# Local Binary Pattern
# (LBP)

Antara Chansarkar 0801CS171013
Jahanvi Sisodiya 0801CS171029

# Contents

# Chapter 1

# Introduction

## 1. Feature Extraction

Feature extraction is a process that identifies important features or attributes of the data. It increases the accuracy of learned models by extracting features from the input data. This phase of the general framework reduces the dimensionality of data by removing the redundant data.

## 2. Local Binary Pattern

Local binary pattern (LBP) is a nonparametric descriptor, which efficiently summarizes the local structures of images. In recent years, it has aroused increasing interest in many areas of image processing and computer vision and has shown its effectiveness in a number of applications, in particular for facial image analysis, including tasks as diverse as face detection, face recognition, facial expression analysis, and demographic classification. Derived from a general definition of texture in a local neighbourhood, LBP is defined as a grayscale invariant texture measure and is a useful tool to model texture images. LBP later has shown excellent performance in many comparative studies, in terms of both speed and discrimination performance. The original LBP operator labels the pixels of an image by thresholding the 3 x 3 neighbourhood of each pixel with the value of the central pixel and concatenating the results binomially to form a number. An LBP can also be considered as the concatenation of the binary gradient directions, and is called a *micropattern*. The histograms of these micropatterns contain information of the distribution of the edges, spots, and other local features in an image. LBP has been successfully used for face recognition. Different from statistic learning methods tuning a large number of parameters, the LBP method is very efficient due to its easy-to-compute feature extraction operation and simple matching strategy.

# Chapter 2

# Mathematical Formulation

## 1. Formulation

The original LBP operator labels the pixels of an image by thresholding the 3 x 3 neighbourhood of each pixel with the value of the central pixel and concatenating the results binomially to form a number. The thresholding function f(.,.) for the basic LBP can be formally represented as

$$f\left(I(Z_0), I(Z_i)\right) = \begin{cases} 0, & if\ I(Z_i) - I(Z_0) \leq \text{threshold} \\ 1, & if\ I(Z_i) - I(Z_0) > \text{threshold} \end{cases}, \quad i = 1, 2, \cdots, 8 \quad (1)$$

Formally, given a pixel at ($x_c$ , $y_c$ ), the resulting LBP can be expressed in decimal form as follows:

$$\mathrm{LBP}_{P,\,R}(x_c, y_c) = \sum_{P=0}^{P-1} s(i_P - i_c)2^P$$

where $i_c$ and $i_P$ are, respectively, gray-level values of the central pixel and $P$ surrounding pixels in the circle neighbourhood with a radius $R$, and function $s(x)$ is defined as

$$s(x) = \begin{cases} 1, & \text{if } x \geq 0 \\ 0, & \text{if } x < 0. \end{cases}$$

From the aforementioned definition, the basic LBP operator is invariant to monotonic gray-scale transformations, which preserve pixel intensity order in the local neighbourhoods. The histogram of LBP labels calculated over a region can be exploited as a texture descriptor.

# Chapter 3

# Algorithm

In this section, we give an overview of the LBP algorithm:

## 1. LBP algorithm

Algorithm:

**Input :** image in .jpg format

**Output :** LBP matrix of the image

1. Convert image into grid of pixels, represented by a matrix.

2. Select a centre pixel and examine its neighbouring pixels.

3. Set threshold value as centre pixel value.

4. Compare threshold value with each neighbouring pixels and assign binary number according to the following condition:

$$f\left(I(Z_0), I(Z_i)\right) = \begin{cases} 0, & if\ I(Z_i) - I(Z_0) \leq \text{threshold} \\ 1, & if\ I(Z_i) - I(Z_0) > \text{threshold} \end{cases}, \quad i = 1, 2, \cdots, 8 \quad \textbf{(1)}$$

where $Z_o$ represents central pixel and $Z_i$ represents neighbouring pixels.

5. Read the binary values in clockwise or anti-clockwise order and convert the 8-bit binary number into decimal number.

6. Replace the centre pixel value by the decimal number.

7. Repeat the above steps for every pixel.

# Chapter 4

# Documentation of API

## 1. Package organization

from feature_extraction import lbp

## 2. Methods

**def lbp(photo)**

Returns LBP of an image

*Parameters:*

Photo**:** Input image

**def assign_bit(picture, x, y, c)**

To assign bits to the neighbouring pixels.

*Parameters:*

picture: input image
x: row co-ordinate
y: column co-ordinate
c: centre pixel

**def local_bin_val(picture, x, y)**

To calculate LBP value of a pixel.

*Parameters:*

picture: input image
x: row co-ordinate of centre pixel
y: column co-ordinate of centre pixel

# Chapter 5

# Example

## 1. Example 1

```
photo = cv2.imread("/content/fruits.jpg",1)
img_lbp=lbp(photo)
```

**Output:**

```
array([[137, 249, 247, ...,   0,  45, 184],
       [  0, 112, 243, ..., 207,  94,   0],
       [128, 224, 241, ...,  14,  57, 176],
       ...,
       [ 14,  30,   1, ...,  19,   2,   0],
       [  2,   1, 160, ..., 177, 231, 192],
       [129, 225, 224, ..., 225, 227, 128]], dtype=uint8)
```

## 2. Example 2

```
photo = cv2.imread("/content/dog.jpg",1)
img_lbp=lbp(photo)
```

**Output:**

```
array([[ 62,  62,  58, ...,  62,  62,  56],
       [255, 251, 241, ..., 255, 255, 248],
       [243, 225, 255, ..., 255, 255, 248],
       ...,
       [255, 131,   6, ..., 131, 231, 232],
       [121, 255, 207, ..., 255, 199,   0],
       [224, 227, 131, ..., 227, 131, 128]], dtype=uint8)
```

# Chapter 6

# Learning Outcome

- Capacity to integrate practical and theoretical knowledge to use local binary pattern's discriminative power and computational simplicity for texture recognition.
- Analysed and evaluated higher mathematical concepts with the ability to clearly implement and present the conclusions and the knowledge behind it.

# Appendix A

# References

1. Di Huang, *Student Member, IEEE*, Caifeng Shan, *Member, IEEE*, Mohsen Ardabilian, Yunhong Wang, *Member, IEEE*, and Liming Chen, *Member, IEEE* TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART C: APPLICATIONS AND REVIEWS, VOL. 41, NO. 6, NOVEMBER 2011

2. Song Ke-Chen, Yan Yun-Hui, Chen Wen-Gui, Zhang Xu. Research and perspective on local binary pattern. Acta Automatica Sinica, 2013