the same idea in DIFT. We know that, DCT approximates the KL transformation for uniformly distributed data [2], [10] and it can transform the local region to the frequency domain. Frequency coefficients increase from top-left to bottom-right in the DCT coefficient matrix. We discard the high-frequency coefficients, and restore the local region using the remaining part. We can compute a new SIFT from the restored region; we want it to be able to retain the description of the original SIFT and thereby measure the cosine similarity between the two SIFTs.

As illustrated in Fig.3, for an 41×41 region, cutoff position 8×8 is the best choice clearly. SIFT description is the same, while the current dimension (64) is only $(8/41)^2 = 3.81\%$ of the original one.

2) Dimensionality Selection: As can be seen in Fig.3, if we choose the top-left 8×8 components, the corresponding similarity is still 99.8%; if we choose 7×7 instead, the similarity is quickly decreased to 96.1%. Hence, further removing components by frequency is not optimum henceforth, we screen these left coefficients via an objective function:

$$U^* = \arg\min_{U} \frac{1}{l} \sum_{i=1}^{l} \mathcal{L}\left(\mathcal{D}\left(\mathcal{C}^{-1}\left(U \cdot \mathcal{C}\left(I\right)\right)\right), \mathcal{D}\left(I\right)\right) + \gamma \, ||U||_1, \tag{17}$$

where l is the number of sample patches. $\mathcal{D}(\cdot)$ denotes the method extracting descriptors (*e.g.*, SIFT); U is the selective weight matrix with either 0 or 1; $\mathcal{C}$ is the DCT; I denotes an image patch; $\mathcal{L}$ is loss function, which can be defined as a matter of cosine similarity between two descriptors; $||U||_1$ is the sparsity regularization term with a parameter γ.

The motivation behind the dimensionality reduction is that we want DIFT to retain the gradient information *e.g.* in SIFT, and in the meantime its dimensionality is as low as possible.

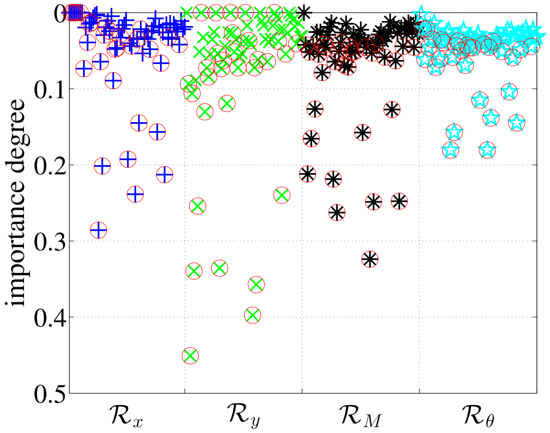


Fig. 4. Mean importance score of each component in the 64 dimensions over 5M local regions. Selected dimensions with largest $\mathcal{R}^d$ are marked with red circles. Their distributions are displayed on the four gradient-based criterions.

The objective function (17) is not convex and it is hard to reach the global optimum. In order to solve this problem, we propose four gradient-based criterions in replacement,

$$\begin{aligned}\mathcal{R}_x\left(\tilde{I}_d\right) &= \frac{1}{N^2}\sum_n \frac{1}{2}\left|Gx_n - \tilde{G}x_n\right|,\\ \mathcal{R}_y\left(\tilde{I}_d\right) &= \frac{1}{N^2}\sum_n \frac{1}{2}\left|Gy_n - \tilde{G}y_n\right|,\\ \mathcal{R}_M\left(\tilde{I}_d\right) &= \frac{1}{N^2}\sum_n \frac{1}{\sqrt{2}}\left|M_n - \tilde{M}_n\right|,\\ \mathcal{R}_\theta\left(\tilde{I}_d\right) &= \frac{1}{N^2}\sum_n \frac{1}{2\pi}\left|\theta_n - \tilde{\theta}_n\right|. \end{aligned} \tag{18}$$

Since method $\mathcal{D}(\cdot)$ is a nonlinear transformation, we adopt a heuristic manner to solve (17): $\tilde{I}_d$ denotes the restored patch with the d-th DCT coefficient being removed. G_x and G_y are the horizontal and vertical gradients at point n in the original patch, $\tilde{G}x$ and $\tilde{G}y$ are the corresponding values in the restored patch. There are N^2 points in total in the patch. M is the gradient module; θ is the angle. We measure the impact of removing every single d-th component in DCT over the four criterions,

$$\mathcal{R}^d = \frac{1}{4}\left|\mathcal{R}_x\left(\tilde{I}_d\right) + \mathcal{R}_y\left(\tilde{I}_d\right) + \mathcal{R}_M\left(\tilde{I}_d\right) + \mathcal{R}_\theta\left(\tilde{I}_d\right)\right|, \tag{19}$$

where the loss function $\mathcal{L}$ is therefore defined as $\mathcal{L} = -\frac{1}{l}\sum_{i=1}^{l} U^d\mathcal{R}_l^d$. U^d denotes selective weight matrix with the d-th entry being zero, and others are 1. We keep the components that produce the maximum entropy in the gradient distribution. These kept components are believed to be the best representatives of the local region. It is rational to evaluate the optimum selection of every entry for each descriptor. However, it would be computationally inefficient in practice. We instead propose a statistical learning scheme to carry out the dimensionality reduction once for all: we extract 5M local regions from Flickr1M dataset [24] and rank the average loss in $\mathcal{R}^d$ by removing every component in the 64 dimensions; we observed that the top-ranked 32 dimensions is a balanced choice overall. As illustrated in Fig.4, the chosen dimensions hold in general distinctive importance scores (large loss in $\mathcal{R}^d$) over the mean values of four criterions: $\mathcal{R}_x$, $\mathcal{R}_y$, $\mathcal{R}_M$ and $\mathcal{R}_\theta$. The selective mask of the dimension index is fixed after the training and drawn in Fig.1. The fixed mask may fall into sub-optimum for a specific descriptor; notwithstanding, it is much faster to conduct the dimensionality reduction with a fixed mask. We restore the local region from the selected DIFT dimensions and extract a new SIFT from the restored region. Its similarity value to the original SIFT is 98.0%. Similar idea of learning the optimal selective mask in a training set can be found in [28] and [37] as well.

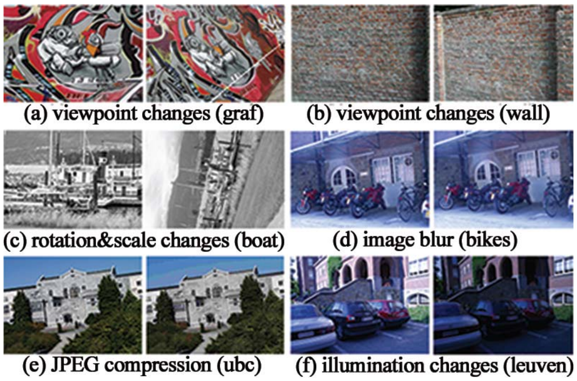

(a) viewpoint changes (graf)
(b) viewpoint changes (wall)
(c) rotation&scale changes (boat)
(d) image blur (bikes)
(e) JPEG compression (ubc)
(f) illumination changes (leuven)

Fig. 5. Example images for image matching experiment.

We concatenate the chosen components in $\mathcal{C}$ as a vector and l_2-normalize it. It thereby produces DIFT. Specifically, we also propose a color space version of DIFT, it takes 32-sub vector from each color channel, and concatenate them together as a 96-dimensional vector, we denote it by DIFT$^{\mathsf{C}}$.

