Image and Vision Computing 89 (2019) 88–94

Contents lists available at ScienceDirect

Image and Vision Computing

journal homepage: www.elsevier.com/locate/imavis

Diffusion-based kernel matrix model for face liveness detection ☆

Changyong Yu[1], Chengtang Yao[1], Mingtao Pei*, Yunde Jia

Beijing Laboratory of Intelligent Information Technology, School of Computer Science, Beijing Institute of Technology, Beijing 100081, PR China

ARTICLE INFO

Article history:
Received 11 February 2018
Received in revised form 30 March 2019
Accepted 27 June 2019
Available online 3 July 2019

Keywords:
Face liveness detection
Anisotropic diffusion
Kernel matrix model
DK feature

ABSTRACT

Face recognition and verification systems are vulnerable to video spoofing attacks. In this paper, we present a diffusion-based kernel matrix model for face liveness detection. We use the anisotropic diffusion to enhance the edges of each frame in a video, and the kernel matrix model to extract the video features which we call the diffusion kernel (DK) features. The DK features reflect the inner correlation of the face images in the video. We employ a generalized multiple kernel learning method to fuse the DK features and the deep features extracted from convolution neural networks to achieve better performance. Our experimental evaluation on two publicly available datasets shows that the proposed method outperforms the state-of-art face liveness detection methods.

1. Introduction

Face recognition and verification [9, 29, 31, 35] has become the most popular technology in high-level security systems with a wide range of applications owing to its natural, intuitive, and less human-invasive face biometrics. Face biometrics is vulnerable to spoofing attacks by using photographs, videos or 3D masks of an actual user. Video spoofing is one of the most common methods of attacking face recognition systems as videos contain not only face images of a valid user, but also facial gestures like eye blinking of a valid user. Many researchers have made much effort to face liveness detection based on image quality [32, 38], spectrum [14, 39], motion information [16, 24], and head pose [4]. These methods extract appropriate features or spectrums to reflect the differences between live and fake faces. Recently, diffusion methods have been applied to face liveness detection [1, 13] to estimate the differences between the live and fake face surface properties and achieve spectacular performance. These methods focus on extracting features from a single image for face liveness detection, without considering the inner correlations between face frames in a video.

In this paper, we present a diffusion-based kernel matrix model for face liveness detection to prevent video spoofing attacks. We use the anisotropic diffusion to enhance the edges of the face images in a video, and the kernel matrix model to extract face features from the video. We call these features the diffusion kernel (DK) features. The DK features reflect the fuzzy degree of the surface and the inner nonlinear correlation of a sequential face frames in temporal dimension. To achieve better performance, we also extract deep features using deep convolution neural networks, and exploit a generalized multiple kernel learning method to fuse the DK features and the deep features. The deep features work well with DK features by providing appearance information. Our method achieves an impressive accuracy on publicly available datasets and outperforms the state-of-art face liveness detection methods. Fig. 1 illustrates the overview of our method.

The rest of the paper is organized as follows: we review the related work in Section 2. In Section 3, we present the diffusion-based kernel matrix model, and introduce the generalized multiple kernel learning method to fuse DK features and deep features. We experimentally evaluate the proposed method and its parameters on several datasets in Section 4, and conclude our work in Section 5.

2. Related work

Many face liveness detection methods have been reported based on facial action, aiming at detecting subconscious response of a face. Given an image sequence, these methods attempt to capture facial response like eye blinking, mouth movement, and head pose for face liveness detection. Pan et al. [24] presented eye blinking behavior detection using a unidirectional conditional graphic framework. A discriminative measure of eye states is also incorporated into the framework to detect face liveness. Singh et al. [30] used a Haar classifier and distinguished fake faces from the real faces by detecting eye and mouth movements. Anjos et al. [3] proposed to detect motion

☆ This paper has been recommended for acceptance by Sinisa Todorovic.

* Corresponding author.
E-mail address: peimt@bit.edu.cn (M. Pei).

[1] These authors equally contributed to this work.

