

Local Feature Extraction

Introductio

.

scale-space

Detect scale-invariant interest points

interest points from scale-space

extrema

Orientatio

Compute

descripto

CHDE

 $_{
m LBP}$

Encodin formula

Calculation process

Local Feature Extraction SIFT, SURF and LBP

常琳

CVBIOUC

http://vision.ouc.edu.cn/~zhenghaiyong

March 18, 2016



Contents

Local Feature Extraction

Introductio

51F .

Construct scale-space Detect scale-invariant interest points from scale-space extrema

Compute descriptor Match imag

SURI

LBP

Encoding formula Calculation process 1 Introduction

2 SIFT

- Construct scale-space
- Detect scale-invariant interest points from scale-space extrema
- Orientation match
- Compute descriptor
- Match image descriptors

3 SURF

4 LBP

- Encoding formula
- Calculation process



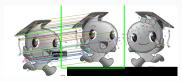
Introduction

Feature Extraction

Introduction



SURF match



SIFT match



SIFT

Local Feature Extraction

Introductio

SIFT

Construct scale-space Detect scale-invariant interest points from scale-space extrema Orientation match

descriptor Match ims

descripto

SURI

LBP

Encoding formula Calculation process

SIFT(Scale Invariant Feature Transform):

- An algorithm in computer vision to detect and describe local features in images.
- Published by David Lowe in 1999.
- Accuracy, stability, scale and rotational invariance.



Steps

Local Feature Extraction

Introduction

SIFT

Construct scale-space Detect scale-invariant interest points from scale-space extrema

Orientation match Compute descriptor

Match ima descriptors

SURI

LBP

Encoding formula Calculation process

Steps:

- 1 Construct scale-space.
- 2 Detect scale-invariant interest points from scale-space extrema.
- 3 Orientation match.
- 4 Compute descriptor.
- 5 Match image descriptors.



Local Feature Extraction

Introduction

SIFT

Construct scale-space Detect scale-invariant interest points from scale-space extrema

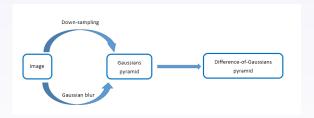
Orientation match Compute

descriptor Match ima descriptors

SURI

LBP

Encoding formula Calculation process Interest points are obtained from a difference-of-Gaussians pyramid.



Construct scale-space

Smoothed image values:

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y)$$
 (1)



Local Feature Extraction

Introductio

SIF".

Construct scale-space Detect scale-invariant interest points from scale-space extrema Orientation

Compute descriptor Match imag

SURF

LBF

Encoding formula Calculation process Gaussian kernels:

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x-m/2)^2 + (y-n/2)^2}{2\sigma^2}}$$
(2)

The difference-of-Gaussians operator:

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y)$$
$$= L(x, y, k\sigma) - L(x, y, \sigma)$$
(3)

In fact, we obtained the difference-of-Gaussians pyramid by taking subtraction between the adjacent two layers.



Local Feature Extraction

Introductio

SIFT

Construct scale-space Detect scale-invarian interest point from scale-space extrema

Orientatio match

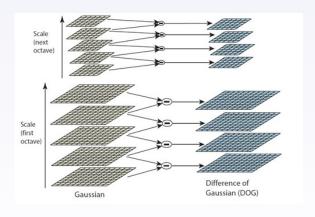
Compute

Match im

SURF

LBP

Encoding formula Calculation process



Contrust difference-of-Gaussians pyramid



Detect scale-space extrema

Local Feature Extraction

Compare the detecting point with other points in a 3*3*3 neighbourhood.

introduction.

SIF'I

scale-spa

Detect scale-invariar interest point from

scale-space extrema

Orientatio

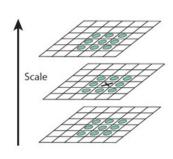
match

descripto

SURE

 $_{
m LBP}$

Encoding formula Calculation process



Extrema detection



Local Feature Extraction

Introduction

SIF

Construction scale-spa

Detect

scale-invarian interest point from scale-space

Orientati

match

Compute descriptor

Match ima

SURE

LBP

formula Calculation process The scale coordinate of keypoints:

$$\sigma(o,s) = \sigma_0 2^{o + \frac{s}{S}}$$
 $o = 0, 1, \dots, O - 1, s = 0, \dots, S + 2$ (4)

The scale of a particular layer:

$$\sigma_{oct}(s) = \sigma_0 2^{\frac{s}{S}} \quad s = 0, \dots, S+2$$
 (5)