In the following, we first test the accuracy and robustness of DIFT on the real image matching task, and then apply DIFT in two scenarios: image retrieval and reconstruction. In the first scenario, we test the performance of DIFT on representative retrieval models. In the second scenario, we give our reconstruction procedure from DIFT and illustrate the restoring results.

IV. Image Matching

Image matching is a typical and challenging task, which is usually utilized for evaluating the robustnesses of local descriptor, *e.g.*, rotation change, affine change, and various deformations. Thus we first employ the proposed DIFT into the context of image matching.

A. Evaluation Protocol

Dataset. We follow the evaluation procedure proposed by Mikolajczyk and Schmid [37]. The code and dataset is downloaded from Oxford VGG website.[1] This dataset consists of real images with different geometric and photometric transformations (*e.g.*, viewpoint changes, image blur, illumination changes, and JPEG compression) and has the ground-truth matches through estimated homography. There are 6 groups of images in total corresponding to different transformations. In each group, one base image is cross matched with

[1] http://www.robots.ox.ac.uk/∼vgg/research/affine/

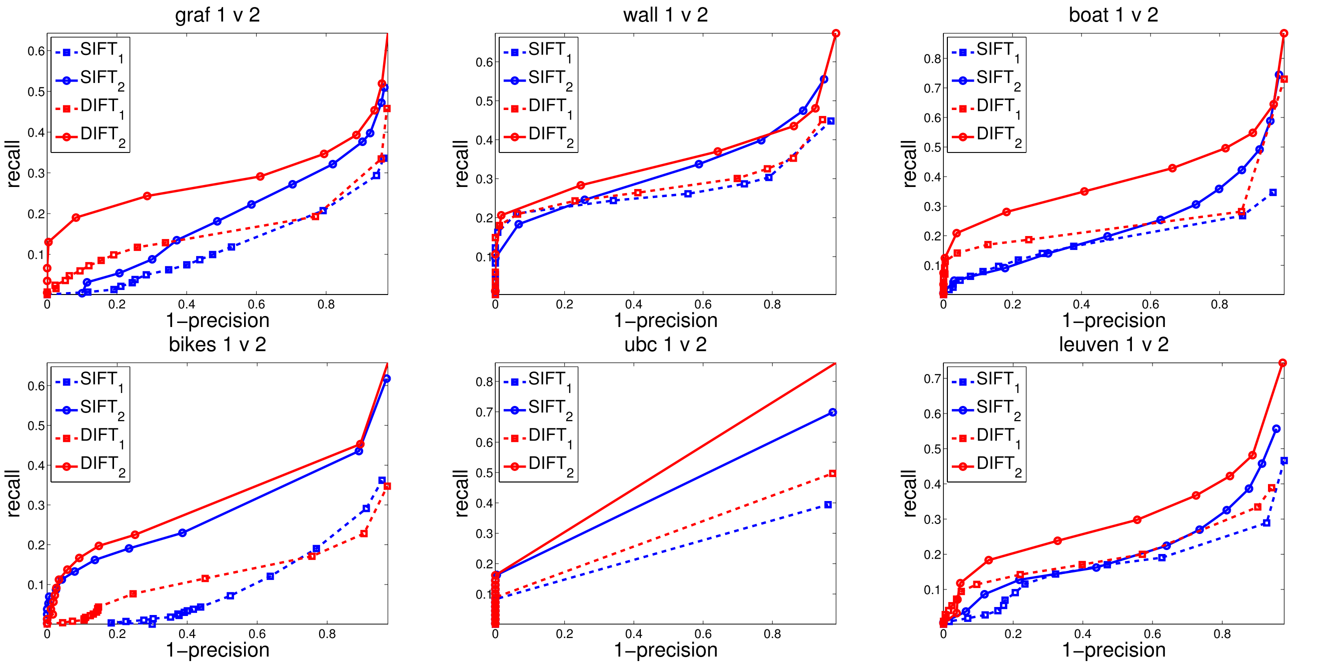


Fig. 6. Image matching assessment between DCT intrinsic orientation and SIFT multiple orientations. DIFT$_1$: DIFT with multiple orientations ϕ; DIFT$_2$: DIFT with the proposed DCT intrinsic orientation φ; SIFT$_1$: SIFT with multiple orientations ϕ; SIFT$_2$: SIFT with the proposed DCT intrinsic orientation φ.

other 5 images. We show the matching results between image 1 and 2, 1 and 3, 1 and 4, 1 and 5 and denote them by 1v2, 1v3, 1v4, 1v5. Fig.5 shows some examples.

1) Description: We use Hessian-affine detector [38] to detect the local region, and normalize it to a circle for feature description. Comparisons are conducted among DIFT, SIFT [32], GLOH (gradient location and orientation histogram) [37] and LIOP (local intensity order pattern) [61]. DIFT dimension is 32, SIFT and GLOH are 128; LIOP is 144. All descriptions use the same image patch.

2) Metric: Evaluation metric is adopted from [37], we choose the threshold-based matching. Given a base image and a transformed image in certain group, all pairs of keypoints in the two images are examined. For a particular pair of keypoints, if the Euclidean distance between their feature vectors falls below a threshold, this pair is regarded a match. Comparing the obtained matches with ground truth, we have *correct* and *false matches*. Define *recall* as ratio between the number of correct matches (*#correct matches*) and the number of ground truth correspondences (*#correspondences*):

$$recall = \frac{\#correct\ matches}{\#correspondences},$$

where ground truth *#correspondences* are obtained by using the original image transformation matrix and comparing the pixel coordinates.

The number of false matches relative to the total number of matches is defined as $1 - precision$:

$$1 - precision = \frac{\#false\ matches}{\#correct\ matches\ +\ \#false\ matches}.$$

Smaller $1 - precision$ and larger *recall* indicate a better performance.

B. Assessment on Feature Orientation

Fig.6 shows the assessment of feature orientation. We compare the DCT intrinsic orientation with the SIFT multiple orientations. Matching performance is conducted between the first and second image in each group. We report the result of two variants of DIFT: DIFT$_1$ (DIFT with multiple orientations ϕ) and DIFT$_2$ (DIFT with DCT intrinsic orientation φ) ; two variants of SIFT: SIFT$_1$ (SIFT with multiple orientations ϕ) and SIFT$_2$ (SIFT with DCT intrinsic orientation ϕ).

DIFT$_2$ with its DCT intrinsic orientation φ achieves the best matching performance over all the variants of DIFT and SIFT. In addition, clear improvement of SIFT$_2$ over SIFT$_1$ can be observed as well. The result is consistent with that shown in Table II. In the assessment of feature description, we use the original versions of DIFT and SIFT, which are DIFT$_2$ and SIFT$_1$.

C. Assessment on Feature Description

Fig.7 shows assessment of feature description. We compare DIFT with SIFT, GLOH and LIOP. Matching performance is conducted between the first and the rest images in each group.

1) Image Blur & JPEG Compression & Rotation + Scale: In the context of these three changes, DIFT clearly outperforms the other descriptions. Since either image blur or compression can be seen as a low-pass filter in the frequency domain, which is similar to the approach we have proposed in Sec.III-D. Thus, DIFT is robust enough to general image blur or JPEG compression. Moreover, it has been demonstrated in Fig.6 that the DCT intrinsic orientation is superior to the traditional gradient-based dominant orientation in rotation change.

