From Local Binary Patterns to Pixel Difference Networks for Efficient Visual Representation Learning

Zhuo Su[1], Matti Pietikäinen[1], and Li Liu[1,2(✉)]

[1] Center for Machine Vision and Signal Analysis, University of Oulu, Oulu, Finland
li.liu@oulu.fi
[2] National University of Defense Technology, Changsha, China

Abstract. LBP is a successful hand-crafted feature descriptor in computer vision. However, in the deep learning era, deep neural networks, especially convolutional neural networks (CNNs) can automatically learn powerful task-aware features that are more discriminative and of higher representational capacity. To some extent, such hand-crafted features can be safely ignored when designing deep computer vision models. Nevertheless, due to LBP's preferable properties in visual representation learning, an interesting topic has arisen to explore the value of LBP in enhancing modern deep models in terms of efficiency, memory consumption, and predictive performance. In this paper, we provide a comprehensive review on such efforts which aims to incorporate the LBP mechanism into the design of CNN modules to make deep models stronger. In retrospect of what has been achieved so far, the paper discusses open challenges and directions for future research.

Keywords: Local binary pattern · Pixel difference convolution · Convolutional neural network · Visual representation learning

1 Introduction

Local binary pattern (LBP) is one of the most prominent feature descriptors in the computer vision community. With its distinctive advantages, namely, ease of implementation, invariance to monotonic illumination changes, and low computational complexity, it was widely studied with numerous LBP variants proposed and applied for a diverse range of applications such as texture classification [27–29], dynamic texture recognition [54], image matching [14], visual inspection [31], image retrieval [8], biomedical image analysis [25,26], facial analysis [1,36,51], motion and activity analysis [17], object detection [30,37], and background subtraction [12].

With the development of deep learning [18], convolutional neural networks (CNNs), on the other hand, appeared to be a more powerful feature extractor for visual inputs. Instead of hand-crafting features like LBP, CNN structures are capable of learning rich hierarchical features from local texture statistics to global semantic information that helps the models to achieve human-level or even surpass-human performances in many visual applications. In the deep learning era, it is necessary to think about the following questions: *Is LBP still worth exploring in computer vision? What role does LBP take when CNNs are so strong?* or *Can LBP help the design of CNNs to make our models stronger and how?*
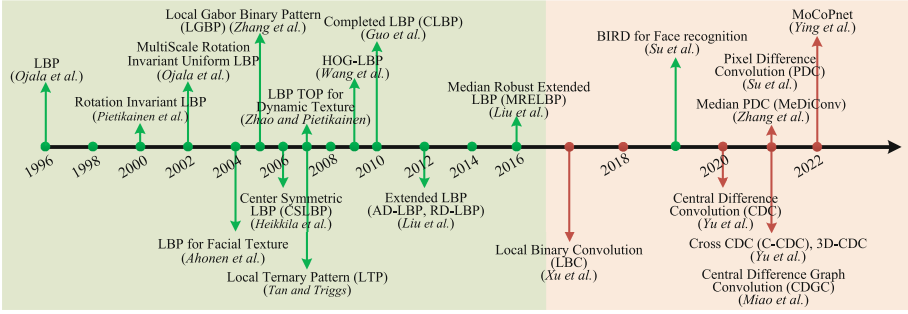
Fig. 1. The evolution of LBP over the past decades.

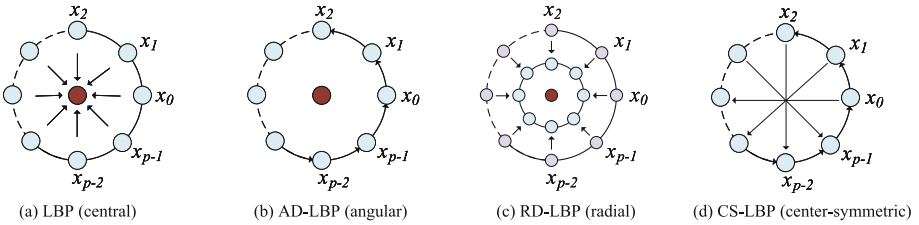| (a) LBP (central) | (b) AD-LBP (angular) | (c) RD-LBP (radial) | (d) CS-LBP (center-symmetric) |

Fig. 2. Illustration of the original LBP [27], angular LBP [22], radial LBP [22], and center-symmetric LBP [13].

In this paper, we aim to answer these questions with a review of the recent efforts on LBP in the deep learning era. Briefly, LBP, or the design philosophy of LBP, has still shown great benefits in the past years that help CNN models to boost their performance in terms of prediction accuracy, efficiency, and memory, on a wide range of applications like face anti-spoofing, edge detection, gesture recognition, and object classification. We hope our review can inspire researchers seeking a higher-level perspective in the future. A brief history of LBP is depicted in Fig. 1.

The rest of the paper is organized as follows. In Sect. 2, we give a short introduction of traditional LBP and its variants. Then, we focus our attention on the works of LBP-inspired CNN modules in Sect. 3. Applications of the introduced methods are present in Sect. 4. Finally, we discuss possible future works and conclude the paper in Sect. 5.

2 Traditional LBP

LBP descriptors were firstly introduced by Ojala *et.al.* [27,28] to encode pixel-wise information in textured images. Specifically, an input image is probed locally by sampling the values from the neighborhood. As shown in Fig. 2 (a), for a certain pixel x_c, the values from neighboring locations $\{x_0, x_1, ..., x_{p-1}\}$ spaced equidistantly around a circle with radius r are extracted to generate a binary code composed of 0 and 1, by comparing each of those values with the central value x_c. That is, the neighboring values greater than or equal to x_c are associated with 1, otherwise with 0. The binary values

0 and 1 are read anticlockwise from the starting point x_0 to the ending point x_{p-1}, leading to a p-length binary code as a descriptive local pattern, which is then represented by a decimal number. The feature of the image is then represented by the histogram of all the possible binary codes. The generation of an LBP code can be formulated as:

$$\text{LBP}_{r,p}(c) = \sum_{i=0}^{p-1} s(x_i - x_c)2^i, \quad s(x) = \begin{cases} 1 & x \geq 0, \\ 0 & x < 0. \end{cases} \tag{1}$$

By altering the number of sampling pixels p and the radius r, LBP patterns can be extracted in different scales and complexities.