Feature Extraction

Local

troductio

Construct cale-spac

Detect scale-invarian interest point from scale-space extrema

Orientatio match

descriptor

Match imag
descriptors

SUI

LBI

Encoding formula Calculation process The Taylor expansion of D(X) is as follows:

$$D(X) = D(X) + \frac{\partial D^{T}}{\partial X}X + \frac{1}{2}X^{T}\frac{\partial^{2}D}{\partial X^{2}}X \quad X = (x, y, \sigma)^{T} \quad (6)$$

Deal with the derivative and let it equal 0. We can obtain the offset of extrema:

$$X_1 = -\frac{\partial^2 D^{-1}}{\partial X_1^2} \frac{\partial D}{\partial X_1} \tag{7}$$

and

$$D(X_1) = D + \frac{\partial D^T}{\partial X_1} X_1 \tag{8}$$

If $|D(X_1)| < 0.3$, throw it.



Local Feature Extraction

Introductio

SIF

Construc

Detect

scale-invariantinterest pointfrom

extrema

Orientat

Compute

descriptor Match ima

SURF

LBP

Encoding formula Calculation process To suppress such points, which will be less useful for matching. We formulate the Hessian matrix:

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix}$$

$$Tr(H) = D_{xx} + D_{yy} = \alpha + \beta$$
 (9)

$$Det(H) = D_{xx}D_{yy} - (D_{xy})^2 = \alpha\beta \tag{10}$$



Local Feature Extraction

Introduction

SIFT

Construc

scale-spa

scale-invariant interest points from scale-space

Orientatio

Compute

descriptor

SURF

LBP

Encoding formula Calculation process Assume that α is bigger than β , and $\alpha = r\beta$.

$$\frac{Tr(H)^2}{Det(H)} = \frac{(\alpha_+\beta)^2}{\alpha\beta} = \frac{(r\beta+\beta)^2}{r\beta^2} = \frac{(r+1)^2}{r}$$
(11)

If $\frac{Tr(H)^2}{Det(H)} > \frac{(r+1)^2}{r}$, throw it.

(Lowe recommended r = 10)



Local Feature Extraction

Introductio

SIF"I

scale-space
Detect
scale-invariatinterest point

extrema Orientation

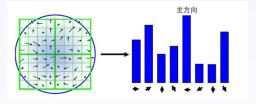
match

Match ima

SURF

LBP

Encoding formula Calculation process Select a window $(r = 3 * 1.5\sigma_{oct})$ around the interest point.



Orientation histogram

With 36 bins in the histogram.



Local Feature Extraction

Introductio

SIFT

Construct scale-space Detect scale-invariant interest points from scale-space extrema

match

descriptor
Match ima
descriptors

SURI

LBP

Encoding Formula Calculation process Using

m(x, y)

$$= \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y-1))^2}$$
(12)

$$\theta = \tan^{-1}((L(x, y+1) - L(x, y-1))/(L(x+1, y) - L(x-1, y)))$$
(13)

to find the dominant orientation.

Compute the orientation histogram based on every gradient direction $\theta(x, y)$.

Peak is the orientation.



Local Feature Extraction

Introduction

SIF'I

Construct scale-space Detect scale-invariant interest points from scale-space extrema

Orientatio: match

descriptor Match im

SURI

 $_{
m LBP}$

Encoding formula Calculation process Multiple peaks are accepted if the height of secondary peaks is above 80 % of the height of the highest peak. Express it approximately as the quadratic function curve to find actual location (the highest point).



Determine the local image region

Local Feature Extraction

Introduction

SIFT

Construct scale-space Detect scale-invariant interest points from scale-space extrema

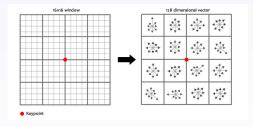
extrema Orientation match

descriptor Match im

SURI

 $_{
m LBP}$

Encoding formula Calculation process Divide the neighborhood into 4*4 regions. Every region is a seed point.



Orientation histogram



Rotate the coordinate

Local Feature Extraction

Introduction

SIF".

scale-space
Detect
Scale-invarian
interest point
from
scale-space
extrema

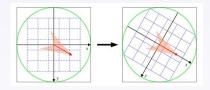
Compute descriptor

Match im descriptor

SURI

LBP

Encoding formula Calculation process



The new coordinate is based on the orientation of interset point.