2) Viewpoint: Results on viewpoint changes correspond to the images in Fig.5 (a) and (b): DIFT curves are

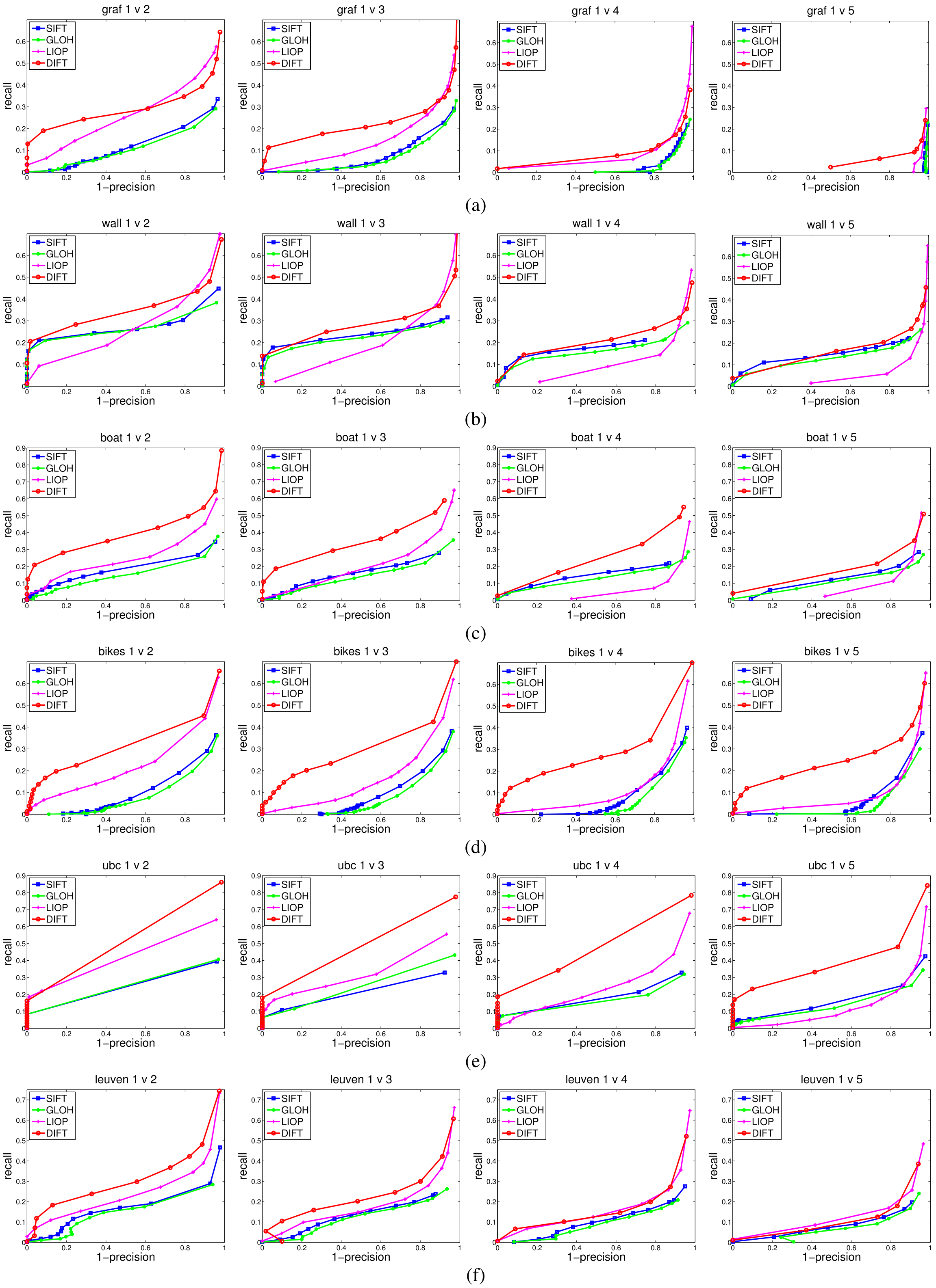


Fig. 7. Image matching assessment among DIFT and other representative descriptions, *i.e.*, SIFT, GLOH and LIOP. Result is presented with different image transformations. From left to right: 1v2 1v3 1v4 1v5. (a) viewpoint changes (graf). (b) viewpoint changes (wall). (c) rotation&scale changes (boat). (d) image blur (bikes). (e) JPEG compression (ubc). (f) illumination changes (leuven).

similar (slightly better) to others. Because this test is designed for evaluating the affine invariance, and the descriptors we use here are both extracted in the same local region, which is normalized by using the Hessian-affine detector.

3) Illumination: Illumination change is specifically tackled in DIFT by subtracting the minimal pixel value in each local region, and normalizing the DCT coefficient matrix. DIFT is thereby not affected by linear illumination changes.

TABLE I

COMPARISONS ON VLAD WITH DIMENSIONALITY SELECTION. PERFORMANCE ON HOLIDAYS IS MEASURED BY mAP. k IS THE VOCABULARY SIZE AND d IS DIFT DIMENSION. S_1 IS OUR SCHEME, S_2 CORRESPONDS TO LOW-FREQUENCY SELECTION, AND S_3 DOES RANDOM SELECTION

S_1	$k = 16$	$k = 64$	S_2	$k = 16$	$k = 64$	S_3	$k = 16$	$k = 64$
$d = 16$	0.361	0.421	$d = 16$	0.342	0.418	$d = 16$	0.295	0.326
$d = 24$	0.474	0.516	$d = 24$	0.469	0.505	$d = 24$	0.317	0.364
$d = \mathbf{32}$	**0.506**	**0.558**	$d = 32$	0.485	0.531	$d = 32$	0.349	0.429
$d = 40$	0.506	0.560	$d = 40$	0.492	0.548	$d = 40$	0.417	0.451
$d = 64$	0.507	0.560	$d = 64$	0.507	0.560	$d = 64$	0.507	0.560

TABLE II

COMPARISON OF THE PERFORMANCE BETWEEN DCT INTRINSIC ORIENTATION φ, SIFT DOMINANT (MULTIPLE) ORIENTATIONS ϕ AND SINGLE DOMINANT ORIENTATION ϕ_1. mAPS ARE REPORTED ON HOLIDAYS UTILIZING VLAD AND VOCABULARY SIZE $k = 64$. WE ALSO GIVE THE mAP OF DIFT WITH LARGER VOCABULARY $k = 256$, SO THAT ITS IMAGE REPRESENTATION D ON VLAD IS SIMILAR TO SIFT

Desc.	ϕ	ϕ_1	φ	$D = k \times d$	mAP	*fea.%*
SIFT	√			64×128	0.526	1.0
SIFT		√		64×128	0.512	0.587
SIFT			√	64×128	0.528	0.587
DIFT	√			64×32	0.554	1.0
DIFT		√		64×32	0.549	0.587
DIFT			√	64×32	0.558	0.587
DIFT			√	256×32	**0.598**	0.587

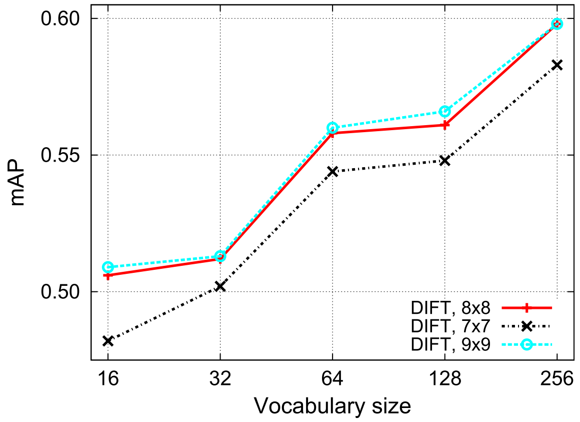


Fig. 8. Retrieval performance with different high-frequency cutoff positions in DIFT. The number followed by DIFT caption denotes the DIFT dimensions. Holidays dataset and VLAD model.