While simple in computation and invariant to illumination changes, the original LBP (Fig. 2 (a)) suffers from significant disadvantages such as exponential growth of possible patterns with the increase of p (*i.e.*, 2^p), failure to detect large-scale texture structures, sensitivity to image rotation and noise. To address the above limitations, numerous LBP variants were successfully developed such as the rotation invariant LBP [29,53], uniform LBP [28], extended LBP [22] (Fig. 2 (b-c)), center-symmetric LBP (Fig. 2 (d)), noise robust LBP [21]. Generally, these methods improve the original LBP from the following aspects: (1) Changing neighborhood topology and sampling [13,22] (*i.e.*, the layout of the sampled neighboring pixels). (2) Adopting different thresholding and quantization methods [36] (*i.e.*, the implementation of the sign function $s(x)$ in Eq. (1)); (3) Designing different encoding or grouping strategies (*e.g.*, rotation invariant patterns [29], uniform patterns [28], learnable LBP patterns [33]); (4) Combining complementary features (*e.g.*, HOG features [39], macro-textons with Gabor filters [2,51], and pixel difference magnitudes [11]). A more comprehensive review can be seen in [19].

3   LBP Meets CNNs: To Benefit from both Worlds

Deep learning has revolutionized computer vision in a broad range of applications. The convolution operators act as the basic local feature descriptors like LBP that probe local details in their current receptive fields. By stacking multiple layers of convolutional layers, the receptive fields of convolution modules are gradually increased, allowing CNN models to effectively capture both fundamental low-level features like textures, colors, corners, and higher-level abstract features that represent object semantics. In contrast, LBP behaves like convolution operators but probes local details in a different way [2].

In the deep learning era, LBP may perform better than CNN models in certain properties.[1] In some cases, LBP variants even outperform CNNs for standard (traditional) texture test sets, *e.g.*, in noisy conditions [19,20]. We list the properties as below:

   i. Computational simplicity and efficiency: LBP codes are binary, thus can be efficiently computed and are memory friendly.

---

[1] It should be noted LBP has possibly more than these three properties. Here we only focus on properties that give more inspirations for designing CNN modules.
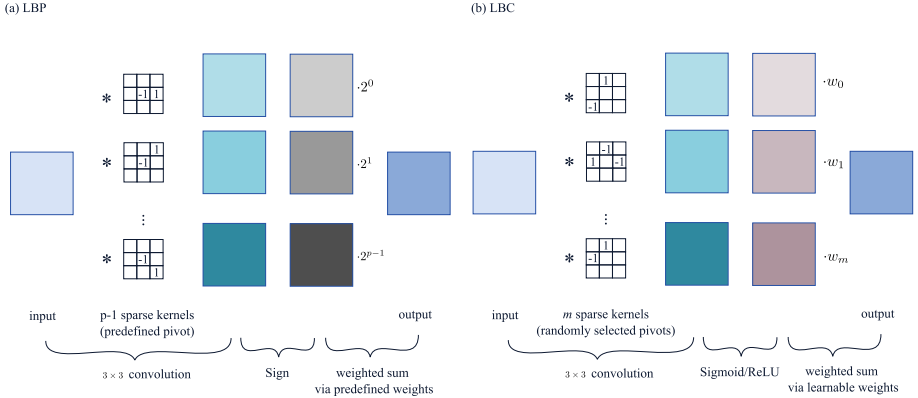
Fig. 3. We can regard LBP as a 3×3 convolutional layer, where the calculation of pixel differences between neighboring pixels and the central pixel is equivalent to a series of 3×3 convolutions with sparse binary kernels. After the convolution, the difference maps are process with sign function followed by linear combination via predefined weights (2^i). LBC upgrades LBP with alternative sparse binary kernels, Simoid and ReLU functions, and learnable weights to pool those difference maps.

   ii. Easing the encoding of higher-order information: the pixel differences within local areas contain rich image gradient cues compared with the original pixel intensities used in CNNs [32].

  iii. Ability of probing various microstructures from images: the numerous neighboring topologies and sampling strategies in LBP and its variants provide diversity to probe rich local patterns for visual inputs.

However, unlike CNNs, LBP descriptors have limited representational capacity due to their fixed ways to calculate the patterns and the associated shallow structures. Since LBP and CNNs are complementary in these properties, how to combine them to benefit from both worlds is an inspiring and meaningful research topic. In the following sections, we discuss some recent efforts under this direction regarding the properties *i.-iii.* of LBP, and hope to give helpful insights for future works.

3.1 Local Binary Convolution

A pioneering work combing LBP and convolutional operations is local binary convolution (LBC) [16,52]. The authors decomposed the pattern generation process of LBP into sub-steps. As implied in Eq. (1), first, the central pixel x_c in a patch is chosen as a pivot to calculate the differences with other neighboring pixels $\{x_i\}_{i=0}^{p-1}$. Second, the nonlinear Sign function is used to convert each pixel difference to a binary value. Finally, all these binary values are pooled via a linear combination with fixed weights $\{2^i\}_{i=0}^{p-1}$. Thereby, LBP can be generalized into a more flexible form by changing the selection of pivot pixel, the nonlinear function, and values of the weights.
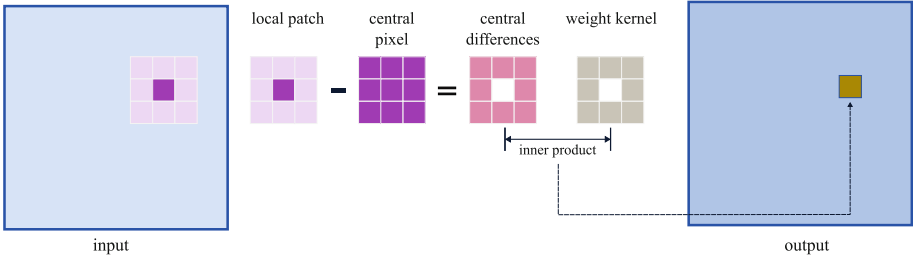
Fig. 4. Unlike standard convolution using pixel intensities, CDC leverages central pixel differences to conduct convolutional operation.

