Global descriptors: HoG

Course: Computer Vision

E. Francisco Roman-Rangel.
francisco.roman@itam.mx

Digital Systems Department.
Instituto Tecnológico Autónomo de México, ITAM.

August 24th, 2020.

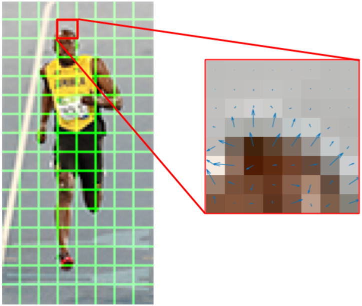
Last time

- Global image descriptors.
- Windowed Fourier Transform.
- Gabor filters.
- GIST.

Outline

Pixel orientation

- What does it mean?
- How come a pixel can have orientation?



2	3	4	4	3	4	2	2
5	11	17	13	7	9	3	4
11	21	23	27	22	17	4	6
23	99	165	135	85	32	26	2
91	155	133	136	144	152	57	28
98	196	76	38	26	60	170	51
165	60	60	27	77	85	43	136
71	13	34	23	108	27	48	110

Gradient Magnitude

80	36	5	10	0	64	90	73
37	9	9	179	78	27	169	166
87	136	173	39	102	163	152	176
76	13	1	168	159	22	125	143
120	70	14	150	145	144	145	143
58	86	119	98	100	101	133	113
30	65	157	75	78	165	145	124
11	170	91	4	110	17	133	110

Gradient Direction

[Image from learnopencv[1]].

[1] https://www.learnopencv.com/histogram-of-oriented-gradients/

Generalization from the notion of slope

$$m = \frac{\delta_y}{\delta_x} = \frac{y_2 - y_1}{x_2 - x_1}$$

m can be computed from the pixel values.

Q: What would m and δ_a mean physically?

From m, we can compute magnitude and orientation

Magnitude:

$$r = \sqrt{(\delta_x)^2 + (\delta_y)^2}.$$

Orientation:

$$\theta = \arctan\left(\frac{\delta_y}{\delta_x}\right).$$

Q: How do r and θ look like?

Generalization from the notion of slope

$$m = \frac{\delta_y}{\delta_x} = \frac{y_2 - y_1}{x_2 - x_1}$$

m can be computed from the pixel values.

Q: What would m and δ_a mean physically?

From m, we can compute magnitude and orientation

Magnitude:

$$r = \sqrt{(\delta_x)^2 + (\delta_y)^2}.$$

Orientation:

$$\theta = \arctan\left(\frac{\delta_y}{\delta_x}\right).$$

Q: How do r and θ look like?

A: They are matrices the same size as the input image.

Outline

Histogram of Oriented Gradients, HoG

Navneet Dalal and Bill Triggs. "Histogram of Oriented Gradients for Human Detection". *IEEE International Conference on Computer Vision Pattern Recognition (CVPR)*. 2005.

Gradient Computation I

Sobel border detector with horizontal and vertical kernels:

$$k_{\delta_x} = [-1, 0, 1] \qquad \text{and} \qquad k_{\delta_y} = [-1, 0, 1]^T$$

Q: How do we do for RGB images?

Gradient Computation I

Sobel border detector with horizontal and vertical kernels:

$$k_{\delta_x} = [-1, 0, 1] \qquad \text{and} \qquad k_{\delta_y} = [-1, 0, 1]^T$$

Q: How do we do for RGB images?
A: Several approaches are possible. Often, process each channel independently, and keep only the one with highest magnitude.

Cell: orientation histogram

Creation of a frequency histogram of orientations (default 9 bins).

Cell

Group of 8×8 pixels.

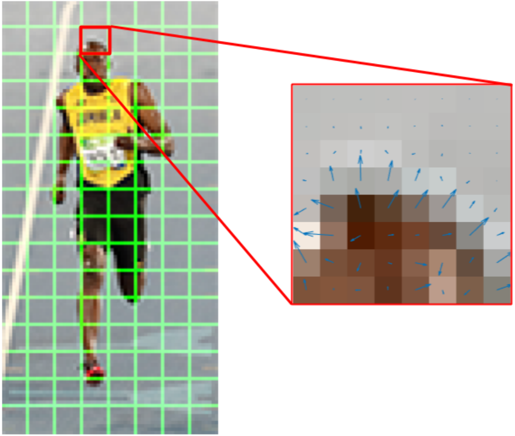
For each cell

- Count the number of pixel orientations within each bin.
- Use a weighted aggregation of elements (i.e., magnitude).
- Option: Split contribution between adjacent bins.

.