https://doi.org/10.1016/j.imavis.2019.06.009
0262-8856/

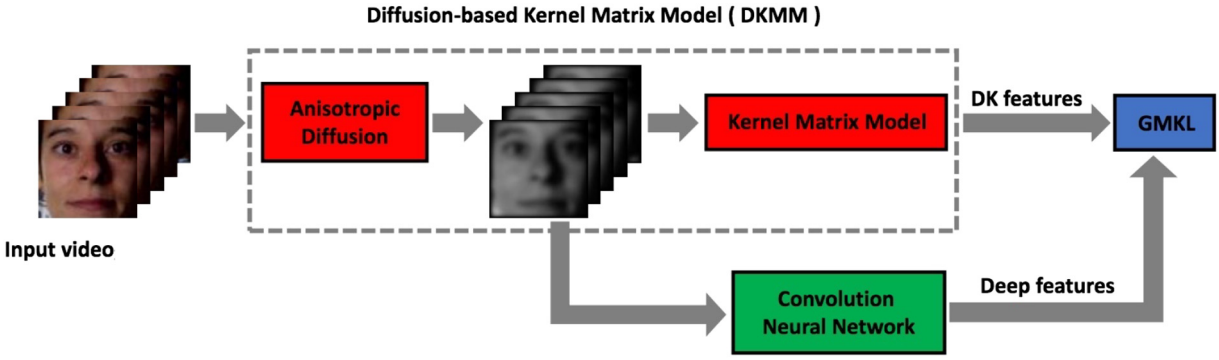

Fig. 1. An illustration of the proposed method. We diffuse the input video by anisotropic diffusion method first, and use kernel matrix model to extract the diffusion kernel (DK) features. Then, we extract deep features from the diffused images by deep convolution neural networks. Finally, the DK features and deep features are fused by a generalized multiple kernel learning (GMKL) method for face liveness detection.

correlations between the users head and the background regions obtained from the optical flow to indicate a spoofing attack. Tirunagari et al. [33] employed the dynamic mode decomposition (DMD) algorithm to capture a face from a video and extract dynamic visual information to detect the spoofing attack. Bharadwaj et al. [6] proposed a face liveness detection method which uses the configuration LBP and motion estimation to extract the facial features. Kollreider et al. [15] used a lightweight novel optical flow to estimate face motion based on the structure tensor. These methods aim to detect subconscious response of a live face. Since fake faces in video replay attack also have facial response like eye blinking and mouth movement, sometimes detecting facial response cannot distinguish fake faces from real ones. Besides, some dynamic methods require user to follow some instructions which is inconvenient for the user. Different from these works, we learn a discriminative model to extract discriminative features that reflect the differences between live and fake faces.

There are more face liveness detection methods based on face image analysis. These methods assume that fake faces tend to lose more information by the imaging system and thus come into a lower quality image under the same capturing condition. Li et al. [20] analyzed the coefficients of Fourier transform since the reflections of light on 2D and 3D surfaces result in different frequency distributions. For example, fake faces are mostly captured twice by the camera, so their high-frequency components are different with those of real faces. Zhang et al. [38] extracted frequency information using multiple DoG filters to detect the liveness of the captured face image. Tan et al. [32] and Peixoto et al. [25] used the combination of a DoG filter and other models to extract efficient features from a face image to improve liveness detection performance. Maatta et al. [21] extracted the micro texture face images using the multi-scale local binary pattern (LBP), and used SVM classifier to perform the face liveness detection. Peng et al. [10] used a guided scale space to reduce the redundancy of the original facial texture and to extract more powerful facial edges. Boulkenafet et al. [7] reported the extraction of speed-up features in different color spaces and represented the facial appearance by Fisher vector encoding on the extracted features. Kim et al. [13] calculated the diffusion speed of a single image, then utilized a local speed model to extract features and a linear SVM classifier to distinguish the fake faces from real ones. Alotaibi and Mahmood [1] used the nonlinear diffusion to detect edges in the input image and convolution neural networks to detect face liveness in the diffused image. These methods estimate the differences between live faces and fake faces by extracting discriminative features to improve detection performance. Similar to these methods, we also extract discriminative features in the diffused images for face liveness detection. The difference is that our work focuses on preserving useful edge information and extracting the inner nonlinear correlation between live and fake face images.

3. Proposed method

We use the anisotropic diffusion to enhance the edges of each face image in a video. Some examples of diffused images are shown in Fig. 2. We use the kernel matrix model to extract the diffusion kernel (DK) features. To achieve better performance against the spoofing attacks, we extract the deep features by deep convolution neural networks, and exploit a generalized multiple kernel learning method to fuse the DK features and the deep features.

3.1. Anisotropic diffusion

Consider the anisotropic diffusion equation [27]:

$$I_t = div(c(x,y,t) \bigtriangledown I) = c(x,y,t)\Delta I + \bigtriangledown c \cdot \bigtriangledown I, \quad (1)$$

where the *div* represents the divergence operator, $\bigtriangledown$ is the gradient, and Δ is the Laplacian operator. If $c(x,y,t)$ is a constant, Eq. (1) reduces to the isotropic heat diffusion equation $I_t = c\Delta I$. Suppose

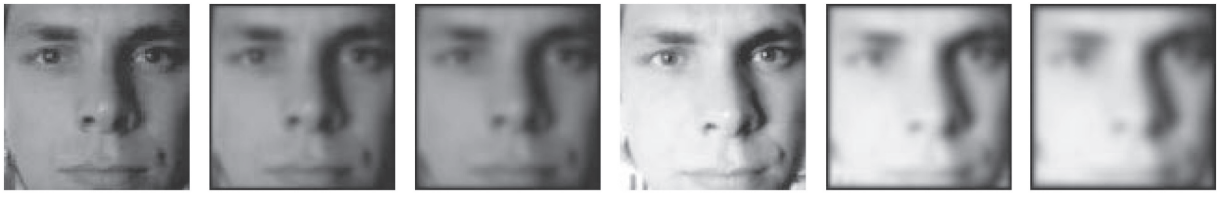

Fig. 2. Examples of diffused images with different iteration numbers (L). (a), (b) and (c) are live face images, (d), (e) and (f) are fake face images. (a) and (d) Original images. (b) and (e) Diffused image with $L = 5$. (c) and (f) Diffused image with $L = 15$.

the locations of the region boundaries or edges for that scale are known at time t, setting the conduction coefficient to be 1 inside each region and 0 at the boundaries can encourage a region to be smooth other than smoothing across the boundaries. The blurring would only occur in each region and the boundaries would remain sharp.