To help embed the above process into the convolutional operation in CNNs, the calculation of differences between pivot pixel and other pixels can be regarded as a 3×3 convolution, where each 3×3 kernel has sparse binary values [2], as shown in Fig. 3 (a). Based on that, LBC randomly generated m sparse binary kernels with different locations of -1 and 1 (the binary kernels were then fixed afterwards). The locations of -1 represent pivot positions. Then, the Sign function was changed to Sigmoid or ReLU to create real-valued maps. Finally, the predefined weights (using base 2) were replaced with learnable real-valued weights to pool these real-valued maps. The process of LBC is illustrated in Fig. 3 (b).

Discussion. Compared with standard convolution, LBC shows considerable reduction in computational cost and memory storage of the parameters. On one hand, the sparse binary kernels in LBC allow it to generate m intermediate feature maps efficiently, since binary operations are much faster than real-valued counterparts [15]. On the other hand, linear combination of those intermediate features with learnable weights can be regarded as an efficient 1×1 convolution. Both binary kernels and 1×1 kernels need much less memory storage than standard 3×3 kernels, leading to lower model size.

3.2   Central Difference Convolution and Its Variants

Central Difference Convolution (CDC).   CDC [44,46,47] incorporates the original LBP into convolutional operation to capture gradient information from the input images. As shown in Fig. 4, to generate a output value in a certain location, a convolution operator firstly samples a local region in the input feature map, consisting of the central pixel x_c and all its neighboring p pixels $\{x_i\}_{i=0}^{p-1}$. Similar to LBP, CDC calculates the differences between the central pixel and its neighboring pixels as central differences, which are aggregated via the learnable kernel weights. Formally, CDC can be written as:

$$y_c = \sum_{i=0}^{p-1} w_i \cdot (x_i - x_c),\tag{2}$$

where y_c is the output value, $\{w_i\}_{i=0}^{p-1}$ represent the kernel weights.

(a) Horizontal & Vertical C-CDC

(b) Diagonal C-CDC

local patch · central pixel · central differences · weight kernel

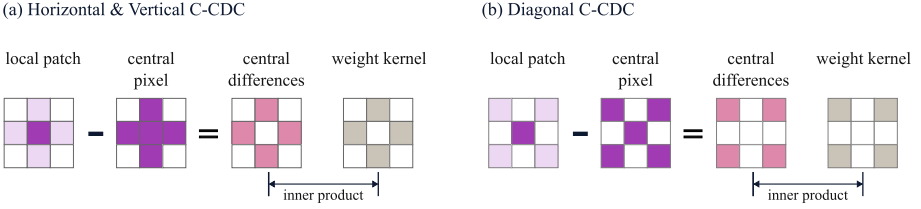local patch · central pixel · central differences · weight kernel

Fig. 5. By changing the sampling locations, CDC are revised to two lightweight C-CDC versions.

CDC was originally proposed for anti-spoofing task. However, both intensity level semantic information and gradient level text details are crucial for distinguishing live and spoofing faces. Therefore, the authors in [47] further proposed to combine standard convolution and CDC to create a generalized CDC operator to capture both types of information:

$$y_c = \theta \cdot \underbrace{\sum_{i=0}^{p-1} w_i \cdot (x_i - x_c)}_{gradient\ level} + (1 - \theta) \cdot \underbrace{(\sum_{i=0}^{p-1} w_i \cdot x_i + w_c \cdot x_c)}_{intensity\ level}, \qquad (3)$$

where θ is a hyperparameter that tradeoffs the contribution between intensity level and gradient level information.

Cross CDC. In CDC, central gradients are calculated from all neighbors, which might be redundant and sub-optimal due to the discrepancy among diverse gradient directions. Yu *et al.* [45] proposed to decouple such directions into two cross components (*i.e.*, horizontal/vertical and diagonal) for better extracting gradient information, deriving two types of cross CDC (C-CDC), as shown in Fig. 5. Similar to CDC, by combining standard convolution, C-CDC can be generalized to capture both intensity and gradient level information:

$$y_c = \theta \cdot \underbrace{\sum_{i=0}^{p_S-1} w_i \cdot (x_i - x_c)}_{gradient\ level} + (1 - \theta) \cdot \underbrace{(\sum_{i=0}^{p_S-1} w_i \cdot x_i + w_c \cdot x_c)}_{intensity\ level}, \qquad (4)$$

where p_S is the number sampling pixels collected according to a certain direction (*i.e.*, horizontal/vertical or diagonal) in the local region centered at x_c.

An additional benefit of C-CDC is either type of C-CDC has less parameters than CDC or standard convolution operator, as only half of the neighboring pixels are sampled.

3D-CDC for Spatial/temporal Representation. For video based visual tasks, spatio-temporal feature representation learning is the core to design well-performed CNN
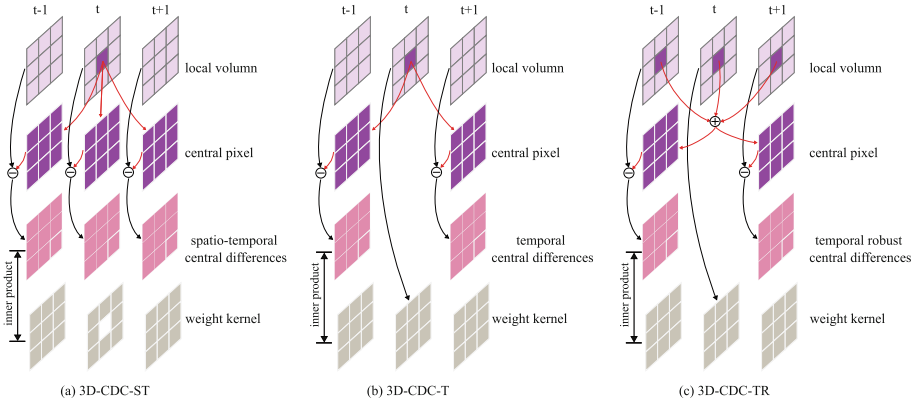
Fig. 6. The design spirit of CDC can also be applied to 3D input, where the differences between pixels in other frames and pixels in current frame can be aggregated during 3D convolution to enrich temporal feature representation. The figure shows three versions of 3D-CDC. $\oplus$ denotes mean operation.