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$



Allocate sampling points

Local Feature Extraction

Introduction

SIFT

Construct scale-space Detect scale-invariant interest points from scale-space extrema Orientation match

descriptor

Match imag
descriptors

SURI

LBF

Encoding formula Calculation process Rotated sampling points was allocated to 4*4 regions. The new coordinate is:

$$\begin{bmatrix} x'' \\ y'' \end{bmatrix} = \frac{1}{3\sigma_{-oct}} \begin{bmatrix} x' \\ y' \end{bmatrix} + \frac{d}{2}$$

The gradient can be computed by Gaussian weighted model as $\sigma = 0.5d$:

$$w = m(a+x, b+y)e^{-\frac{(x')^2 + (y')^2}{2x(0.5d)^2}}$$
(14)

a, b is the coordinate in Gaussian pyramid.



Compute the gradient of 8 directions by interpolation

Local Feature Extraction

Introductio

SIF

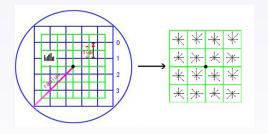
Construct scale-space Detect scale-invariant interest points from scale-space extrema

Compute

SURI

LBP

Encoding formula Calculation process



Linear interpolation is used on (x'', y'') (the red points) for computing its contribution to every seed point.



Compute the gradient of 8 directions by interpolation

Local Feature Extraction

Introduction

SIF

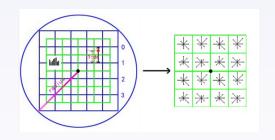
Construct scale-space Detect scale-invariant interest points from scale-space extrema

Compute descriptor

SURF

LBF

Encoding formula Calculation process



Its contributions to

[0,3]: dr [1,3]: 1-dr

Column 2: dc Column 3: 1 - dc

Neighbor directions: do and 1 - do.



Compute the gradient of 8 directions by interpolation

Local Feature Extraction

Introduction

SIFT

Construct scale-space Detect scale-invariant interest points from scale-space extrema

Compute descriptor

descript

SURF

LBP

Encoding formula Calculation process The eventual gradient magnitude added on every direction is:

$$weight = w * dr^{k} * (1 - dr)^{1-k} * dc^{m} * (1 - dc)^{1-m} * do(1 - do)^{1-n}$$
(15)

k, m, n = 0 or 1.



Normalize

Local Feature Extraction

Introduction

SIFT

Construct scale-space Detect scale-invariant interest points from scale-space extrema Orientation match

Compute descriptor Match ims

SURI

LBP

Encoding formula Calculation process Taken together, the local histograms computed at all the 4×4 grid points and with 8 quantized directions lead to an image descriptor $H=(h_1,h_2,\ldots,h_{128})$ with $4\times4\times8=128$ dimensions for each interest point.

Normalize H to avoid the effect of illumination:

$$l_i = \frac{h_i}{\sqrt{\sum_{j=1}^{128} h_j}} \tag{16}$$



Match image descriptors

Local Feature Extraction

Introduction

SIF"I

Construct scale-space Detect scale-invariant interest points from scale-space extrema

Orientation match Compute

Match image

SURF

$_{ m LBP}$

Encoding formula Calculation process $R_i = (r_{i1}, r_{i2}, \dots, r_{i128})$: Descriptors from model image.

 $S_i = (s_{i1}, s_{i2}, \dots, s_{i128})$: Descriptors from another image.

$$d(R_i, S_i) = \sqrt{\sum_{j=1}^{128} (r_{ij} - s_{ij})^2}$$
: Distance between R_i and S_i .

If $\frac{d_{min}(R_i,S_i)}{d_{the_second_min}(R_i,S_j)} < Threshold$, they matched.



SURF

Local Feature Extraction

Introduction

SIFT

Construct scale-space Detect scale-invariant interest points from scale-space extrema Orientation match

descriptor

Match imag
descriptors

SURF

LBF

Encoding formula Calculation process

SURF(Speeded Up Robust Features):

- An algorithm in computer vision to detect and describe local features in images.
- First presented by Herbert Bayet al. in 2006.
- Fast than SIFT, accuracy, stability, scale and rotational invariance.



Steps

Local Feature Extraction

Introduction

SIFT

Construct scale-space Detect scale-invariant interest points from scale-space extrema

Compute descriptor Match imag

SURI

LBF

Encoding formula Calculation process

Steps:

- **1** Construct scale-space.
- 2 Detect scale-invariant interest points from scale-space extrema.
- 3 Orientation match.
- 4 Compute descriptor.
- 5 Match image descriptors.



Steps

Local Feature Extraction

Introduction

SIF"I

Construct scale-space Detect scale-invariant interest points from scale-space extrema

Orientation match

Match ima

SURI

LBP

Encoding formula Calculation process

Steps:

- **1** Construct scale-space.
- 2 Detect scale-invariant interest points from scale-space extrema.
- 3 Orientation match.
- 4 Compute descriptor.
- **5** Match image descriptors.