Perona and Malik [27] presented the best estimate of boundary locations appropriate to a scale for localizing the region boundaries at the scale. Let $E(x,y,t)$ be such an estimate, the conduction coefficient $c(x,y,t)$ is chosen to be a function $c = g(||E||)$ of the magnitude of E. $E = 0$ means the points are in the interior of a region and in other cases means the points are at the edge. Specially, if the function $g(.)$ is chosen properly, the diffusion will preserve and sharpen the brightness edges, where the conduction coefficient is chosen locally as a function of the magnitude of the gradient of the brightness function:

$$c(x,y,t) = g(||\bigtriangledown I(x,y,t)||) \tag{2}$$

3.1.1. Edge enhancement

The blurring of edges is the main cost to pay for eliminating the noise with conventional low-pass filtering and linear diffusion. The high-pass filtering or the diffusion equation backwards in time is used to perform the edge enhancement and reconstruction of blurry images.

The expression for the divergence operator is simplified to

$$div(c(x,y,t)\bigtriangledown I) = \frac{\partial}{\partial x}(c(x,y,t)I_x), \tag{3}$$

where c is a function of the gradient of I as in Eq. (2).

3.1.2. Diffusion scheme

Following the numerical scheme of anisotropic diffusion [27], Eq. (1) is discretized on a square lattice with brightness values associated to the vertices, and conduction coefficients to the arcs. A 4-nearest neighbors discretization of the Laplacian operator is given by

$$\begin{aligned} I_{i,j}^{L+1} =& I_{i,j}^{L} + \lambda[c_N \cdot \bigtriangledown_N I + c_S \cdot \bigtriangledown_S I \\ &+ c_E \cdot \bigtriangledown_E I + c_W \cdot \bigtriangledown_W I]_{i,j}^{L} \end{aligned} \tag{4}$$

where $0 \le \lambda \le 1/4$ for the numerical scheme to be stable, N, S, E, W represent North, South, East and West, respectively. is the nearest-neighbor differences:

$$\begin{aligned} \bigtriangledown_N I_{i,j} &\equiv I_{i-1,j} - I_{i,j} \\ \bigtriangledown_S I_{i,j} &\equiv I_{i+1,j} - I_{i,j} \\ \bigtriangledown_E I_{i,j} &\equiv I_{i,j+1} - I_{i,j} \\ \bigtriangledown_W I_{i,j} &\equiv I_{i,j-1} - I_{i,j}. \end{aligned} \tag{5}$$

If the difference in a direction is significant, it indicates that this point may be at an edge and we should preserve and sharpen the edge information. Fig. 3 shows the illustration of detecting edges and preserving them. If a point is at an edge, we will preserve and sharpen it, otherwise, the point will be blurred.

The conduction coefficients of the gradient can be computed by

$$\begin{aligned} c_{N_{i,j}}^{L} &= g\left(\left|\bigtriangledown_N I_{i,j}^{L}\right|\right) \\ c_{S_{i,j}}^{L} &= g\left(\left|\bigtriangledown_S I_{i,j}^{L}\right|\right) \\ c_{E_{i,j}}^{L} &= g\left(\left|\bigtriangledown_E I_{i,j}^{L}\right|\right) \\ c_{W_{i,j}}^{L} &= g\left(\left|\bigtriangledown_W I_{i,j}^{L}\right|\right). \end{aligned} \tag{6}$$

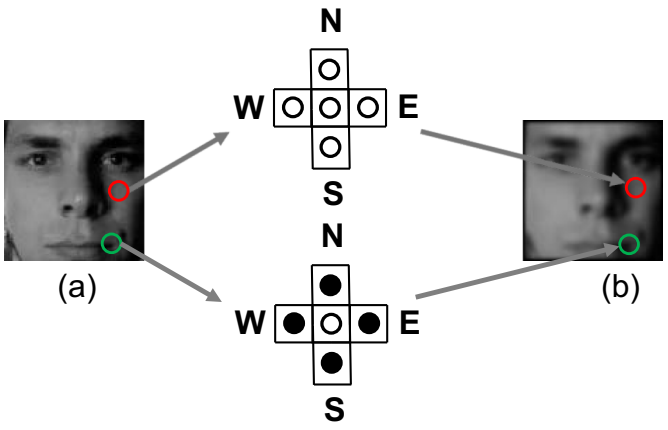

Fig. 3. An illustration of detecting and preserving edges. (a) Original image. (b) Diffused image. A point in the red circle is similar with its nearest-neighbors that indicates this point is in the interior of a region. A point in the green circle has distinct differences with its nearest-neighbors that indicates this point is at the edge.