models. To exploit the rich local motion and enhance the spatio-temporal representation for 3D CNNs, 3D-CDC [48] was proposed to aggregate central differences along spatial and temporal directions as shown in Fig. 6.

Via different subtraction strategies, 3D-CDC can be flexibly designed to capture spatial gradient information or temporal differences. Specifically, the authors in [48] developed three 3D-CDC operators for gesture recognition. First, spatio-temporal 3D-CDC (3D-CDC-ST), as shown in Fig. 6 (a), was designed considering both spatial and temporal gradient cues to effectively enhance local texture and motion details in RGB sequences. The second operator illustrated in Fig. 6 (b), temporal 3D-CDC (3D-CDC-T) was then proposed to focus on temporal central differences to better model temporal dynamics. Finally, considering sensor noise especially in the depth modality of RGB-D input, a more robust version of 3D-CDC (3D-CDC-TR) aims to reduce the sensitiveness to noise in pixels such as pixel jitters from the adjacent time steps by averaging the spatial centers of all time steps before calculating central differences (Fig. 6 (c)).

The formulations of 3D-CDC can be easily derived by extending Eq. (3) or Eq. (4) to a 3D version which also uses the tradeoff parameter θ to combine both intensity and gradient level image cues.

Discussion. Unlike LBC [16] that leverages binary sparse kernels to reduce computation and memory for convolutional operations (*i.e.*, property *i.* of LBP as we discussed in the beginning of this section), CDC series focus on property *ii.*: explicit encoding of higher-order information. In other words, rich gradient cues in both 2D images or 3D image sequences can be effectively captured by CDC to enhance visual feature representation, which are proved to be helpful for better prediction accuracy on various computer vision tasks.
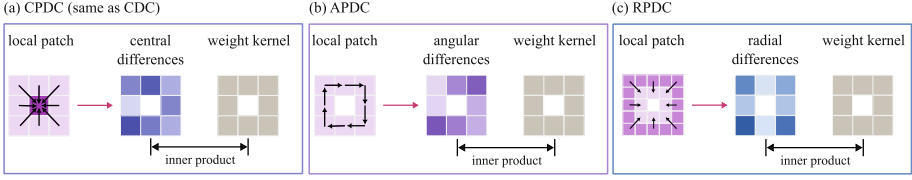
Fig. 7. PDC is more general than CDC to probe local difference patterns in various encoding directions. The figure shows three versions of PDC, where CDC is a special case (a). By changing the sampling strategy, more PDC instances can be derived.

3.3 Pixel Difference Convolution and Its Variants

Pixel Difference Convolution (PDC). PDC [32] has a more general form in encoding local differences by changing the sampling strategies in the local region, allowing it to probe micro structures in a more flexible way, which is an essential property of LBP descriptors. In this way, the above introduced CDC becomes a special case of PDC where the central differences are adopted.

As shown in Fig. 7 (a-c), PDC incorporates the calculation of pixel differences in a more general way when conducting convolutional operation. We can compare the standard convolution and PDC with the following formulations:

$$y_c = f(\boldsymbol{x}, \boldsymbol{\theta}) = \sum_{i=1}^{p-1} w_i \cdot x_i + w_c \cdot x_c, \qquad \text{(standard convolution)} \qquad (5)$$

$$y_c = f(\Delta \boldsymbol{x}, \boldsymbol{\theta}) = \sum_{(x_i, x_i') \in \boldsymbol{\mathcal{P}}} w_i \cdot (x_i - x_i'), \qquad \text{(PDC)} \qquad (6)$$

where $\boldsymbol{\mathcal{P}} = \{(x_1, x_1'), (x_2, x_2'), ..., (x_m, x_m')\}$ is the set of pixel pairs picked from the current local region and m is the number of pixel pairs, $\boldsymbol{\theta} = \{w_1, w_2, ..., w_i, ...\}$ are the kernel weights.

To better capture diverse micro-structural patterns, pixel pairs can be selected according to various probing strategies in the LBP literature. In [32], LBP and ELBP [22,28,33] were adopted to encode pixel relations from varying directions (angular and radial). By integrating LBP and ELBP into convolution, three types of PDC instances were derived, namely, central PDC (CPDC), angular PDC (APDC), and radial PDC (RPDC), respectively (Fig. 7 (a-c)). For example, for APDC with a 3×3 kernel, 8 pairs are first selected in the angular direction within the 3×3 local region (thus $m = 8$). Then, pixel differences between these pixel pairs are aggregated with the kernel weights to calculate the output value. Obviously, both CDC and C-CDC are special cases of PDC by composing pixel pairs including the central pixel. In CDC, $m = 8$ and in C-CDC, $m = 4$. For both cases, $x_i' \equiv x_c$.