DIFT curves are apparently superior on the cases 1v2, 1v3, but tangled with LIOP on case 1v4 and 1v5. LIOP is formed by intensity order patterns which can handle the nonlinear illumination transformation very well.

Overall, DIFT outperforms SIFT, GLOH and LIOP in all cases and all transformations. Additionally, the dimensionality and the feature number per image of DIFT are much fewer than those of others, which makes its matching process much faster.

V. IMAGE RETRIEVAL

In this section, we apply the proposed DIFT in image retrieval task; it is a more complicated and difficult task compared to image matching. We shall present the experimental setup, the impact of parameters, and the overall comparison.

A. Experimental Setup

1) Dataset: We conduct experiments on standard retrieval benchmarks, namely Holidays [22], UKB [39], Oxford [43]. We investigate the impact of parameters on Holidays. To evaluate performance on a large scale, we also add Flickr1M [24] to Holidays as distractors.

2) Evaluation: SIFT and DIFT descriptors are extracted with the Hessian-affine detector using default parameters [36]. In SIFT multiple orientations are detected to increase robustness for the case when the dominant orientation is ambiguous. For instance on the Holidays and UKB datasets, DIFT produces much fewer features than SIFT due to the intrinsic orientation. Basically, there are 4.4M and 2.6M features for SIFT and DIFT on Holidays dataset, 19.4M and 11.2M correspondingly on UKB. We also use a lower threshold for the detector to yield a larger feature set for DIFT, which is comparable to the size of the SIFT feature set. This helps evaluate the performance at the same memory cost. We use -L to denote the lower threshold in the detector, *e.g.*, -L corresponds to threshold 300 and the default is 500 on Holidays. On the Oxford dataset, it has been shown that a single up-right orientation gives the best results; experiment on this dataset will therefore demonstrate the discriminative power of DIFT *v.s.* SIFT without the influence of the dominant orientation.

DCT matrix size is initially set to 41×41 and cutoff to 8×8 by default. Via dimensionality selection, we only keep 32 components as a 32-dimensional vector. For the DIFT description in the color space, DIFT^c, it is a 96-dimensional vector. The performance for the Holidays, Oxford and Flickr1M datasets is measured via the mean average precision (mAP) [21]. For the UKB dataset, the score is standardly computed as the average number of correct images in the top-4 positions (4-recall4), the best score is 4. We report the feature set size of DIFT *fea.%* as a ratio to SIFT feature set size, *e.g.* for Holidays, the denominator is 4.4M. We thus evaluate the memory-performance trade-off between mAP and *fea.%*

3) Retrieval Models: We conduct experiment on two representative retrieval models, BOW [25] and VLAD [43]. BOW employs local features, *i.e.*, SIFT and DIFT for image

TABLE III

RETRIEVAL RESULTS IN COMPARISON WITH THE STATE-OF-THE-ART. WE USE -L (*e.g.* DIFT-L) TO DENOTE THE LOWER THRESHOLD IN THE FEATURE DETECTOR. THERE ARE 4.4M SIFT FEATURES IN ORIGINAL HOLIDAYS DATASET AND 19.4M IN UKB. WE USE *fea.%* TO DENOTE THE RATIO OF FEATURE NUMBER FROM DIFT TO SIFT. $D = k \times d$. SIFT AND ROOTSIFT RESULTS ARE DIRECTLY REPORTED FROM REFERENCES [5], [25], AND [26]

VLAD[25]	$D = k \times d$	SSR[20]	IN[5]	Holiday	*fea.%*	UKB	*fea.%*
SIFT[24]	16×128			0.496	1.0	3.07	1.0
SIFT[24]	64×128			*0.526*	1.0	3.17	1.0
SIFT+PCA[25]	16×64	√		0.526	1.0	-	-
SIFT+PCA[25]	64×64	√		0.557	1.0	3.28	1.0
SIFT+PCA[25]	256×64	√		*0.587*	1.0	-	-
SIFT+PCA	256×32	√		0.576	1.0	-	-
rootSIFT[5]	256×128			0.592	1.0	-	-
rootSIFT[5]	256×128		√	0.614	1.0	-	-
rootSIFT[5]	256×128	√		*0.617*	1.0	-	-
DIFT	16×32			0.506	0.587	3.10	0.576
DIFT	64×32			0.558	0.587	3.23	0.576
DIFT	64×32	√		0.574	0.587	3.31	0.576
DIFT	256×32			0.598	0.587	3.31	0.576
DIFT	256×32		√	0.617	0.587	3.32	0.576
DIFT	256×32	√		**0.630**	0.587	**3.36**	0.576
DIFT^c	64×96	√		0.589	0.587	3.41	0.576
DIFT^c	256×96	√		**0.702**	0.587	**3.49**	0.576
DIFT-L	256×32			0.627	0.971	-	-
DIFT-L	256×32		√	0.638	0.917	-	-
DIFT-L	256×32	√		*0.645*	0.917	-	-
DIFT^c-L	256×96	√		*0.725*	0.917	-	-

TABLE IV

COMPARISONS OF DIFFERENT DESCRIPTORS ON BOW, HOLIDAYS DATASET

BOW	SIFT	DCT$_1$	DCT$_2$	DCT$_3$	DIFT	DIFT^c
k = 1000	0.401	0.230	0.056	0.370	**0.404**	**0.494**
k = 20000	0.437	0.248	0.147	0.402	**0.449**	**0.581**

TABLE V

A COMPARISON BETWEEN SIFT/ROOTSIFT AND U-DIFT/U-DIFT^C ON OXFORD5K DATASET

VLAD	$D = k \times d$	SSR	IN	Oxford
SIFT	64×64	√		0.304
rootSIFT	64×128	√		0.367
rootSIFT	256×128		√	0.389
rootSIFT	256×128	√		0.375
U-DIFT	64×32	√		0.321
U-DIFT	256×32	√		0.367
U-DIFT^c	64×96	√		0.385
U-DIFT^c	256×96			0.421
U-DIFT^c	256×96		√	0.424
U-DIFT^c	256×96	√		**0.466**

TABLE VI

THE COMPARISON BETWEEN DIFT^c AND THE PCA-COMPRESSED NEURAL CODES (128 DIMENSIONS). NEURAL CODES ARE TRAINED ON ILSVRC [46]

Descriptor	Holidays	UKB	Oxford
Neural codes [6]	0.747	3.42	0.433
DIFT^c	0.702	3.49	0.466

representation; VLAD aggregates local features into global representation. We compare DIFT and SIFT on both models.

B. Impact of Parameters

1) High-Frequency Cutoff: The cutoff position is set to 8×8 in Sec.III-D.1, as it retains the exact representation with SIFT. Here, we illustrate the cutoff impact in the retrieval context in Fig.3: we evaluate the performance using the 7×7, 8×8 and 9×9 cutoff positions, the corresponding dimensions are 49, 64, and 81. It can be seen that the 8×8 position performs almost the same as 9×9. Despite the fact the larger DCT matrix reflects more details of the local region, it increases the computational cost in the meantime; on the other hand, 7×7 is clearly inferior than 8×8 in terms of retrieval precision. Therefore, 8×8 achieves the best balance overall.

