國立高雄第一科技大學
National Kaohsiung First University of Science and Technology

HOG: Histogram of Oriented Gradients

N. Dalal and B. Triggs

IEEE Intl. Conf. on Computer Vision and Pattern Recognition,

(CVPR) 2005

Speaker: Shih-Shinh Huang

October 12, 2018

Navneet Dalal and Bill Triggs, "Histograms of Oriented Gradients for Human Detection," *IEEE Intl. Conf. on CVPR*, pp. 886—893, 2005.

Outline

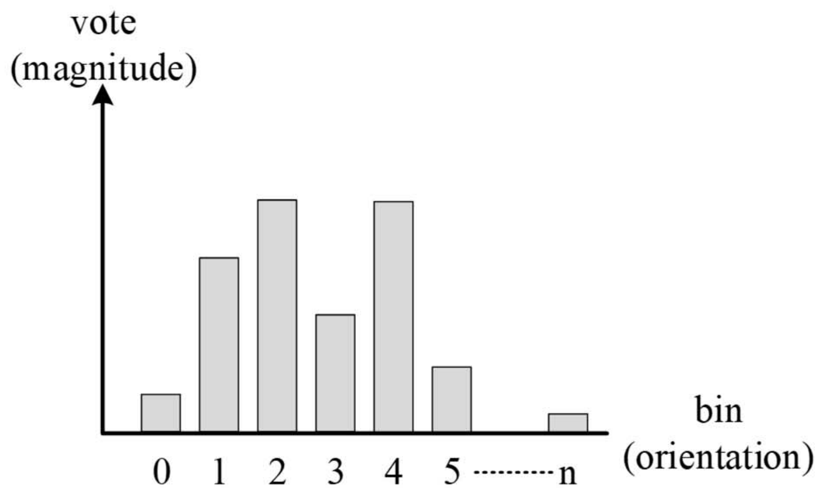
- Introduction
- Gradient Computation
- Orientation Quantization
- Block Description

Introduction

- Origin of HOG
  - Detecting pedestrians in images is an important topic in many applications.
  - Extracting effective features to describe pedestrian is critical in pedestrian detection.
  - HOG is a feature proposed for describing pedestrian appearance in visible images

Introduction

- About HOG
  - HOG is generally used to describe the texture of a rectangular block.
  - HOG is a form of histogram of oriented gradient



Gradient Computation

- Horizontal/Vertical Gradient
  - use centered derivative to compute horizontal $d_h(.)$ and vertical gradient $d_v(.)$