The authors in [32] also proved that PDC can be converted to standard convolution with a novel reparameterization strategy to avoid double computation and runtime memory due to the calculation of pixel difference (*i.e.*, Eq. (6) costs more computation and memory than Eq. (5)). An example of such conversion of APDC is illustrated
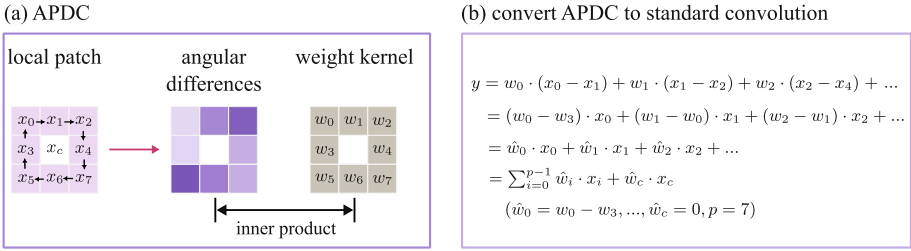
(b) convert APDC to standard convolution

local patch   angular   weight kernel
               differences

$x_0 \rightarrow x_1 \rightarrow x_2$
$x_3 \quad x_c \quad x_4$
$x_5 \leftarrow x_6 \leftarrow x_7$

$w_0 \quad w_1 \quad w_2$
$w_3 \qquad w_4$
$w_5 \quad w_6 \quad w_7$

inner product

$$y = w_0 \cdot (x_0 - x_1) + w_1 \cdot (x_1 - x_2) + w_2 \cdot (x_2 - x_4) + ...$$
$$= (w_0 - w_3) \cdot x_0 + (w_1 - w_0) \cdot x_1 + (w_2 - w_1) \cdot x_2 + ...$$
$$= \hat{w}_0 \cdot x_0 + \hat{w}_1 \cdot x_1 + \hat{w}_2 \cdot x_2 + ...$$
$$= \sum_{i=0}^{p-1} \hat{w}_i \cdot x_i + \hat{w}_c \cdot x_c$$
$$(\hat{w}_0 = w_0 - w_3, ..., \hat{w}_c = 0, p = 7)$$

Fig. 8. PDC can be converted to standard convolution to save computation and memory.

in Fig. 8. By doing so, PDC is implemented as efficient as standard convolution during inference.

Median Pixel Difference Convolution (MeDiConv). Inspired by median robust extended LBP (MRELBP) [21] which improves LBP with greatly enhanced robustness to noise by considering local median values, MeDiConv [50] adopts a similar strategy when designing the convolution process. Specifically, by replacing x_i' in Eq. (6) with the local median value, MeDiConv can be written as:

$$y_c = f(\Delta \boldsymbol{x}_m, \boldsymbol{\theta}) = \sum_{i=1}^{p-1} w_i \cdot (x_i - x_m), \qquad \text{(MeDiConv)}, \qquad (7)$$

where x_m is the median value of the sampled local region.

Compared with standard convolution, MeDiConv is a nonlinear smoothing operation, which can effectively remove outliers with limited impact on the ability of feature extraction. For standard convolution, noise corrupted image patterns could lead to false activations, leading to a significant decrease in accuracy. In contrast, for MeDiConv, the effect of applying MeDiConv at multiple layers can be considered as applying multiple median filters of different kernel sizes on the original image as each MeDiConv layer has a different receptive field. Consequently, MeDiNet was demonstrated to be able to deal with the noise of different levels.

Discussion. PDC inherits the properties *ii.* and *iii.* from LBP, forming a more versatile convolution operator to both capture gradient information from the image, and probing microstructural cues from different encoding directions. PDC enables a more straightforward way to integrate traditional LBP variant into modern CNN modules. Since both LBP and PDC involve sampling local pixels and selecting certain pixel differences, an LBP variant can be easily embedded into PDC when PDC adopts the same sampling and selection strategies following the corresponding LBP variant. MeDiConv is an example of such transfer from MRELBP to convolution, *i.e.*, both calculate the differences between neighboring pixels and the local median value. Therefore, the noise robustness property of MRELBP is inherited in MeDiConv. Meanwhile, MeDiConv owns the feature extraction ability of CNNs. Benefits from both worlds are well combined.

4   Applications

An extensive review on applications of traditional LBP has been already included in [19]. Therefore, we focus on providing a complementary review on applications of those LBP inspired CNN modules in the deep learning era.

Table 1. Applications of LBP inspired CNN modules.

Architecture name	Module	Application	Year	Description
Local binary convolutional neural networks (LBCNN) [16]	LBC	Image classification	2017	By using LBC in AlexNet, LBCNN saves $6.622\times$ learnable parameters in the convolutional layers, while performing comparably to the original AlexNet on ImageNet dataset [7]
Central Difference Convolutional Network (CDCN, CDCN++) [47]	CDC	Face anti-spoofing	2020	CDCN is built by manually stacking multiple CDC layers; CDCN++ is searched using neural architecture search (NAS) technique where CDC operators with different channels become basic elements in the search space. It is demonstrated that standard convolution fails to capture the consistent spoofing pattern while CDC is able to extract the invariant detailed spoofing features, *e.g.*, lattice artifacts. Comprehensive experiments are performed on six benchmark datasets to show that 1) CDC not only achieves superior performance on intra-dataset testing, 2) it also generalizes well on cross-dataset testing
Multi-Rate and Multi-Modal Temporal Enhanced Networks [48]	3D-CDC	Gesture recognition	2021	It is the first NAS based method for RGB-D gesture recognition. Similar to CDCN++, a novel search space is created by leveraging 3D-CDC-T and 3D-CDC-TR in the basic operators. 3D-CDC shows great ability to enhance the spatio-temporal representation for video understanding tasks. The resulting network achieves state-of-the-art performance on three benchmark datasets

(continued)

Table 1. (*continued*)