2) Dimensionality Selection: We show that the 64-dimensional descriptor can be further compressed via dimensionality selection. We compare three strategies S$_1$, S$_2$ and S$_3$. S$_1$ is our scheme, S$_2$ is low-frequency selection and S$_3$ is a random selection strategy. Table I shows that S$_1$ clearly outperforms the strategies of S$_2$ and S$_3$. Note that we carried out the random selection strategy (S$_3$) five times and report the average mAP. It is obvious that using 32 components in S$_1$ is of the best tradeoff between performance and efficiency. We keep this setting in the following experiment. It makes DIFT a 32-dimensional vector. Selected dimensions are illustrated in Fig.1.

3) DCT intrinsic Orientation φ: Referring to Sec.III-C, this section tests the effectiveness of DCT intrinsic orientation φ.

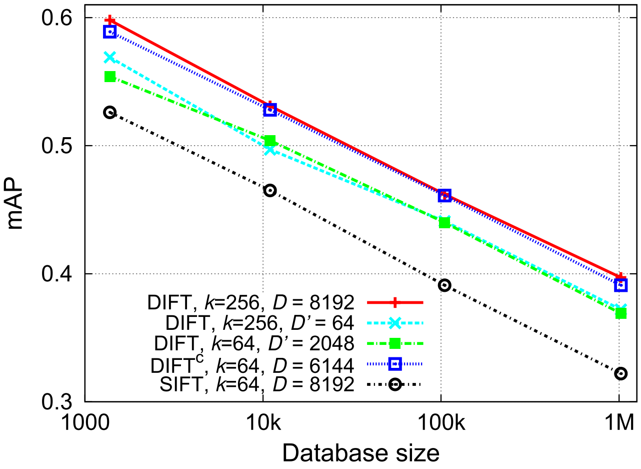


Fig. 9. mAP values of SIFT and DIFT (DIFT^C) corresponding to different sizes of Holidays + Flickr (1k, 10k, 100k, 1M). Baseline using SIFT on VLAD follows [25].

We compare it with the SIFT multiple orientations ϕ. We first apply ϕ as the dominant orientations for both SIFT and DIFT descriptions, then replace ϕ with φ and test again. Performance is measured by mAP, we also report *fea.%* which is related with memory overhead. The result shows that replacing ϕ with φ will slightly improve the mAP, *i.e.*, for SIFT (ϕ) and SIFT (φ), the mAPs are respective 0.526 and 0.528 ($k = 64$); for DIFT (ϕ) and DIFT (φ), the mAPs are respective 0.554 and 0.558 ($k = 64$). More importantly, the memory overhead ($fea.\%$) is significantly reduced to 58.7% by replacing ϕ with φ. Therefore, we claim that the DCT intrinsic orientation is mainly designed to reduce the memory overhead, and the DIFT description (Sec:III-D) is mainly designed to reduce the feature dimension and improve the precision in retrieval.

To clearly distinguish the DCT intrinsic orientation from the SIFT single dominant orientation, we also report the corresponding results of SIFT with single dominant orientation ϕ_1, the performance is no better than the multiple orientations.

C. Comparison on BOW

Table IV compares DIFT, SIFT and three other DCT-based descriptors: DCT$_1$ [40], DCT$_2$ [14] and DCT$_3$ [53] on BOW. To make a fair comparison, local regions are all extracted with the Hessian-affine detector. k is the vocabulary size.

DIFT outperforms SIFT and other DCT-based descriptors. The dimensions of DCT$_1$ and DCT$_2$ are quite low and therefore not representative; DCT$_3$ is like SIFT based on subregion division and it loses global information in the local region.

D. Comparison on VLAD

Table III compares our result on Holidays and UKB with other representative works. Retrieval model is VLAD. RootSIFT [4], SSR [21] and intra-norm (IN) [5] are also employed in VLAD to improve the performance. Local features are DIFT and SIFT, respectively.

Algorithm 1 Image Reconstruction by Utilizing DIFT

```
1. Extract DIFT descriptors from affine-invariant regions in image, plus $K^2$ DIFTs by dividing the whole image into $K \times K$ sub-regions.
2. Reconstruct a rough image using $K^2$ additional DIFTs for background;
2. Sort other DIFTs by their scales from large to small;
3. For each DIFT in sequence; reconstruct the local region via inverse DCT and stitch it on the reconstructed image; end.
```

So far, all the experiments we carry out use the same local region detector with SIFT, which means the memory overhead of our method is much lower than that of SIFT. To evaluate the performance at the same memory, we report DIFT-L (refer to Sec.V-A) results on Holidays in Table III, which has a comparable size 4.0M with SIFT 4.4M.

If we have a look at the results of SIFT and DIFT on Holidays, the performance of DIFT is superior to that of SIFT when using the same vocabulary size and even lower dimensionality, *e.g.*, $k = 64$, SIFT mAP is 0.526 and DIFT is 0.558. According to [26], applying PCA to reduce the SIFT dimensions to 64 will improve the mAP to 0.557 ($k = 64$) and 0.587 ($k = 256$). In the same setup in DIFT, we obtain mAP 0.630 with even lower dimension ($d = 32 < 64$). To compare with rootSIFT, its highest mAP (+SSR) is 0.617, which is close to our 0.630, nevertheless, the memory overhead of DIFT is much less than SIFT and rootSIFT, *i.e.*, the ratio of feature set size between DIFT and rootSIFT on Holidays is 0.587, in this sense DIFT is much superior considering the memory-performance trade-off. Particularly, query cost is directly related with *fea.%*, which reduces a lot in DIFT description.