If the in a direction changes greater, the value of the conduction coefficient should be smaller, and thus will preserve and sharpen the edges.

The conduction tensor in the diffusion equation is diagonal with entries $g(|I_x|)$ and $g(|I_y|)$ instead of $g(||\bigtriangledown I||)$ and $g(||\bigtriangledown I||)$. The diffusion scheme preserves the property of the continuous Eq. (1) that the total amount of brightness in the image is preserved.

3.2. Kernel matrix

The kernel matrix M can be used as a generic feature representation [36]. For a large set of kernel functions, the kernel matrix is guaranteed to be nonsingular, even if samples are scarce, and can model nonlinear feature relationship efficiently.

The entry $k_{i,j}$ of a kernel matrix M is defined as

$$k_{i,j} = < \phi(f_i), \phi(f_j) > = \kappa(f_i, f_j), \tag{7}$$

where $\phi(\cdot)$ is an implicit nonlinear mapping and $\kappa(\cdot,\cdot)$ is the induced kernel function. The mapping $\kappa(\cdot,\cdot)$ is applied to each feature f_i other than to each sample x_i.

The similarity of feature distributions can be evaluated by some specific kernels like Bhattacharyya kernel [17]. When we are not sure about the nonlinear relationship, we can apply a kernel representation, such as the Gaussian RBF kernel

$$\kappa(f_i, f_j) = exp(-\gamma||f_i - f_j||^2) \tag{8}$$

The kernel matrix also has its advantages in dealing with the singularity problem. When the number of the features is greater than or equal to the dimension of them, some feature representations like covariance matrix are bound to be singular, but kernel matrix can handle this situation well. According to Micchellis Theorem [12], let $f_1, f_2, \cdots, f_d$ be a set of different n-dimensional vectors. The matrix $M_{d\times d}$ computed by a RBF kernel $\kappa(f_i,f_j) = exp(-\gamma||f_i - f_j||^2)$ is guaranteed to be nonsingular, no matter what values d and n are. The RBF kernel also satisfies the above theorem to ensure the non-singularity of DK features. The non-singularity guarantees the validity of the DK features.

To reduce the computational complexity, we use the RBF kernel as the kernel of the matrix for its superior properties. Given n d-dimensional vectors, $x_1, \cdots, x_n$, computing all the entries

$||f_i - f_j||^2 (i,j = 1,\cdots,d)$ has the complexity of $O(nd^2)$. In addition, the proposed kernel representation based on the RBF kernel could be quickly given by using integral images. Note that $||f_i - f_j||^2 = f_i^\top f_i - 2f_i^\top f_j + f_j^\top f_j$, and d^2 integral images is precomputed for the inner product of any two feature dimensions.

Generally, kernel evaluation is more reliable with more samples. In the RBF kernel function, more samples make the parameters converge towards their true values. In practice, however, the number of available training samples is limited. Also, the size of the proposed kernel matrix is fixed ($d \times d$) and independent of the number of samples (n) in a set. Therefore, the kernel-based representations obtained from two different-sized sets can be compared directly.

3.3. Diffusion kernel (DK) features

The diffusion-based kernel matrix model includes two processes. First, given a face video C_o, the anisotropic diffusion is used to diffuse each frame of the video to enhance the edges. After several diffusion iterations, the edge will be preserved and become sharper.

Next, we extract DK features from the diffused video C_{dif}. As we use the RBF kernel as the kernel function, the model is defined as

$$f = \kappa(C_{dif}, C_{dif}) = exp(-\gamma||C_{dif} - C_{dif}||^2). \tag{9}$$

We vectorize pixel values of each frame of C_{dif} as a column vector, and the video C_{dif} is represented as a matrix $M_{d\times n}$, where d is the dimension of the images and n is the number of the frames.

The dimension d of the matrix is so high and will cost huge computing resource and time, thus, we use the PCA (Principal Component Analysis) to reduce the dimension of each frame of C_{dif}. After the dimensionality reduction, the matrix $M_{d\times n}$ which represents the diffused face video clip C_{dif} becomes $M_{lowd\times n}$. We input the low dimension matrix as the representation of C_{dif} into the model and gain a $M_{lowd\times lowd}$, that is, the DK feature. We use the DK features to distinguish fake face images from real one effectively since they reflect the inner correlation of sequential face images like C_{dif}.

3.4. Deep features

We use Convolution Neural Network (CNN) to extract the local features by combining three architectural concepts that perform some degree of shift, scale, distortion invariance, local receptive fields, shared weight and subsampling. The ability of both convolution layers and subsampling layers to learn distinctive features from the diffused image helps to extract features and achieve better performance for face liveness detection.