Cell: quantization I

Cells and gradients per pixel:



2	3	4	4	3	4	2	2
5	11	17	13	7	9	3	4
11	21	23	27	22	17	4	6
23	99	165	135	85	32	26	2
91	155	133	136	144	152	57	28
98	196	76	38	26	60	170	51
165	60	60	27	77	85	43	136
71	13	34	23	108	27	48	110

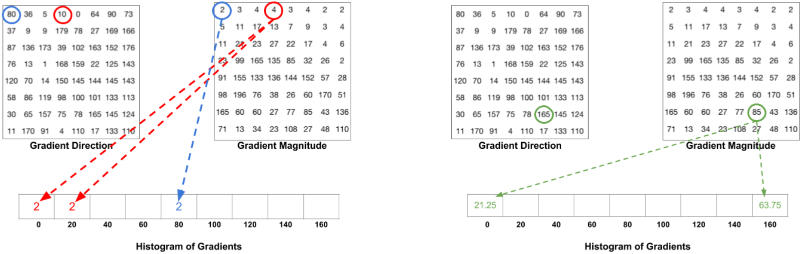
Gradient Magnitude

80	36	5	10	0	64	90	73
37	9	9	179	78	27	169	166
87	136	173	39	102	163	152	176
76	13	1	168	159	22	125	143
120	70	14	150	145	144	145	143
58	86	119	98	100	101	133	113
30	65	157	75	78	165	145	124
11	170	91	4	110	17	133	110

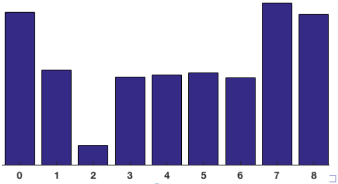
Gradient Direction

Cell: quantization II

Cell histogram construction:



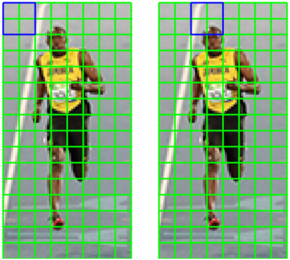
Final histogram per cell:



Blocks: normalization

Robustness against lighting changes.

- Gradients of an image are sensitive to overall lighting.
- Define blocks of 16×16 pixels, i.e., 2×2 cells (This is a 36-D vector).
- Divide each element by the magnitude of the full 36-D vector.



Q: What is the result of summing up all 36 elements of the vector after normalization?

Image representation, a.k.a, HoG descriptor

Concatenate vectors using an overlapping-block scheme:

- Slide the block definition by half its size, i.e., 8 pixels.
- For each slide, get the 36-D vector corresponding to the current block. Concatenate all vector-blocks.
- Overlap: redundancy.

Consider an input image of 128×64 pixels.
Q: What is the length of its HoG descriptor?

Image representation, a.k.a, HoG descriptor

Concatenate vectors using an overlapping-block scheme:

- Slide the block definition by half its size, i.e., 8 pixels.
- For each slide, get the 36-D vector corresponding to the current block. Concatenate all vector-blocks.
- Overlap: redundancy.

Consider an input image of 128×64 pixels.
Q: What is the length of its HoG descriptor?

A: Moving 8 pixels at a time, across a 128×64 space, gives 7 horizontal steps and 15 vertical steps, thus 105 overlapping blocks. This results in a $105 \times 36 = 3780$-D vector.

Outline

Classifier

Given a class label for each element in a set of known images (a.k.a., training set), create a mathematical model capable of assigning the right class label for new unseen images (a.k.a., query images or test set).

- Known images are often already indexed by their feature descriptors, e.g., HoG.

kNN

k-nearest neighbors (kNN) is the simplest classifier.

- Using the same image descriptor (as the one used for the training set), compute the mathematical representation for a query image.
- Compare the query descriptor against all descriptors in the training set.
- Find the class of the most similar element, and use its label as the prediction for the query image.

k in kNN

Using the closest element is known as 1NN or kNN ($k = 1$).

Evaluating different values of k might help avoid misclassification in some cases (outliers).

Odd values for k help solving ties.

k in kNN

Using the closest element is known as 1NN or kNN ($k = 1$).

Evaluating different values of k might help avoid misclassification in some cases (outliers).

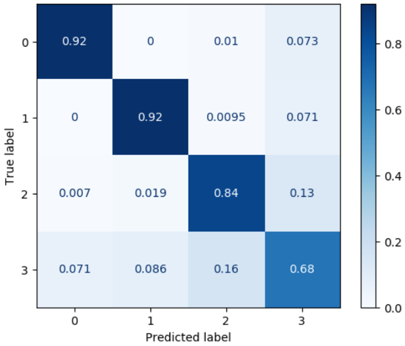
Odd values for k help solving ties.

Most commonly, $k = 1$ works well.

Outline

Confusion matrix

Shows rate of right and wrong classification scores. Useful for scenarios of multiple classes.



Computing the mean across its diagonal, corresponds to the average classification accuracy.

Q&A

Thank you!

francisco.roman@itam.mx