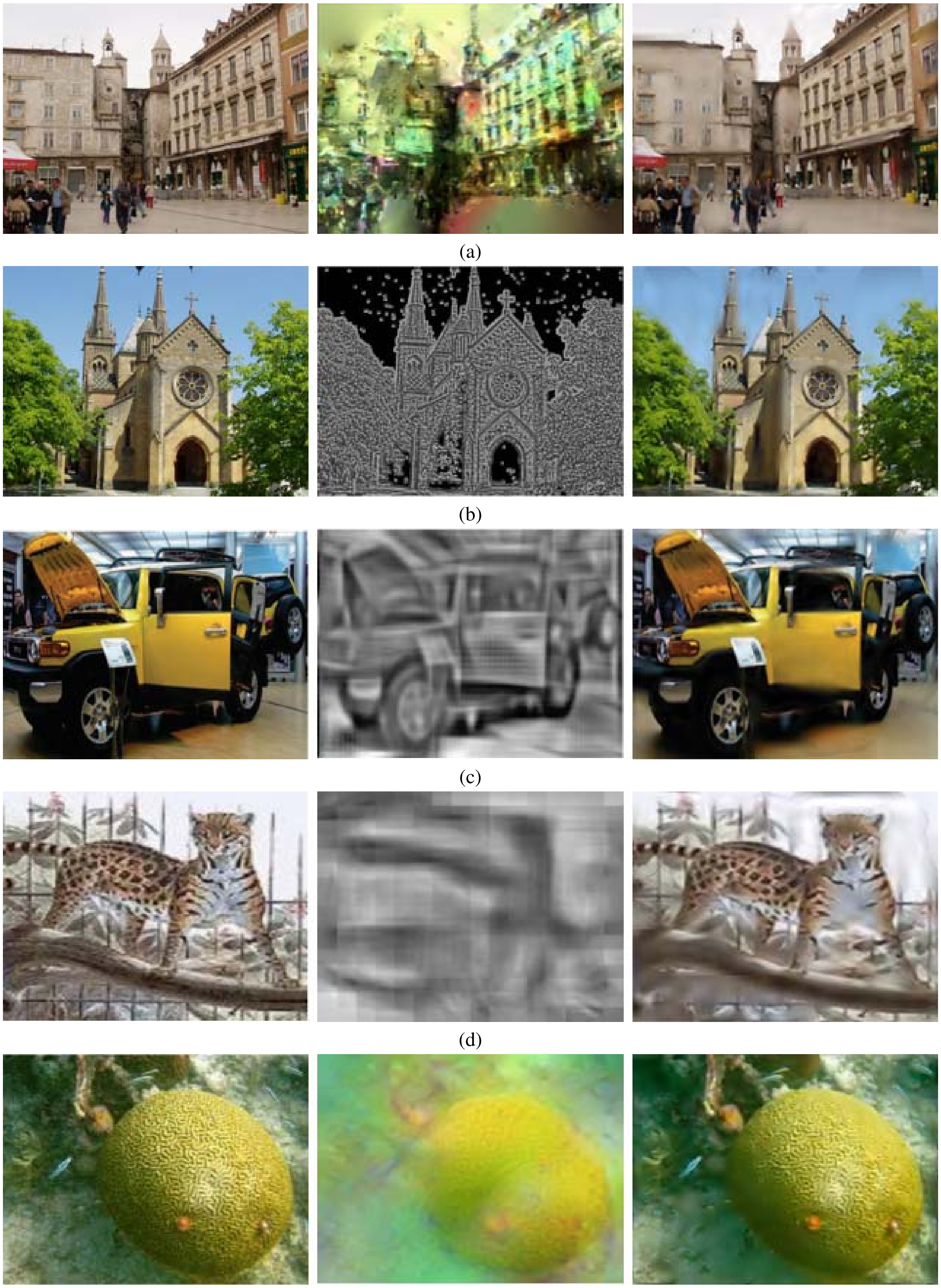
Moreover, it shows that with the similar *fea.%* to SIFT, the highest mAP of DIFT-L on VLAD + SSR reaches 0.645, which is the best. This is another possible way for the trade-off between memory and precision [48].

We also report the results of DIFT^C in color space. Specifically, it produces the highest mAP with 96-dimensional descriptor: DIFT^C + SSR: 0.702 and DIFT^C-L + SSR: 0.725 on Holidays dataset; DIFT^C + SSR: 3.49 on UKB dataset.

1) One Exceptional: it is known that, *e.g.*, on Oxford5k dataset, the single up-right orientation works best [22], [56]; neither multiple nor DCT intrinsic orientation is effective. In this situation, we simply adopt up-right orientation to every descriptor and denote DIFT/DIFT^C description by U-DIFT/U-DIFT^C. We compare it to SIFT on Oxford5k dataset in Table V. Considering both the precision and query cost, our schemes U-DIFT/U-DIFT^C yields better performance than SIFT/rootSIFT.

2) Large Scale: Fig.9 reports the mAP values obtained when we gradually add images from Flickr1M as distractors to Holidays dataset. Note VLAD dimension $D = k \times d$, where k is the vocabulary size, d is 128 for SIFT and 32 for DIFT. For larger database, the mAP of DIFT performs better. We suggest this behaviour as a result of the distinctiveness of DIFT from relevant images to irrelevant images.

Following the same setup in [25], if we apply PCA reduction to VLAD representation to keep only 64 dimensions of D,



(a)

(b)

(c)

(d)

(e)

Fig. 10. Comparisons on image reconstruction. From left to right are the original images, the reconstructed results in references and our results. (a) Original Image; Method in [62]: $\mathrm{PSNR}_1 = 11.66$; DIFT: $\mathrm{PSNR}_2 = 24.94$. (b) Original Image; Method in [11]: $\mathrm{PSNR}_1 = 10.82$; DIFT: $\mathrm{PSNR}_2 = 24.94$. (c) Original Image; Method in [59]: $\mathrm{PSNR}_1 = 8.91$; DIFT: $\mathrm{PSNR}_2 = 23.81$. (d) Original Image; Method in [27]: $\mathrm{PSNR}_1 = 11.58$; DIFT: $\mathrm{PSNR}_2 = 23.39$. (e) Original Image; Method in [33]: $\mathrm{PSNR}_1 = 12.13$; DIFT: $\mathrm{PSNR}_2 = 27.14$.

we find that DIFT performance ($D' = 64$) is still better than using SIFT ($D = 8192$). Notice that, to keep D the same between SIFT and DIFT, their vocabulary size is different due to the different d. Notwithstanding, we also give the result of DIFT by using the same vocabulary size k with SIFT, in terms of this, the dimensionality of DIFT is $D = 2048$, and it is still superior to SIFT. Performance on $\mathrm{DIFT}^{\mathsf{C}}$ is also presented. Be aware that the feature set size in DIFT is much smaller than in SIFT. Overall, DIFT is much more robust than SIFT on large-scale.

E. Discussion

Above all, we compared DIFT with SIFT and its variants *e.g.*, rootSIFT and PCA-SIFT. These feature descriptions are often conceptualized as handcrafted features. The reason behind their popularity is that handcrafted features do not rely on any labeled data and have very efficient training algorithms. The main problem of these methods, however, is that their modeling capacities are limited by the fixed transformations (filters) that stay the same for different sources of data [30].

Motivated by the success of CNNs [18], [29], researchers are currently working intensively towards developing CNN equivalents for learning visual features. Many accomplishments have been reported from using CNN features for a number of computer vision tasks [6], [9], [18], [29], [57], [60]. Particularly in the image retrieval task, we compare DIFT with one of the representative works [6]. Table VI reports the mAP result between DIFT^c and PCA-compressed neural codes from [6]. Neural codes are taken from the output of the second-last layer (FC6) of the CNN model proposed by Krizhevsky *et.al.* [29]. The CNN model is pre-trained for whole-image classification on ILSVRC [46]. This produces a 4096-dimensional feature vector for each image. The neural codes can be compressed via PCA to 128 dimensions almost without any quality loss; in contrast, the global representation in VLAD using DIFT^c is much higher, $D = 256 \times 96$. Notwithstanding, handcrafted features do not rely on any labeled data for training, and therefore can be very fast to extract. Particularly in the DIFT description, we have saved nearly half of the memory overhead on Holidays and UKB during the computation, which makes it even faster.

It should be noted that, with the rapid development of CNNs, many state of the art techniques, such as Fast-RCNN [17], [44], are proposed to speedup the training for deep features. One might obtain even superior retrieval performance by employing these fancy techniques; on the other hand, there are still a number of researchers devoted themselves into the work of handcrafted features [11], [59], [63]. Due to the clear physical meaning of its components, the handcrafted feature is still popularly used in a lot of computer vision tasks, *e.g.*, image reconstruction [11], visualization [59], and matching [63]. In the following section, we present another advantage of DIFT in image reconstruction.