$$f'(x) = \lim_{h \to 0} \frac{f(x+h) - f(x-h)}{2h}$$

- $d_h(x, y) = I(x+1, y) - I(x-1, y)$

-1	0	+1

- $d_v(x, y) = I(x, y+1) - I(x, y-1)$

-1
0
+1

Gradient Computation

4	2	1	3	4	5
3	2	3	4	3	2
2	1	2	3	4	2
1	2	5	6	7	2
6	3	4	6	7	1
3	1	5	2	1	3

-1
0
+1

$d_v()$

-1	1	0	0
0	2	2	4
2	2	3	3
-1	0	-4	-6

$\Rightarrow 5 - 3 = 2$

-1	0	+1

$d_h()$

0	2	0	-2
0	2	2	-1
4	4	2	-4
-2	3	3	-5

$\Rightarrow 3 - 1 = 2$

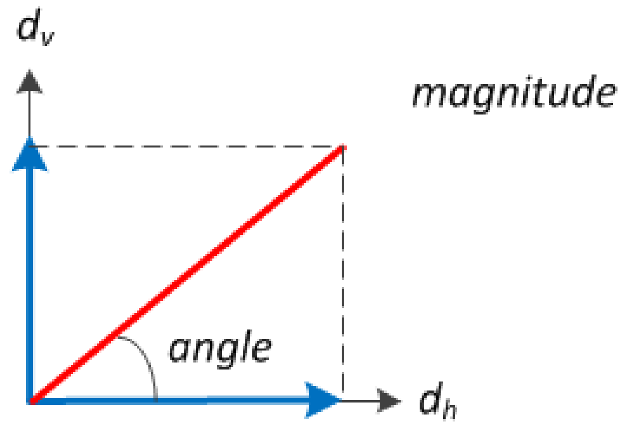
Gradient Computation

- Magnitude $mag(.)$ and Orientation $\theta(.)$

$$mag(x,y) = \sqrt{d_h(x,y)^2 + d_v(x,y)^2}$$

$$mag(x,y) \approx |d_h(x,y)| + |d_v(x,y)|$$

$$\theta(x,y) = \tan^{-1}\left(\frac{d_v(x,y)}{d_h(x,y)}\right)$$



Gradient Computation

$d_h()$

0	2	0	-2
0	2	2	-1
4	4	2	-4
-2	3	3	-5

$d_v()$

-1	1	0	0
0	2	2	4
2	2	3	3
-1	0	-4	-6

$mag(x, y)$

1	3	0	2
0	4	4	5
6	6	5	7
3	3	7	11

$\Rightarrow |4| + |2| = 6$

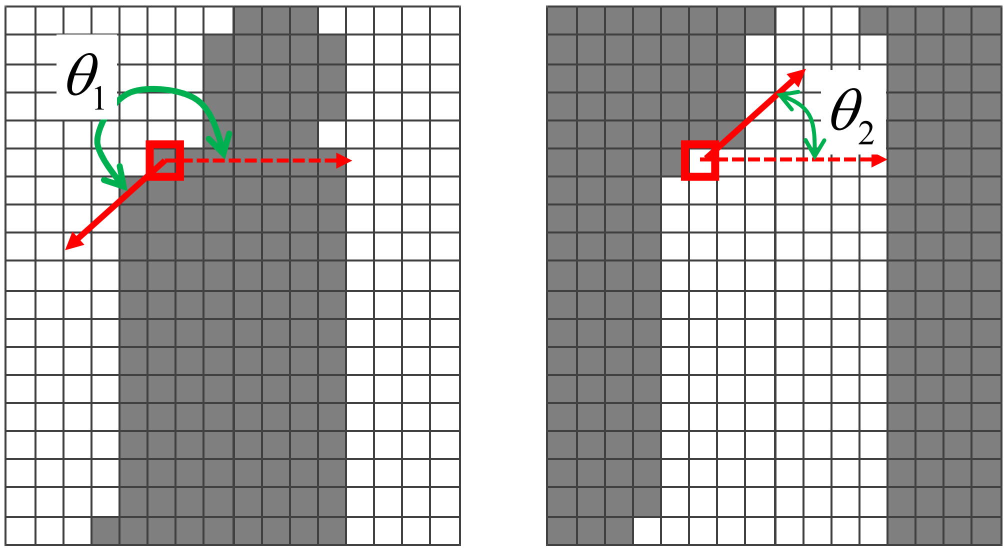
$\theta(x, y)$ (degree)

0	26	0	179
0	44	44	104
26	26	56	143
206	0	307	230

$\Rightarrow \tan^{-1}\left(\frac{2}{4}\right) \approx 26^\circ$

Orientation Quantization

- Polarity Change
  - make descriptor be invariant to the relative change between the background and foreground.



Orientation Quantization

- Polarity Change

$$\theta'(x,y)=\begin{cases}\theta(x,y) & \text{if } \theta(x,y)<180\\ \theta(x,y)-180 & \theta(x,y)\geq 180\end{cases}$$

0	26	0	179
0	44	44	104
26	26	56	143
206	0	307	230

$\theta(x,y)$

0	26	0	179
0	44	44	104
26	26	56	143
26	0	127	50

$\theta'(x,y)$

Orientation Quantization

- Bin Index Computation $bin(x, y)$
  - A bin spans over 20 degrees.
  - The range of orientation is divided into 9 bins.

