Lesson 06

BoW, Optical Flow, Lucas-Kanade, Background Subtraction

Ing. Marek Hrúz, Ph.D.

Katedra Kybernetiky
Fakulta aplikovaných věd
Západočeská univerzita v Plzni

Bag of Words

Optical Flow

Lucas-Kanade approach

Background subtraction

DEPARTMENT OF
CYBERNETICS

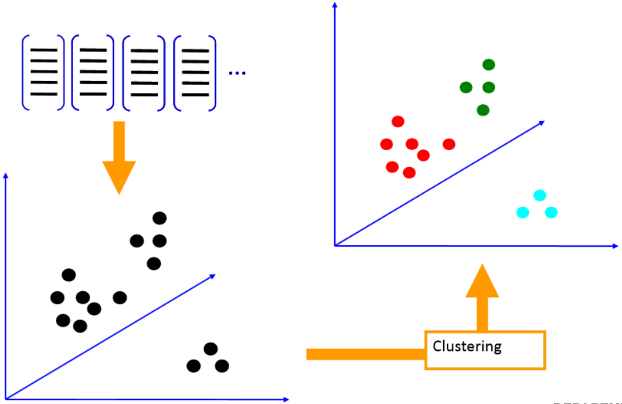
Bag of Words

- is an image classification method using ideas from document classification
- uses a sparse histogram of words from a vocabulary - local image features
- **WORD:**
- a word is a local feature
- the feature should be independent on scale, rotation, translation, intensity and contrast changes - SIFT
- the features will have some diversity - like normal words
- we want to obtain one representative form of the word

DEPARTMENT OF CYBERNETICS

Vocabulary

- all features are put together - we don't know what the individual words are yet
- the features are clustered using k-means
- 'k' will define the size of the vocabulary



DEPARTMENT OF CYBERNETICS

Image descriptor

- we now have the words and vocabulary defined
- SIFT analysis will provide us with features from the image
- each SIFT is classified as a word using nearest neighbor
- the image is represented as a histogram of these words

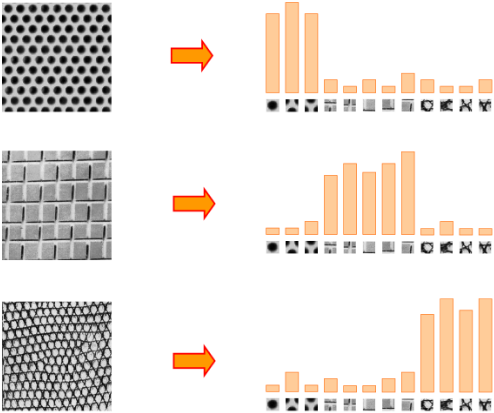


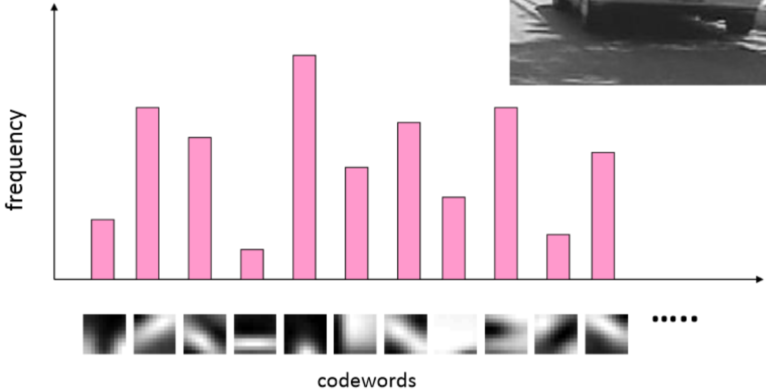
Image classification

- there are several options how to classify the unknown vector - Machine Learning - but not yet
- our vector is the histogram of the local features - words
- we have defined several measures for comparing histogram - LBP histogram comparison
- in BoW an angle between the histograms is used

$$\cos\alpha = \frac{H^1.H^2}{\|H^1\|\|H^2\|} \tag{1}$$

- the smaller the angle the more similar the histograms are

DEPARTMENT OF CYBERNETICS


frequency
codewords
.....