Architecture name	Module	Application	Year	Description
Dual-Cross Central Difference Network (DC-CDN) [45]	C-CDC	Face anti-spoofing	2021	Based on C-CDC operators, DC-CDN is established with cross feature interaction modules for mutual relation mining and local detailed representation enhancement. Comprehensive experiments are performed on four benchmark datasets with three testing protocols to demonstrate the state-of-the-art performance
Pixel Difference Networks (PiDiNet) [32]	PDC	Edge detection	2021	PiDiNet is structured by sequentially stacking different PDC instances as well as the standard convolution to capture gradient information in different encoding directions. Since PDC can be converted to standard convolution, PiDiNet does not suffer from extra computation and memory consumption. With an efficient design of backbone and side structures, PiDiNet is the first deep network that can achieve human-level performance without ImageNet pretraining, when evaluated on popular edge detection datasets
Median Pixel Difference Convolutional Network (MeDiNet) [50]	MeDi-Conv	Face recognition	2021	MeDiNet is built by embedding MeDiConv into CNN architectures to equip CNNs with built-in robustness to noise of different levels. MeDiNet is tested on popular face datasets with challenging settings by adding different levels of noise (*e.g.*, by changing blur kernels, noise intensities, scales, and JPEG quality factors). Extensive experiments show that MeDiNet can effectively remove noisy pixels in feature maps and suppress the negative impact of noise, leading to a more robust performance than standard CNNs

Table 1. (*continued*)

Architecture name	Module	Application	Year	Description
S-RaPiDiNet based on Random Pixel Different Convolution [23]	PDC	Face perception	2021	Unlike PDC adopting predefined sampling strategies for pixel pairs, Random PDC randomly samples pixel in the local region. The design mechanism is inspired by BRIEF descriptor [3]. Despite the simple strategy, S-RaPiDiNet shows great performance in face perception tasks
3D Central Difference Convolution Attention Network [55]	3D-CDC	rPPG measure-ment	2021	A central difference convolutional attention network for rPPG measurement is proposed. The adopted 3D-CDC can capture rich temporal context by gathering time difference information
Graph network based on Central Difference Graph Convolution (CDGC) [24]	CDC	Action recognition	2021	CDGC is designed for skeleton based action recognition. Like CDC aggregating the difference between neighboring pixels and the central pixel, CDGC takes differences between the features of the adjacent nodes and the central node. By generalizing CDGC with the incorporation of vanilla graph convolution, it is able to aggregate both node information and gradient information, leading to better accuracy than state-of-the-art methods
Multi-scale Texture Difference model (MTD-Net) [42]	CDC	Face forgery detection	2021	MTD-Net considers both pixel intensity and pixel gradient information to give a stationary description of texture difference information by using CDC in the network
Gradient Siamese Network (GSN) [6]	CDC	Image quality assessment	2022	Using CDC to obtain both semantic features and detail difference hidden in image pair. GSN won the second place in NTIRE 2022 Perceptual Image Quality Assessment Challenge track 1 Full-Reference [9]

<div align="right">(continued)</div>

Table 1. (*continued*)

Architecture name	Module	Application	Year	Description
Suppression-Strengthen Network (S2N) [38]	CDC	Event-based recognition	2022	An evolution guided density-adaptive central difference convolution scheme is proposed to progressively encode the local center-surrounding variation and adaptively aggregate the features into a complete event representation under the guidance of the motion evolution map
Depth Dynamic Center Difference Convolution (DDCDC) based network [40]	CDC	Monocular 3D Object Detection	2022	DDCDC introduces surrounding pixel cues in depth estimation and has different convolution kernels weights for each pixel of all examples. DDCDC not only overcomes the limitations of conventional 2D convolution, but also highlights the differences in depth information between the target and the background, so more attention is paid to interesting objects
Fast Saliency Model (FSM) [49]	CDC	Saliency prediction	2022	FSM consists of a modified U-net architecture, a location-dependent fully-connected layer, and CDC layers. Using the CDC layers at different scales enables capturing more robust and biologically motivated features
Local Motion and Contrast Priors Driven Deep Network (MoCoPnet) [43]	CDC	Infrared small target super-resolution	2022	Motivated by the local contrast prior in the spatial dimension, a central difference residual group is proposed to incorporate CDC into the feature extraction backbone, which can achieve center-oriented gradient-aware feature extraction to improve the target contrast for the task
CDC-based Multi-Receptive-Field (CDC-MRF) module [4]	CDC	Image enhance-ment	2022	CDC is embedded in the architecture to effectively extract multi-scale edge/texture features on thermal images. The extracted thermal features are then utilized as important guidance to facilitate the following low-light visible image enhancement task

Table 1. (*continued*)

Architecture name	Module	Application	Year	Description
RPDC based network structure [41]	PDC	yarn contour detection	2022	Firstly, according to the directional gradient distribution of yarn images in the visual system, a radial pixel difference convolution is improved to extract interested edge features. Secondly, considering the graphical characteristics of the yarn to be inspected, a mask layer of texture erosion is designed to further filter irrelevant details from extracted edge features.
Semantic Diffusion Network (SDN) [35]	CDC	Semantic segmentation	2022	SDN is proposed for approximating the diffusion process, which contains a parameterized semantic difference convolution operator followed by a feature fusion module and constructs a differentiable mapping from original backbone features to advanced boundary-aware features

Table 1 lists the methods published in recent years that leverage LBP inspired CNN modules on a wide range of applications, including image classification [16], face anti-spoofing [45,47], gesture recognition [48], edge detection [32], facial analysis [23,42,50], remote photoplethysmography (rPPG) measurement [55], action recognition [24], image quality accessment [6], event-based recognition [38], monocular object detection [40], saliency detection [49], infrared target super-resolution [43], and semantic segmentation [35]. Here, it would be tricky to compare different methods with quantitative evaluations due to the big variation of tasks. Generally, the three properties of LBP have been well-integrated into CNN architectures and shown significant advantages in boosting convolutional architectures for various of computer vision tasks.

5   Future Works and Conclusion

Future Works. Although great success has been made during the last years on developing novel modules with LBP, which enhance CNNs in efficiency, representational capacity, and discriminative power, we believe that it is just a starting step and the following aspects are worth exploring in the future.

– Currently, most LBP inspired deep modules are based on CNN architectures (*e.g.*, CDC, PDC) to extract gradient information on regular data like images and videos. However, gradient information is also important for irregular data like point clouds [10,34] and 3D meshes [5]. Designing novel modules that benefit irregular data can be a valuable direction.