Local Feature Extraction

Introductio

SIFT

Construc

scale-spa

Detect

interest poi

scale-space

extrema

Orientati

match

Compute

descripte

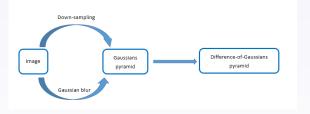
descriptors

SURF

LBP

formula

Calculation process



Construct scale-space



Local Feature Extraction

Introduction

SIF

Construct scale-space Detect scale-invariant interest points from scale-space extrema

extrema
Orientation
match
Compute

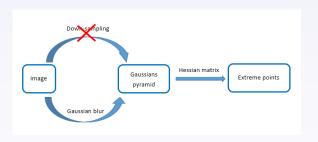
descriptor

Match imag
descriptors

SURI

LBF

Encoding formula Calculation process



Construct scale-space

The discriminant of Hessian matrix to detect extrema can be approximated as follows:

$$Det(Happrox) = D_{xx}D_{yy} - (0.9D_{xy})^2$$
 (17)

 D_{xx}, D_{xy}, D_{yy} are the convolution results of Gaussian kernels and original image.



Local Feature Extraction

Introduction

DIF.

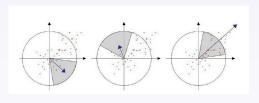
Construct scale-space Detect scale-invariant interest points from scale-space extrema Orientation

Compute descriptor Match image descriptors

SURI

LBF

Encoding formula Calculation process



Estimate the dominant orientation

Angle: $\frac{\pi}{3}$.

Radius: 6s (s is the scale at which the interest pointwas detected).

Calculate the Haar-wavelet responses in x and y direction. The maximum sum of all responses is the dominant orientation.



Compute descriptor

Feature Extraction

Local

oductio

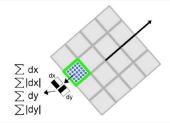
SIFT

Detect scale-space Detect scale-invariant interest points from scale-space extrema Orientation match

Match image descriptors

SUM

Encoding formula Calculatio process Construct a square region (20s) aligned to the selected orientation. 4*4 square sub-regions, 5*5 regularly spaced sample points.



Compute descriptor

We call dx the Haar wavelet response in horizontal direction and dy the Haar wavelet response in vertical direction. Vector $\mathbf{v} = (\sum dx, \sum |dx|, \sum dy, \sum |dy|)$

LBP

Local Feature Extraction

Introduction

SIFT

scale-space
Detect
scale-invariant
interest points
from
scale-space
extrema
Orientation
match
Compute
descriptor

SURF

LBP

Encoding formula Calculation process

LBP(Local Binary Pattern):

- One of the method about local information extraction. It reflects the gray value relationship between the pixel and points around it.
- Proposed by Ojala et al.
- LBP operator has significant effect in the description of texture feature extraction.



Encoding formula

Local Feature Extraction

Introduction

SIFT

Construct scale-space Detect scale-invariant interest points from scale-space extrema Orientation match

descriptor

Match imag
descriptors

SURI

LBP

Encoding formula Calculation process The encoding formula for any LBP operator is:

$$LBP_{P,R} = \sum_{i=0}^{p-1} s(g_i - g_c)2^i, s(x) = \begin{cases} 1 & x \ge 0\\ 0 & x < 0 \end{cases}$$
 (18)

P: The number of pixels in the (P, R) neighborhood.

R: The neighborhood radius.

 $g_i(i=1,2...P-1)$: The pixel values in the neighborhood with a threhold g_c .



Calculation process

Local Feature Extraction

Introductio

SIF"

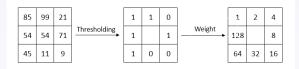
Construct scale-space Detect scale-invariant interest points from scale-space extrema Orientation

match
Compute
descriptor
Match imag

SURI

LBP

Encoding formula Calculation process



Basic LBP operator
$$(P = 8 \text{ and } R = 1)$$

The LBP operator= 1+2+8+64+128=203Count the ratio of LBP operator between 0 and 255 to a histogram.

Obtain a data with 256-dimensional features.



Calculation process

Local Feature Extraction Convert all the images size into 64*64 and each image is divided into 8*8 local regions.

Introductio

SIF"I"

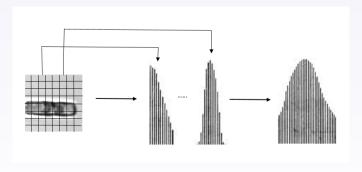
Construct scale-space Detect scale-invariant interest points from scale-space extrema

Orientation match Compute descriptor

descriptor

LBP

Encoding formula Calculation process



Global image description based on LBP

Concatenate the regional histograms to build a global histogram.