0	26	0	179
0	44	44	104
26	26	56	143
26	0	127	50

$\theta'(x, y)$

$$bin(x, y) = \left\lfloor \frac{\theta(x, y)}{20} \right\rfloor$$

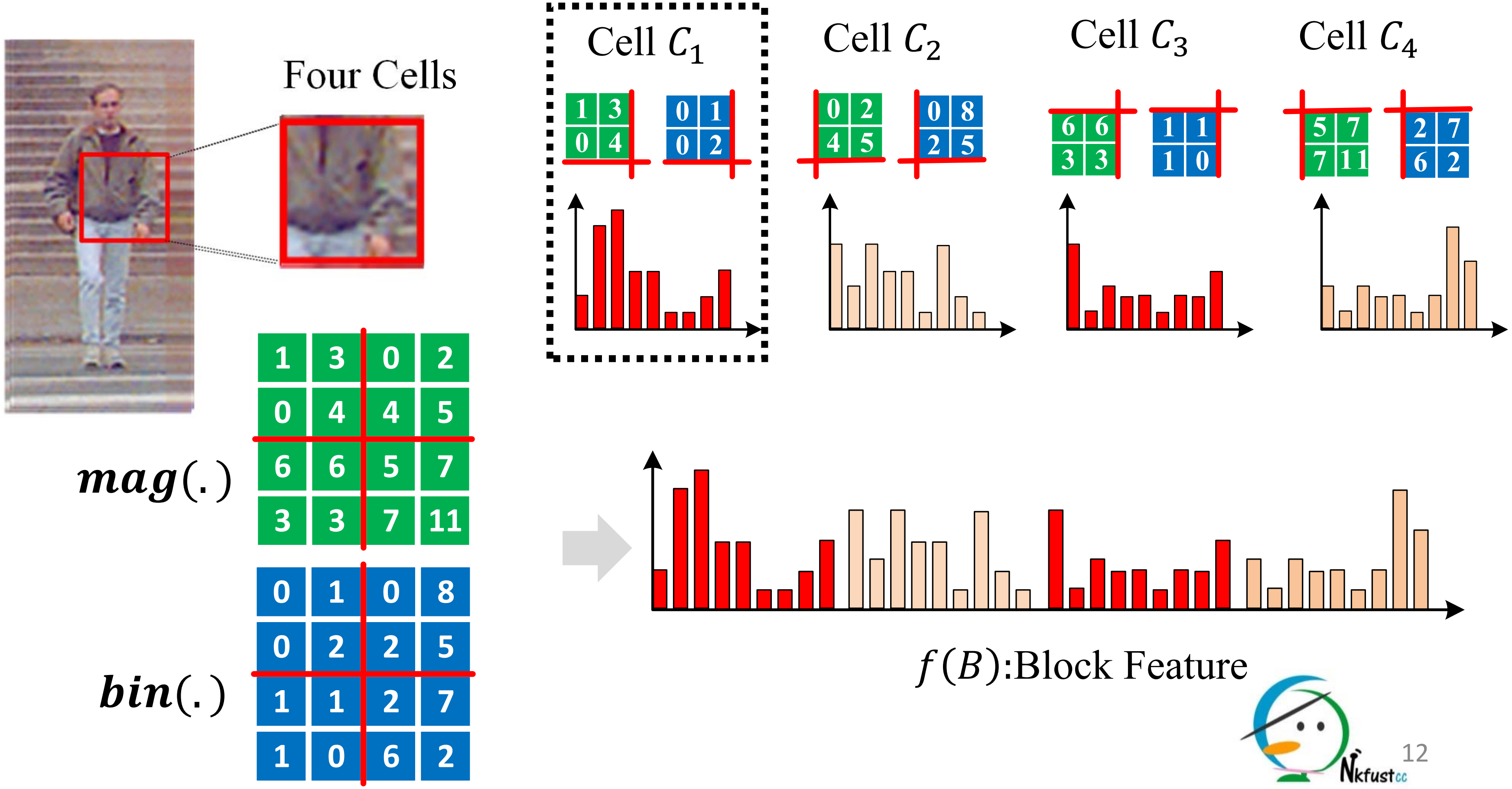
$$\left\lfloor \frac{44}{20} \right\rfloor = \lfloor 2.2 \rfloor = 2$$