– The three properties we discussed in Sect. 3 are the main motivations when designing LBP inspired modules. However, none of the existing methods utilizes all the three properties in tandem, limiting the current modules to fully enjoy the benefits of LBP. Further innovation can be made to meet such inspiring target.

– PDC enjoys the general form to organize the encoding of pixel differences, making it possible to incorporate any LBP variant to the convolutional module. While only the orignial LBP [28] and ELBP variants [22] are now explored and show positive effect on the task of edge or contour detection [32,41], we believe more LBP variants can be considered to tackle other different vision tasks.

– Like 3D-CDC, PDC can also be generalized to 3D scenes, where the temporal difference cues can be extracted in a more flexible way, rather than only considering the central differences.

Conclusion. This paper presents a review on LBP and the derived CNN modules proposed in the literature. We provide a comprehensive study on how traditional LBP and its variants inspired the design of modern CNN modules to boost the performance of vision models in the deep learning era. The associated applications in recent years are also elaborated. Finally, we list possible future directions. As an example of combining traditional vision descriptors with deep learning, we believe our review can also encourage more researchers to rethink the role of traditional descriptors when developing deep learning methods.

References

1. Ahonen, T., Hadid, A., Pietikäinen, M.: Face recognition with local binary patterns. In: Pajdla, T., Matas, J. (eds.) ECCV 2004. LNCS, vol. 3021, pp. 469–481. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-24670-1_36

2. Ahonen, T., Pietikäinen, M.: Image description using joint distribution of filter bank responses. Pattern Recogn. Lett. **30**(4), 368–376 (2009)

3. Calonder, M., Lepetit, V., Ozuysal, M., Trzcinski, T., Strecha, C., Fua, P.: Brief: Computing a local binary descriptor very fast. IEEE Trans. Pattern Anal. Mach. Intell. **34**(7), 1281–1298 (2011)

4. Cao, Y., et al.: A deep thermal-guided approach for effective low-light visible image enhancement. Neurocomputing **522**, 129–141 (2022)

5. Chen, H., Tang, H., Shi, H., Peng, W., Sebe, N., Zhao, G.: Intrinsic-extrinsic preserved gans for unsupervised 3d pose transfer. In: ICCV, pp. 8630–8639 (2021)

6. Cong, H., et al.: Image quality assessment with gradient siamese network. In: CVPR, pp. 1201–1210 (2022)

7. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: a large-scale hierarchical image database. In: CVPR (2009)

8. Doshi, N.P., Schaefer, G.: A comprehensive benchmark of local binary pattern algorithms for texture retrieval. In: ICPR, pp. 2760–2763. IEEE (2012)

9. Gu, J., et al.: Ntire 2022 challenge on perceptual image quality assessment. In: CVPR, pp. 951–967 (2022)

10. Guo, Y., Wang, H., Hu, Q., Liu, H., Liu, L., Bennamoun, M.: Deep learning for 3d point clouds: a survey. IEEE Trans. Pattern Anal. Mach. Intell. **43**(12), 4338–4364 (2021). https://doi.org/10.1109/TPAMI.2020.3005434

11. Guo, Z., Zhang, L., Zhang, D.: A completed modeling of local binary pattern operator for texture classification. IEEE Trans. Image Process. **19**(6), 1657–1663 (2010)

12. Heikkila, M., Pietikainen, M.: A texture-based method for modeling the background and detecting moving objects. IEEE Trans. Pattern Anal. Mach. Intell. **28**(4), 657–662 (2006)

13. Heikkilä, M., Pietikäinen, M., Schmid, C.: Description of interest regions with center-symmetric local binary patterns. In: Kalra, P.K., Peleg, S. (eds.) ICVGIP 2006. LNCS, vol. 4338, pp. 58–69. Springer, Heidelberg (2006). https://doi.org/10.1007/11949619_6

14. Heikkilä, M., Pietikäinen, M., Schmid, C.: Description of interest regions with local binary patterns. Pattern Recogn. **42**(3), 425–436 (2009)

15. Hubara, I., Courbariaux, M., Soudry, D., El-Yaniv, R., Bengio, Y.: Binarized neural networks. In: NeurIPS, pp. 4107–4115 (2016)

16. Juefei-Xu, F., Naresh Boddeti, V., Savvides, M.: Local binary convolutional neural networks. In: CVPR, pp. 19–28 (2017)

17. Kellokumpu, V., Zhao, G., Pietikäinen, M.: Human activity recognition using a dynamic texture based method. In: BMVC. vol. 1, p. 2 (2008)

18. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: NeurIPS, pp. 1106–1114 (2012)

19. Liu, L., Fieguth, P., Guo, Y., Wang, X., Pietikäinen, M.: Local binary features for texture classification: taxonomy and experimental study. Pattern Recogn. **62**, 135–160 (2017)

20. Liu, L., Fieguth, P., Wang, X., Pietikäinen, M., Hu, D.: Evaluation of LBP and deep texture descriptors with a new robustness benchmark. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9907, pp. 69–86. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46487-9_5

21. Liu, L., Lao, S., Fieguth, P.W., Guo, Y., Wang, X., Pietikäinen, M.: Median robust extended local binary pattern for texture classification. IEEE Trans. Image Process. **25**(3), 1368–1381 (2016)

22. Liu, L., Zhao, L., Long, Y., Kuang, G., Fieguth, P.: Extended local binary patterns for texture classification. Image Vis. Comput. **30**(2), 86–99 (2012)

23. Liu, W., Su, Z., Liu, L.: Beyond vanilla convolution: Random pixel difference convolution for face perception. IEEE Access **9**, 139248–139259 (2021)

24. Miao, S., Hou, Y., Gao, Z., Xu, M., Li, W.: A central difference graph convolutional operator for skeleton-based action recognition. IEEE Trans. Circuits Syst, Video Technol (2021)

25. Nanni, L., Brahnam, S., Lumini, A.: A local approach based on a local binary patterns variant texture descriptor for classifying pain states. Expert Syst. Appl. **37**(12), 7888–7894 (2010)