VI. Image Reconstruction

It has been pointed out in [59] that the reconstruction capability is one of most important properties certain feature description should possess. It is particularly crucial for the generative detection and recognition tasks. Image features are usually extracted via nonlinear transforms from images (*e.g.*, local description and coding followed by spatial pooling). It is not straightforward to estimate the original image [41], [62]. Previous approaches have made some achievements in reconstructing images from local features, *e.g.*, SIFT [32] or HOG [59]. However, these traditional feature descriptions are based on gradient histograms and disregard most of the spatial information in the local region. The reconstruction result is therefore biased, *e.g.*, a human was restored from a dog's HOG descriptors [59]. In contrast, the proposed DIFT stores almost the complete spatial information in its description, so that we can directly utilize it to reconstruct the original image.

A. Procedure

We present the basic procedure to conduct image reconstruction from DIFT in Algorithm 1. Local patch can be reconstructed by reversing the DCT coefficients. To reconstruct the whole image, we follow the basic procedure in [62] that is to stitch coarse to fine from large scale patches to small scale. Since local patches are not dense enough to cover the entire image, [62] uses poisson image editing [31], [41] to refine the blank area, which is a time-consuming process and is not accurate at all. Considering the restoring attribute of DIFT, we divide the original image into $K \times K$ sub-regions and generate K^2 DIFTs. It turns out that they are able to retain the coarse information of the entire image including the blank area. For example, in Fig.10, we use $5 \times 5 = 25$ extra DIFTs together with original DIFTs to conduct the reconstruction.

The reconstruction procedure in Algorithm 1 is simple but quite effective and efficient. Neither external dataset nor learned dictionaries are required; the whole procedure simply operates as an inverse transform of DCT, which makes it superfast.

B. Result

Fig.10 illustrates some examples in comparison with [11], [27], [59], and [62]. Our results are clearly the best among them, particularly, we are able to obtain the color image from DIFT. We measure the performance in terms of PSNR (peak signal to noise ratio), it can be seen that our result yields the largest PSNR over all examples.

We also compare the reconstruction results between DIFT and CNN features [33]. As discussed in Sec.V-E, deep features are supposed to give an abstract sketch of an image/patch. It is hard to invert features to the original image. Fig.10(e) shows the reconstruction result. Our scheme clearly outperforms [33].

VII. Conclusion

This paper utilizes DCT to present DCT intrinsic orientation, and propose a new DCT inspired image feature transform description, DIFT. Each detected scale and affine invariant local region is rotated by the proposed DCT intrinsic orientation. Afterwards, each local region is described by the selected 32 elements in the DCT coefficient matrix to form a DIFT. Due to DCT intrinsic orientation, the amount of feature descriptions dramatically reduces, meanwhile, the dimensionality of DIFT is only of $\frac{1}{4}$ of SIFT, but DIFT achieves higher performance in benchmark retrieval.

On another side, as we know, JPEG for image compression is to encode DCT coefficients to reduce image data. In this paper, we have built a bridge between feature description for image retrieval and data coding for image compression since we can directly use DIFT to reconstruct an image. This natural consistency can help to merge the feature space and the image data space, and dramatically reduce the memory requirement simultaneously.

References

[1] A. E. Abdel-Hakim and A. A. Farag, "CSIFT: A SIFT descriptor with color invariant characteristics," in *Proc. CVPR*, 2006, pp. 1978–1983.

[2] N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete cosine transform," *IEEE Trans. Comput.*, vol. C-23, no. 1, pp. 90–93, Jan. 1974.

[3] T. Ahonen, J. Matas, C. He, and M. Pietikäinen, "Rotation invariant image description with local binary pattern histogram Fourier features," in *Proc. SCIA*, 2009, pp. 61–70.

[4] R. Arandjelović and A. Zisserman, "Three things everyone should know to improve object retrieval," in *Proc. CVPR*, 2012, pp. 2911–2918.

[5] R. Arandjelović and A. Zisserman, "All about VLAD," in *Proc. CVPR*, 2013, pp. 1578–1585.

[6] A. Babenko, A. Slesarev, A. Chigorin, and V. Lempitsky, "Neural codes for image retrieval," in *Proc. ECCV*, 2014, pp. 584–599.

[7] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *Comput. Vis. Image Understand.*, vol. 110, no. 3, pp. 346–359, 2008.

[8] S. Belongie, J. Malik, and J. Puzicha, "Shape context: A new descriptor for shape matching and object recognition," in *Proc. NIPS*, 2000, p. 3.

[9] H. Bilen, M. Pedersoli, and T. Tuytelaars, "Weakly supervised object detection with convex clustering," in *Proc. CVPR*, 2015, pp. 1081–1089.

[10] W. Chen, M. J. Er, and S. Wu, "Illumination compensation and normalization for robust face recognition using discrete cosine transform in logarithm domain," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 36, no. 2, pp. 458–466, Apr. 2006.

[11] E. d'Angelo, L. Jacques, A. Alahi, and P. Vandergheynst, "From bits to images: Inversion of local binary descriptors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 5, pp. 874–887, May 2014.

[12] B. Fan, F. Wu, and Z. Hu, "Aggregating gradient distributions into intensity orders: A novel local image descriptor," in *Proc. CVPR*, 2011, pp. 2377–2384.

[13] W. T. Freeman and E. H. Adelson, "The design and use of steerable filters," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 9, pp. 891–906, Sep. 1991.

[14] R. Fusek and E. Sojka, "Gradient-DCT (G-DCT) descriptors," in *Proc. IPTA*, 2014, pp. 1–6.

[15] Y. Gao, M. Shi, D. Tao, and C. Xu, "Database saliency for fast image retrieval," *IEEE Trans. Multimedia*, vol. 17, no. 3, pp. 359–369, Mar. 2015.

[16] S. Gauglitz, M. Turk, and T. Höllerer, "Improving keypoint orientation assignment," in *Proc. BMVC*, 2011, pp. 93.1–93.11.

[17] R. Girshick, "Fast R-CNN," in *Proc. ICCV*, 2015, pp. 1440–1448.

[18] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. CVPR*, 2014, pp. 580–587.

[19] C. Harris and M. Stephens, "A combined corner and edge detector," in *Proc. Alvey Vis. Conf.*, 1988, p. 50.

[20] D. Huang, C. Zhu, Y. Wang, and L. Chen, "Hsog: A novel local image descriptor based on histograms of the second-order gradients," *IEEE Trans. Image Process.*, vol. 23, no. 11, pp. 4680–4695, Nov. 2014.

[21] H. Jégou and O. Chum, "Negative evidences and co-occurences in image retrieval: The benefit of PCA and whitening," in *Proc. ECCV*, 2012, pp. 1–14.

[22] H. Jégou, M. Douze, and C. Schmid, "Hamming embedding and weak geometric consistency for large scale image search," in *Proc. ECCV*, 2008, pp. 304–317.

[23] H. Jégou, M. Douze, and C. Schmid, "On the burstiness of visual elements," in *Proc. CVPR*, 2009, pp. 1169–1176.

[24] H. Jégou, M. Douze, and C. Schmid, "Packing bag-of-features," in *Proc. CVPR*, 2009, pp. 2357–2364.

[25] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," in *Proc. CVPR*, 2010, pp. 3304–3311.

[26] H. Jégou, F. Perronnin, M. Douze, J. Sánchez, P. Pérez, and C. Schmid, "Aggregating local image descriptors into compact codes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 9, pp. 1704–1716, Sep. 2012.

[27] H. Kato and T. Harada, "Image reconstruction from bag-of-visual-words," in *Proc. CVPR*, 2014, pp. 955–962.