Optical Flow

- if a camera moves in 3D scene, the apparent motion in the sequence is called optical flow
- describes the direction and the speed of the motion of features in an image
- the computation is based on two assumptions:
  1. The observed brightness of any object point is constant over time.
  2. Neighboring pixels in the image plain move in a similar manner.

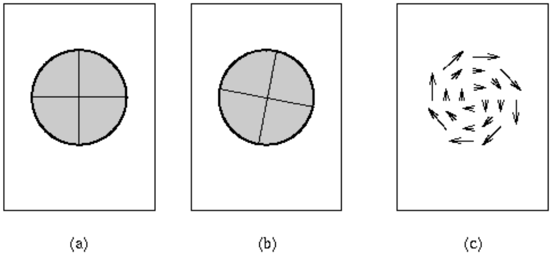


Figure 14.6 *Optical flow: (a) Time t_1, (b) time t_2, (c) optical flow.*

DEPARTMENT OF CYBERNETICS

- let $f(x, y, t)$ be a dynamic image function
- we want to observe the changes $(\delta x, \delta y)$ in consecutive frames meaning that $t = t + \delta t$
- for this reason we express the dynamic image function as a Taylor series

$$f(x+\delta x, y+\delta y, t+\delta t) = f(x, y, t)+\frac{\partial f}{\partial x}\delta x+\frac{\partial f}{\partial y}\delta y+\frac{\partial f}{\partial t}\delta t+O(\partial^2) \tag{2}$$

- since the pixel at (x, y, t) will move to $(x + \delta x, y + \delta y, t + \delta t)$ and we will assume the intensity of the pixel is constant, we can write

$$f(x + \delta x, y + \delta y, t + \delta t) = f(x, y, t) \tag{3}$$

- and we can substitute to the prior equation (with notation $\frac{\partial f}{\partial x} = f_x$)

$$-f_t = f_x \frac{\delta x}{\delta t} + f_y \frac{\delta y}{\delta t} \tag{4}$$

DEPARTMENT OF CYBERNETICS

- we have an image in time t and image in time $t + \delta t$
- the goal is to compute the velocity $c = (\frac{\delta x}{\delta t}, \frac{\delta y}{\delta t}) = (u, v)$ in every point of the image function
- recall: $-f_t = f_x \frac{\delta x}{\delta t} + f_y \frac{\delta y}{\delta t}$
- the partial derivatives of the image function can be approximated directly from the image itself
- the spatial derivatives f_x, f_y refer to changes in the brightness pattern, high values mean corners
- the time derivate f_t describes the change of brightness in time
- the equation has two unknown parameters and thus additional constrains need to be implemented

Lucas-Kanade approach

- this method sets the additional conditions to compute the optical flow
- the method assumes that pixels in local neighborhood move in similar matter
- thus the optical flow equations can be computed in a least squares fashion
- this means that the optical flow equation must hold for all pixels in a window centered at (x, y)
- in other words the flow vector (u, v) must satisfy:

$$\begin{aligned} f_x(q_1)u + f_y(q_1)v &= -f_t(q_1) \\ f_x(q_2)u + f_y(q_2)v &= -f_t(q_2) \\ &\vdots \\ f_x(q_n)u + f_y(q_n)v &= -f_t(q_n) \end{aligned} \tag{5}$$

- the equation can be written in matrix form $Av = b$

$$A = \begin{bmatrix} f_x(q_1) & f_y(q_1) \\ f_x(q_2) & f_y(q_2) \\ \vdots & \vdots \\ f_x(q_n) & f_y(q_n) \end{bmatrix} \quad v = \begin{bmatrix} u \\ v \end{bmatrix} \quad b = \begin{bmatrix} -f_t(q_1) \\ -f_t(q_2) \\ \vdots \\ -f_t(q_n) \end{bmatrix} \tag{6}$$