0	1	0	8
0	2	2	5
1	1	2	7
1	0	6	2

$bin(x, y)$

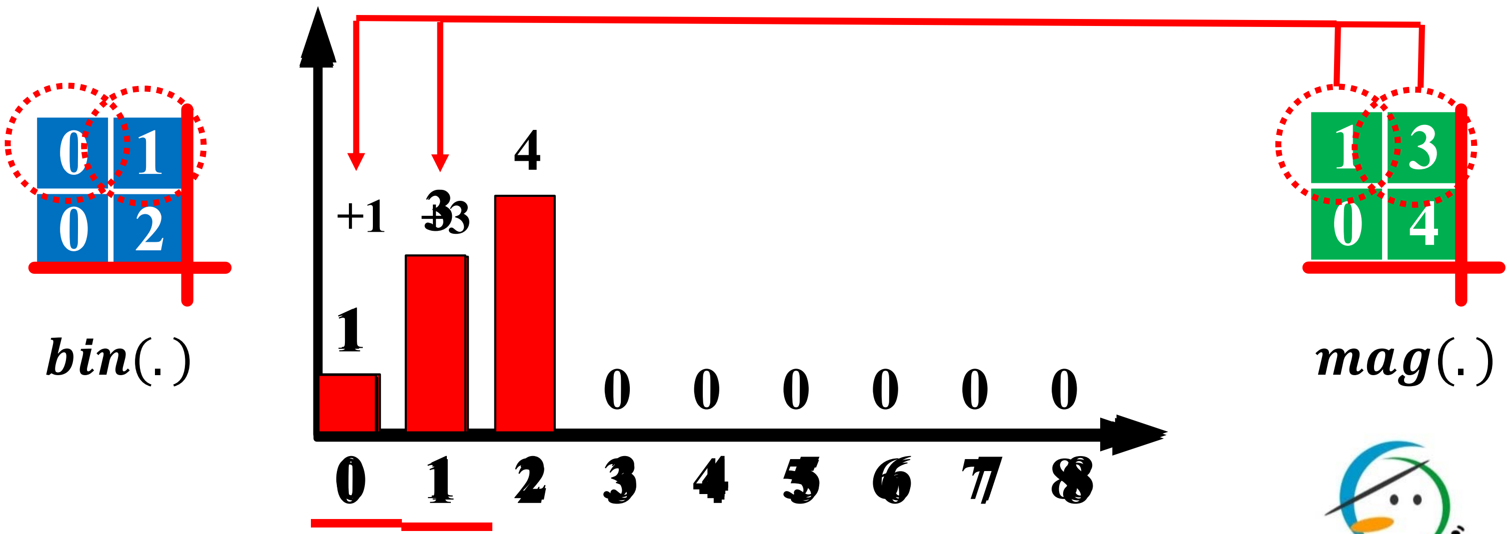
Block Description

- Overview



Block Description

- Cell Description
  - Form a 9-D cell descriptor by voting mechanism
    - $bin(.)$: voting index
    - $mag(.)$: voting weight



Block Description

- Concatenation and Normalization
  - concatenate the four 9-d feature vectors to form a 36-d block feature vector

$$f(B) = \{v_0, v_1, \ldots., v_{35}\}$$

  - normalize the block feature to unity

$$f(B) = \frac{1}{Z}\{v_0, v_1, \ldots., v_{35}\}$$

Normalization Term

L1-Norm: $Z = (|v_o| + |v_1| + \cdots |v_{35}|)$

L2-Norm: $Z = \sqrt{(v_o)^2 + (v_1)^2 + \ldots (v_{35})^2}$


thank you
© 1996, 2002 SANRIO CO., LTD.