The pre-trained model we used is the AlexNet [18] for its impressive performance. The AlexNet contains convolutional layers, normalization layers, linear layers, ReLU activation layers, and max-pooling layers. For simplicity, we use L1-5 to represent the 5 convolutional layers, and L6-8 describe the 3 linear layers. The L3-5 are connected to one another without any intervening pooling or normalization layers. The fully-connected layers L6-8 have 4096 neurons each. The L6-7 output features with the dimension of 4096, and the dimensionality of features in L8 is 1000. The L8 is followed by a softmatx classifier to generate probability distribution for classification. Previous studies [18, 23] show that the 4096-dimensional features of L7 perform better than many handcrafted features. In our network, the L1-7 layers are used as the feature extractor, and we use the 4096-dimensional features of L7 as the deep features.

As mentioned above, the input is a video C_o, we randomly select a frame to represent the whole video clip. We assume that the deep feature of this frame can represent the deep feature of input video C_o.

3.5. Generalized multiple kernel learning

Multiple kernel learning is to learn an optimal linear/non-linear combination of a predefined set of kernels. It selects the optimal kernel and parameters from a larger set of kernels to reduce bias.

Given one kind of features $\{x_i|i = 1,2,\cdots,N\}$ and the other kind of features $\{y_i|i = 1,2,\cdots,N\}$, we fuse these two kinds of features and train multiple binary classifiers by multiple kernel learning. The decision function is given by

$$f(x,y) = c_1 w_1^T \varphi_1(x) + c_2 w_2^T \varphi(y) + b, \tag{10}$$

where c_1 and c_2 are the combination coefficients of the two kinds of features with the constraints that $c_1 + c_2 = 1$ and $c_1, c_2 \geq 0$, w and b are parameters of the standard SVM, and $\varphi(\cdot)$ is a function to map those two kinds of features to high dimensional space. c, w and b are learned by solving

$$\begin{aligned} &\min_{w,b,c} \frac{1}{2}\sum_{t=1}^{2}\left(c_t||w_t||^2 + c_t^2\right) + C\sum_{i}^{N} l(p_i, f(x_i,y_i)) \\ &s.t. \quad c_1 + c_2 = 1, c_1, c_2 \geq 0, \end{aligned} \tag{11}$$

where $l(p_i, f(x_i,y_i)) = max(0, 1 - p_i f(x_i,y_i))$ is the loss function, and $p_i = \{+1,-1\}$ is the label of the i-th training sample. Similar to [34], we reformulate Eq. (10) by using the dual form of the SVM:

$$\begin{aligned} &\min_{c} \frac{1}{2}\left(c_1^2 + c_2^2\right) + J(c_1,c_2) \\ &s.t. \quad c_1 + c_2 = 1, c_1, c_2 \geq 0, \end{aligned} \tag{12}$$

where

$$\begin{aligned} &J(c_1,c_2) = \max_{\alpha} \sum_{i=1}^{N}\alpha_1 - \frac{1}{2}\sum_{i,j=1}^{N}\alpha_i\alpha_j p_i p_j (c_1 k_1(x_i,x_j) + c_2 k_2(y_i,y_j)) \\ &s.t. \quad \sum_{i_1}^{N}\alpha_i p_i = 0, 0 \leq \alpha_i \leq C, i = 1,2,\cdots,N, \end{aligned} \tag{13}$$

α is the dual variable, $k_1(\cdot,\cdot)$ and $k_2(\cdot,\cdot)$ are kernel functions for the two kinds of training features, respectively. Here, the RBF kernel function $k_1(x_i,x_j) = exp(-\gamma||x_i - x_j||^2)$and the linear kernel function $k_2(y_i,y_j) = y_i^T y_j$ are used, where $\gamma > 0$ is the kernel parameter. Following [34], we solve Eq. (11) by iteratively updating the linear combination coefficients c and the dual variable α.

4. Experimental results

We evaluate our method on two public datasets: the CASIA dataset and the Replay-Attack dataset. We compare our method with a number of face liveness detection methods and demonstrate the outstanding performance of the proposed method.

4.1. Datasets

4.1.1. CASIA

The CASIA Face Anti-Spoofing Dataset [38] consists of 600 video clips of 50 different subjects. These video clips are divided into 150 real-access videos and 450 spoofing attack videos. The fake faces are made from the high quality records of the real faces. Warped photo attack, cut photo attack and video attack are the three spoofing attacks in the dataset. The dataset also takes into consideration the different imaging qualities in spoofing attacks. Some samples of the CASIA dataset are shown in Fig. 4.

Fig. 4. Samples from the CASIA dataset (upper row: live faces; lower row: fake faces).