26. Nanni, L., Lumini, A., Brahnam, S.: Local binary patterns variants as texture descriptors for medical image analysis. Artificial Intell. Med. **49**(2), 117–125 (2010)

27. Ojala, T., Pietikäinen, M., Harwood, D.: A comparative study of texture measures with classification based on featured distributions. Pattern Recogn. **29**(1), 51–59 (1996)

28. Ojala, T., Pietikainen, M., Maenpaa, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Trans. Pattern Anal. Mach. Intell. **24**(7), 971–987 (2002)

29. Pietikäinen, M., Ojala, T., Xu, Z.: Rotation-invariant texture classification using feature distributions. Pattern Recogn. **33**(1), 43–52 (2000)

30. Satpathy, A., Jiang, X., Eng, H.L.: Lbp-based edge-texture features for object recognition. IEEE Trans. Image Process. **23**(5), 1953–1964 (2014)

31. Silvén, O., Niskanen, M., Kauppinen, H.: Wood inspection with non-supervised clustering. Machine Vision Appl. **13**(5), 275–285 (2003)
32. Su, Z., et al.: Pixel difference networks for efficient edge detection. In: ICCV, pp. 5117–5127 (2021)
33. Su, Z., Pietikäinen, M., Liu, L.: BIRD: learning binary and illumination robust descriptor for face recognition. In: BMVC (2019)
34. Su, Z., Welling, M., Liu, L., Pietikäinen, M.: Svnet: Where so (3) equivariance meets binarization on point cloud representation. In: 3DV (2022)
35. Tan, H., Wu, S., Pi, J.: Semantic diffusion network for semantic segmentation. In: Oh, A.H., Agarwal, A., Belgrave, D., Cho, K. (eds.) NeurIPS (2022)
36. Tan, X., Triggs, B.: Enhanced local texture feature sets for face recognition under difficult lighting conditions. IEEE Trans. Image Process. **19**(6), 1635–1650 (2010)
37. Trefnỳ, J., Matas, J.: Extended set of local binary patterns for rapid object detection. In: Computer Vision Winter workshop, pp. 1–7 (2010)
38. Wan, Z., Wang, Y., Tan, G., Cao, Y., Zha, Z.J.: S2n: Suppression-strengthen network for event-based recognition under variant illuminations. In: ECCV, pp. 716–733. Springer (2022). https://doi.org/10.1007/978-3-031-20062-5_41
39. Wang, X., Han, T.X., Yan, S.: An hog-lbp human detector with partial occlusion handling. In: ICCV, pp. 32–39. IEEE (2009)
40. Wu, X., Ma, D., Qu, X., Jiang, X., Zeng, D.: Depth dynamic center difference convolutions for monocular 3d object detection. Neurocomputing **520**, 73–81 (2022)
41. Xu, C., Wang, J., Tao, J., Zhang, J., Zheng, P.: A knowledge augmented deep learning method for vision-based yarn contour detection. J. Manufact. Syst. **63**, 317–328 (2022)
42. Yang, J., Li, A., Xiao, S., Lu, W., Gao, X.: Mtd-net: learning to detect deepfakes images by multi-scale texture difference. IEEE Trans. Inf. Forensics Secur. **16**, 4234–4245 (2021)
43. Ying, X., Wang, Y., Wang, L., Sheng, W., Liu, L., Lin, Z., Zhou, S.: Local motion and contrast priors driven deep network for infrared small target superresolution. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. **15**, 5480–5495 (2022)
44. Yu, Z., et al.: Multi-modal face anti-spoofing based on central difference networks. In: CVPR Workshops, pp. 650–651 (2020)
45. Yu, Z., Qin, Y., Zhao, H., Li, X., Zhao, G.: Dual-cross central difference network for face anti-spoofing. In: Zhou, Z. (ed.) IJCAI, pp. 1281–1287 (2021)
46. Yu, Z., Wan, J., Qin, Y., Li, X., Li, S.Z., Zhao, G.: Nas-fas: Static-dynamic central difference network search for face anti-spoofing. IEEE Trans. Pattern Anal. Mach. Intell. **43**(9), 3005–3023 (2020)
47. Yu, Z., et al.: Searching central difference convolutional networks for face anti-spoofing. In: CVPR, pp. 5295–5305 (2020)
48. Yu, Z., et al.: Searching multi-rate and multi-modal temporal enhanced networks for gesture recognition. IEEE Trans. Image Process. **30**, 5626–5640 (2021)
49. Zabihi, S., Tavakoli, H.R., Borji, A., Mansoori, E.: A compact deep architecture for real-time saliency prediction. Signal Process. Image Commun. **104**, 116671 (2022)
50. Zhang, J., Su, Z., Liu, L.: Median pixel difference convolutional network for robust face recognition. In: BMVC (2022)
51. Zhang, W., Shan, S., Gao, W., Chen, X., Zhang, H.: Local gabor binary pattern histogram sequence (lgbphs): A novel non-statistical model for face representation and recognition. In: ICCV. vol. 1, pp. 786–791. IEEE (2005)
52. Zhang, X., Liu, L., Xie, Y., Chen, J., Wu, L., Pietikainen, M.: Rotation invariant local binary convolution neural networks. In: ICCV Workshops, pp. 1210–1219 (2017)
53. Zhao, G., Ahonen, T., Matas, J., Pietikainen, M.: Rotation-invariant image and video description with local binary pattern features. IEEE Trans. Image Process. **21**(4), 1465–1477 (2012). https://doi.org/10.1109/TIP.2011.2175739

54. Zhao, G., Pietikainen, M.: Dynamic texture recognition using local binary patterns with an application to facial expressions. IEEE Trans. Pattern Anal. Mach. Intell. **29**(6), 915–928 (2007)
55. Zhao, Y., Zou, B., Yang, F., Lu, L., Belkacem, A.N., Chen, C.: Video-based physiological measurement using 3d central difference convolution attention network. In: IEEE International Joint Conference on Biometrics (IJCB), pp. 1–6. IEEE (2021)