[28] Y. Ke and R. Sukthankar, "PCA-SIFT: A more distinctive representation for local image descriptors," in *Proc. CVPR*, 2004, pp. 506–513.

[29] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. NIPS*, 2012, pp. 1097–1105.

[30] Z. Lan, S.-I. Yu, M. Lin, B. Raj, and A. G. Hauptmann. (2015). "Handcrafted local features are convolutional neural networks." [Online]. Available: http://arxiv.org/abs/1511.05045

[31] S. Lefkimmiatis and M. Unser, "Poisson image reconstruction with Hessian Schatten-norm regularization," *IEEE Trans. Image Process.*, vol. 22, no. 11, pp. 4314–4327, Nov. 2013.

[32] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[33] A. Mahendran and A. Vedaldi, "Understanding deep image representations by inverting them," in *Proc. CVPR*, 2015, pp. 5188–5196.

[34] K. Mikolajczyk and C. Schmid, "Indexing based on scale invariant interest points," in *Proc. ICCV*, 2001, pp. 525–531.

[35] K. Mikolajczyk and C. Schmid, "An affine invariant interest point detector," in *Proc. ECCV*, 2002, pp. 128–142.

[36] K. Mikolajczyk and C. Schmid, "Scale & affine invariant interest point detectors," *Int. J. Comput. Vis.*, vol. 60, no. 1, pp. 63–86, 2004.

[37] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 10, pp. 1615–1630, Oct. 2005.

[38] K. Mikolajczyk *et al.*, "A comparison of affine region detectors," *Int. J. Comput. Vis.*, vol. 65, no. 1, pp. 43–72, 2005.

[39] D. Nister and H. Stewenius, "Scalable recognition with a vocabulary tree," in *Proc. CVPR*, 2006, pp. 2161–2168.

[40] S. Obdržálek and J. Matas, "Image retrieval using local compact DCT-based representation," in *Proc. Joint Pattern Recognit. Symp.*, 2003, pp. 490–497.

[41] P. Pérez, M. Gangnet, and A. Blake, "Poisson image editing," *ACM Trans. Graph.*, vol. 22, no. 3, pp. 313–318, Jul. 2003.

[42] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," in *Proc. ECCV*, 2010, pp. 143–156.

[43] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in *Proc. CVPR*, 2007, pp. 1–8.

[44] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. NIPS*, 2015, pp. 91–99.

[45] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "Orb: An efficient alternative to SIFT or SURF," in *Proc. ICCV*, 2011, pp. 2564–2571.

[46] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.

[47] B. Shen and I. K. Sethi, "Direct feature extraction from compressed images," in *Proc. SPIE, Electron. Imag., Sci. Technol.*, vol. 2670. Jan. 1996, pp. 1–12.

[48] M. Shi, Y. Avrithis, and H. Jégou, "Early burst detection for memory-efficient image retrieval," in *Proc. CVPR*, 2015, pp. 605–613.

[49] M. Shi, T. Furon, and H. Jégou, "A group testing framework for similarity search in high-dimensional spaces," in *Proc. 22nd ACM MM*, 2014, pp. 407–416.

[50] M. Shi, X. Sun, D. Tao, and C. Xu, "Exploiting visual word co-occurrence for image retrieval," in *Proc. 20th ACM MM*, 2012, pp. 69–78.

[51] M. Shi, X. Sun, D. Tao, C. Xu, G. Baciu, and H. Liu, "Exploring spatial correlation for visual object retrieval," *ACM Trans. Intell. Syst. Technol.*, vol. 6, no. 2, 2015, Art. no. 24.

[52] M. Shi, R. Xu, D. Tao, and C. Xu, "W-tree indexing for fast visual word generation," *IEEE Trans. Image Process.*, vol. 22, no. 3, pp. 1209–1222, Mar. 2013.

[53] T. Song and H. Li, "Local polar DCT features for image description," *IEEE Signal Process. Lett.*, vol. 20, no. 1, pp. 59–62, Jan. 2013.

[54] F. Tang, S. H. Lim, N. L. Chang, and H. Tao, "A novel feature descriptor invariant to complex brightness changes," in *Proc. CVPR*, 2009, pp. 2631–2638.

[55] G. Tolias, Y. Avrithis, and H. Jégou, "To aggregate or not to aggregate: Selective match kernels for image search," in *Proc. ICCV*, 2013, pp. 1401–1408.

[56] G. Tolias, T. Furon, and H. Jégou, "Orientation covariant aggregation of local descriptors with embeddings," in *Proc. ECCV*, 2014, pp. 382–397.

[57] G. Tolias, R. Sicre, and H. Jégou, "Particular object retrieval with integral max-pooling of CNN activations," in *Proc. ICLR*, 2016, pp. 1–12.

[58] T. Tuytelaars and K. Mikolajczyk, "Local invariant feature detectors: A survey," *Fundations Trends Comput. Graph. Vis.*, vol. 3, no. 3, pp. 177–280, Jan. 2008.

[59] C. Vondrick, A. Khosla, T. Malisiewicz, and A. Torralba, "HOGgles: Visualizing object detection features," in *Proc. ICCV*, 2013, pp. 1–8.

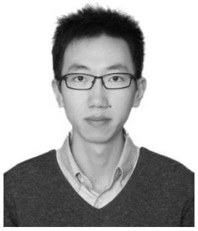
[60] C. Wang, K. Huang, W. Ren, J. Zhang, and S. Maybank, "Large-scale weakly supervised object localization via latent category learning," *IEEE Trans. Image Process.*, vol. 24, no. 4, pp. 1371–1385, Apr. 2015.

[61] Z. Wang, B. Fan, and F. Wu, "Local intensity order pattern for feature description," in *Proc. ICCV*, 2011, pp. 603–610.

[62] P. Weinzaepfel, H. Jégou, and P. Pérez, "Reconstructing an image from its local descriptors," in *Proc. CVPR*, 2011, pp. 337–344.

[63] H. Yang and I. Patras, "Mirror, mirror on the wall, tell me, is the error small?" in *Proc. CVPR*, 2015, pp. 4685–4693.

Yunhe Wang received the B.E. degree from Xidian University in 2013. He is currently pursuing the Ph.D. degree with the Key Laboratory of Machine Perception (Ministry of Education), Peking University. His research interests lie primarily in machine learning and computer vision.



Miaojing Shi is a Post-Doctoral Researcher with the University of Edinburgh. He received the Ph.D. degree from Peking University in 2015. He was a recognized student with the University of Oxford from 2012 to 2013, and a Visiting Student with INRIA Rennes from 2014 to 2015. His research interests include visual search and computer vision. He has authored or co-authored 20 publications.

Shan You received the B.E. degree from Xi'an Jiaotong University in 2014. He is currently pursuing the Ph.D. degree with the Key Laboratory of Machine Perception (Ministry of Education), Peking University. His research interests lie primarily in machine learning and computer vision.

Chao Xu received the B.E. degree from Tsinghua University in 1988, the M.S. degree from the University of Science and Technology of China in 1991, and the Ph.D. degree from the Institute of Electronics, Chinese Academy of Sciences, in 1997. From 1991 to 1994, he was an Assistant Researcher by the University of Science and Technology of China. Since 1997, he has been with the Key Laboratory of Machine Perception (Ministry of Education), Peking University, where he has been a Professor since 2005. His research interests are in image and video processing, multimedia technology. He has authored or co-authored more than 120 publications and holds eight patents in these fields.