4.1.2. Replay-Attack

The Replay-Attack dataset [8] consists of 1300 video clips of 50 different subjects. These video clips are divided into 300 real-access videos and 1000 spoofing attack videos. The resolution of the videos is 320 × 240 pixels. The dataset takes into consideration the different lighting conditions used in spoofing attacks. Some samples of the Replay-Attack dataset are shown in Fig. 5. Note that the Replay-Attack database consists of training set, development set and testing set.

4.2. Parameter settings

In our anisotropic diffusion scheme, we use $c(\cdot) = exp(-/K^2)$ as the gradient value function and the constant K was fixed as 15. The iteration numbers (L) is 15. We fixed the value of λ as 0.15 in Eq. (4). In our generalized multiple kernel learning method, the combination coefficients c_1 and c_2 are initialized as $1/2$. Since the face liveness detection is a binary classification problem, parameter γ in the RBF kernel function $k_1(x_i, x_j)$ is fixed as $1/2$.

4.3. Diagnostic analysis

In this section, we analyze three aspects of our method for face liveness detection, including the effect of the anisotropic diffusion, the effect of the frame selection, and the effect of the deep features.

1) Effectiveness of the anisotropic diffusion: To evaluate the contribution of the anisotropic diffusion, we construct a baseline method (baseline1) that uses the kernel matrix feature without diffusion (K+Deep+MKL), and compare the face liveness detection results obtained by using the proposed method (DK+Deep+MKL) and baseline1. The comparison results are show in Table 1. We can see that the anisotropic diffusion can improve the overall performance greatly in the CASIA dataset, which proves the effectiveness of the anisotropic diffusion.

2) Effectiveness of the frame selection: In the proposed method, one frame is randomly selected as the entry to the CNN to extract deep features. We also test the impact of representing the video by averaging the deep features of all the frames (baseline2). The comparison results are also shown in Table 1. The results show that averaging the deep features of all the frames degrades the performance.

3) Effectiveness of the deep features: We construct baseline3 (using the DK features only), baseline4 (using deep features only, the deep features are extracted on a randomly selected frame), and baseline 5 (using deep features only, the deep features are obtained by averaging the deep features of all the frames). The comparison results are shown in Table 1. The results show that the DK features and deep features extracted on a randomly selected frame obtain similar performance and the CASIA dataset, and DK features obtain much better performance on the Replay-Attack dataset. And the combination of DK features and deep features can further improve the performance.

4.4. Performance evaluation on the CASIA dataset

The half total error rate (HTER) [5] is used to measure and compare the performance. The HTER is half of the sum of the false rejection rate (FRR) and false acceptance rate (FAR):

$$HTER = \frac{FRR + FAR}{2}$$

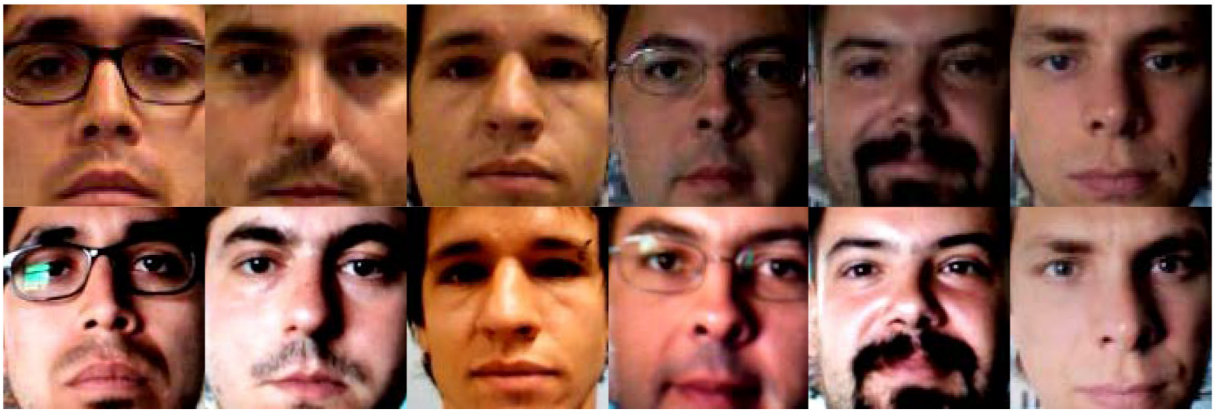

Fig. 5. Samples from the Replay-Attack dataset (upper row: live faces; lower row: fake faces).

Table 1
Comparison results between baseline methods.

Method	CASIA				Replay Attack			
	Accuracy%	FAR%	FRR%	HTER%	Accuracy%	FAR%	FRR%	HTER%
Baseline1 (K+Deep(Random)+MKL)	93.89	6.67	4.44	5.56	100	0	0	0
Baseline2 (DK+Deep(Average)+MKL)	72.22	13.33	71.11	42.44	100	0	0	0
Baseline3 (DK)	94.72	12.22	2.96	7.59	100	0	0	0
Baseline4 (Deep(Random))	92.78	7.78	5.56	6.67	87.92	11.75	13.75	12.75
Baseline5 (Deep(Average))	51.67	53.7	32.22	42.96	62.71	36.5	41.25	38.88
Ours (DK+Deep(Random)+MKL)	**97.5**	**2.22**	**3.33**	**2.78**	**100**	**0**	**0**	**0**

The bold values in the table are results of our method.

We compare our method with the LiveNet [28], Guided Scale Texture [26], Perceptual Image Quality [37] and Deep Stack Net [22], and the comparison results are shown in Table 2. As the anisotropic diffusion can enhance the edges of each face image in a video, and the kernel matrix can capture the inner correlation of the face images in the video, our method obtains better performance than the compared methods. The result of MKL shows that the weight of the DK feature is 0.78 and the weight of the deep feature is 0.22.

4.5. Performance evaluation on the Replay-Attack dataset

A performance comparison with previously proposed methods is shown in Table 3. In the MKL process, the weight of the DK feature is 0.81 and the weight of the deep feature is 0.19. The HTER of our method is 0% on the Replay-Attack test set and devel set. From Table 3, we can see that the DK feature can evidently estimate the difference in texture properties between live and fake face images. The result of our method (DK+Deep+MKL) is better than other diffusion-based methods [1, 13] which indicates that the kernel matrix can capture the inner correlation of the face images in the video.

5. Conclusions

In this paper, we present a diffusion-based kernel matrix model for face liveness detection. The anisotropic diffusion can enhance the edges of face images. Diffusion kernel (DK) features extracted from the diffused images can capture the differences in texture properties and inner correlations between live and fake face images. The DK features and the deep features are fused by a generalized multiple kernel learning method. Experimental results show the superiority and outstanding performance of our method.

Table 2
Performance comparison on the CASIA dataset with other methods.

Methods	HTER
LiveNet [28]	4.59%
Perceptual Image Quality [37]	12.7%
Guided Scale Texture [26]	3.05%
Deep Stack Net [22]	3.42%
DK+Deep+MKL(Ours)	2.78%

Table 3
Performance comparison using HTER measure on the Replay-Attack dataset.

Methods	devel	test
ND-CNN [2]	–	10%
DS-LSP [13]	13.73%	12.50%
LBP+SVM [21]	13.90%	13.87%
$LBP^{u2}_{3\times3}$+SVM [8]	14.84%	15.16%
$LBP^{u2}_{3\times3}$+LDA [8]	19.60%	17.17%
$LBP^{u2}_{3\times3}+\chi^2$ [8]	31.24%	34.01%
LiveNet [28]	–	5.74%
Perceptual Image Quality [37]	–	5.38%
Guided Scale Texture [26]	0.69	3.31%
DK+Deep+MKL(Ours)	0%	0%

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] A. Alotaibi, A. Mahmood, Deep face liveness detection based on nonlinear diffusion using convolution neural network, Signal, Image and Video Processing (2016) 1–8.
[2] A. Alotaibi, A. Mahmood, Deep face liveness detection based on nonlinear diffusion using convolution neural network, Signal Image and Video Processing 11 (4) (2017) 713–720.
[3] A. Anjos, M.M. Chakka, S. Marcel, Motion-based counter-measures to photo attacks in face recognition, IET biometrics 3 (3) (2014) 147–158.
[4] W. Bao, H. Li, N. Li, W. Jiang, A liveness detection method for face recognition based on optical flow field, International Conference on Image Analysis and Signal Processing, IEEE. 2009, pp. 233–236.
[5] S. Bengio, J. Mariéthoz, A statistical significance test for person authentication, Proceedings of Odyssey 2004: The Speaker and Language Recognition Workshop, 2004.EPFL-CONF-83049.
[6] S. Bharadwaj, T.I. Dhamecha, M. Vatsa, R. Singh, Computationally efficient face spoofing detection with motion magnification, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2013. pp. 105–110.
[7] Z. Boulkenafet, J. Komulainen, A. Hadid, Face antispoofing using speeded-up robust features and Fisher Vector Encoding, IEEE Signal Processing Letters 24 (2) (2017) 141–145.
[8] I. Chingovska, A. Anjos, S. Marcel, On the effectiveness of local binary patterns in face anti-spoofing, BIOSIG-Proceedings of the International Conference of the Biometrics Special Interest Group (BIOSIG), IEEE. 2012, pp. 1–7.
[9] M. De Marsico, C. Galdi, M. Nappi, D. Riccio, Firme: face and iris recognition for mobile engagement, Image and Vision Computing 32 (12) (2014) 1161–1172.
[10] P. Fei, Q. Le, L. Min, Face presentation attack detection using guided scale texture, Multimedia Tools and Applications (3) (2017) 1–27.
[12] S. Haykin, N. Network, A comprehensive foundation, Neural Networks 2 (2004) (2004) 41.
[13] W. Kim, S. Suh, J.-J. Han, Face liveness detection from a single image via diffusion speed model, IEEE transactions on Image processing 24 (8) (2015) 2456–2465.
[14] Y. Kim, J. Na, S. Yoon, J. Yi, Masked fake face detection using radiance measurements, JOSA A 26 (4) (2009) 760–766.
[15] K. Kollreider, H. Fronthaler, J. Bigun, Non-intrusive liveness detection by face images, Image and Vision Computing 27 (3) (2009) 233–244.
[16] K. Kollreider, H. Fronthaler, M.I. Faraj, J. Bigun, Real-time face detection and motion analysis with application in "liveness" assessment, IEEE Transactions on Information Forensics and Security 2 (3) (2007) 548–558.
[17] R. Kondor, T. Jebara, A kernel between sets of vectors, ICML, 20, 2003. pp. 361.
[18] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, Advances in neural information processing systems, 2012. pp. 1097–1105.
[20] J. Li, Y. Wang, T. Tan, A.K. Jain, Live face detection based on the analysis of fourier spectra, Defense and Security, International Society for Optics and Photonics. 2004, pp. 296–303.
[21] J. Määttä, A. Hadid, M. Pietikäinen, Face spoofing detection from single images using micro-texture analysis, International Joint Conference on Biometrics (IJCB), IEEE. 2011, pp. 1–7.
[22] X. Ning, W. Li, M. Wei, L. Sun, X. Dong, Face anti-spoofing based on deep stack generalization networks., ICPRAM, 2018. pp. 317–323.

[23] M. Oquab, L. Bottou, I. Laptev, J. Sivic, Learning and transferring mid-level image representations using convolutional neural networks, Proceedings of the IEEE conference on computer vision and pattern recognition, 2014. pp. 1717–1724.
[24] G. Pan, L. Sun, Z. Wu, S. Lao, Eyeblink-based anti-spoofing in face recognition from a generic webcamera, IEEE 11th International Conference on Computer Vision, IEEE. 2007, pp. 1–8.
[25] B. Peixoto, C. Michelassi, A. Rocha, Face liveness detection under bad illumination conditions, 18th IEEE International Conference on Image Processing (ICIP), IEEE. 2011, pp. 3557–3560.
[26] F. Peng, L. Qin, M. Long, Face presentation attack detection using guided scale texture, Multimedia Tools and Applications (2018)
[27] P. Perona, J. Malik, Scale-space and edge detection using anisotropic diffusion, IEEE Transactions on pattern analysis and machine intelligence 12 (7) (1990) 629–639.
[28] Y.A.U. Rehman, P.L. Man, M. Liu, LiveNet: improving features generalization for face liveness detection using convolution neural networks, Expert Systems with Applications (2018)
[29] F. Schroff, D. Kalenichenko, J. Philbin, Facenet: a unified embedding for face recognition and clustering, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015. pp. 815–823.
[30] A.K. Singh, P. Joshi, G.C. Nandi, Face recognition with liveness detection using eye and mouth movement, International Conference on Signal Propagation and Computer Technology (ICSPCT), IEEE. 2014, pp. 592–597.
[31] Y. Sun, Y. Chen, X. Wang, X. Tang, Deep learning face representation by joint identification-verification, Advances in neural information processing systems, 2014. pp. 1988–1996.
[32] X. Tan, Y. Li, J. Liu, L. Jiang, Face liveness detection from a single image with sparse low rank bilinear discriminative model, European Conference on Computer Vision, Springer. 2010, pp. 504–517.
[33] S. Tirunagari, N. Poh, D. Windridge, A. Iorliam, N. Suki, A.T. Ho, Detection of face spoofing using visual dynamics, IEEE transactions on information forensics and security 10 (4) (2015) 762–777.
[34] M. Varma, B.R. Babu, More generality in efficient multiple kernel learning, Proceedings of the 26th Annual International Conference on Machine Learning, ACM. 2009, pp. 1065–1072.
[35] E. Vazquez-Fernandez, D. Gonzalez-Jimenez, Face recognition for authentication on mobile devices, Image and Vision Computing 55 (2016) 31–33.
[36] L. Wang, J. Zhang, L. Zhou, C. Tang, W. Li, Beyond covariance: feature representation with nonlinear kernel matrices, Proceedings of the IEEE International Conference on Computer Vision, 2015. pp. 4570–4578.
[37] C.-H. Yeh, H.-H. Chang, Face liveness detection based on perceptual image quality assessment features with multi-scale analysis, 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), IEEE. 2018, pp. 49–56.
[38] Z. Zhang, J. Yan, S. Liu, Z. Lei, D. Yi, S.Z. Li, A face antispoofing database with diverse attacks, 5th IAPR international conference on Biometrics (ICB), IEEE. 2012, pp. 26–31.
[39] Z. Zhang, D. Yi, Z. Lei, S.Z. Li, Face liveness detection by learning multispectral reflectance distributions, IEEE International Conference on Automatic Face & Gesture Recognition and Workshops (FG), IEEE. 2011, pp. 436–441.