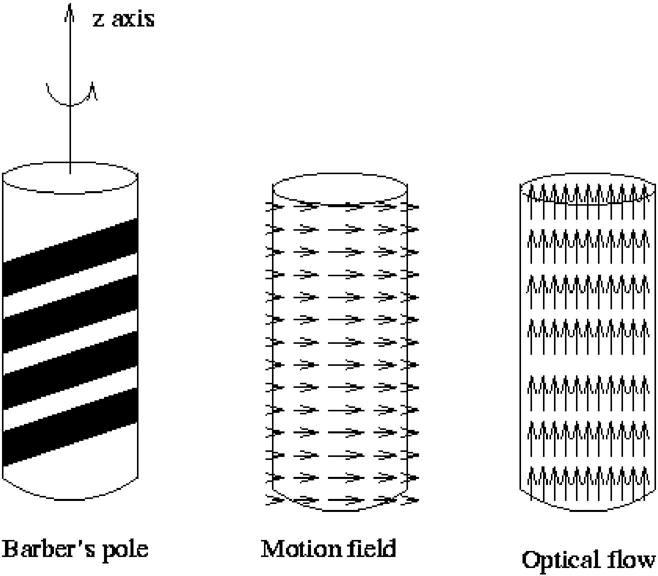
- the least squares method then states

$$\begin{aligned} A^T A v &= A^T b \\ v &= (A^T A)^{-1} A^T b \end{aligned} \tag{7}$$

- in this scenario all the pixels in the neighborhood have the same importance
- however in practice it is beneficial to weight the pixels so that the further they are from the center the less weight they have

$$A^T W A v = A^T W b \tag{8}$$

- W is usually set to be Gaussian


z axis
Barber's pole
Motion field
Optical flow

Lucas Kanade tracking applications

- https://www.youtube.com/watch?v=8VbyICRn3iI
- https://www.youtube.com/watch?v=oL67qe-Fhps

DEPARTMENT OF CYBERNETICS

Background subtraction

- is a method of computer vision which extracts the moving foreground from an image sequence
- requires static camera
- the method computes differences between an actual frame and a reference frame
- the image is divided into segments S - pixel, superpixel, region, ...
- each segment S_i is labeled either as 1 - movement occurs in the segment, or 0 - segment is background

$$F_i(t) = \begin{cases} 1 & \text{if } d(S_i(t), B) > \tau \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

Basic approach

- easiest way of obtaining the background model B is to model the image in which no movement is present
- such image will be composed only of segments that are zero: $\forall i : F_i = 0$
- the model can be adapted with time:

$$B_i(t+1) = (1-\alpha)B_i(t) + \alpha S_i(t) \tag{10}$$

$$\begin{aligned}
d_0 &= |S_i(t) - B_i(t)| \\
d_1 &= |S_i(t)^R - B_i(t)^R| + |S_i(t)^G - B_i(t)^G| + |S_i(t)^B - B_i(t)^B| \\
d_2 &= (S_i(t)^R - B_i(t)^R)^2 + (S_i(t)^G - B_i(t)^G)^2 + (S_i(t)^B - B_i(t)^B)^2 \\
d_\infty &= \max\{|S_i(t)^R - B_i(t)^R|,\ |S_i(t)^G - B_i(t)^G|,\ |S_i(t)^B - B_i(t)^B|\}
\end{aligned}$$

Modeling the background

- **Gaussian distribution**
- for the training we need samples - in time and in space
- this requires to obtain more images of the background in time
- each segment S_i has a color distribution in time and space

$$\begin{bmatrix} s_{i0}(t_0) & s_{i1}(t_0) & s_{i2}(t_0) & \dots & s_{iN}(t_0) \\ s_{i0}(t_1) & s_{i1}(t_1) & s_{i2}(t_1) & \dots & s_{iN}(t_1) \\ \vdots & & & & \\ s_{i0}(t_n) & s_{i1}(t_n) & s_{i2}(t_n) & \dots & s_{iN}(t_n) \end{bmatrix} \tag{11}$$

- note: if the segment consist only of one pixel, we have:

$$\begin{bmatrix} s_i(t_0) & s_i(t_1) & s_i(t_2) & \dots & s_i(t_n) \end{bmatrix}^T \tag{12}$$

DEPARTMENT OF CYBERNETICS

- nevertheless we have our training data as a sample of the background in given position
- from this sample we can approximate the distribution - in this case a Gaussian distribution - μ, Σ

$$\mu_i = \frac{1}{N} \sum_{j,t} s_{ij}(t) \tag{13}$$

$$\Sigma_i = \frac{1}{N-1} \sum_{j,t} (s_{ij}(t) - \mu_i)^T (s_{ij}(t) - \mu_i) \tag{14}$$

$$\eta\left(\mathbf{S}_i(t), \boldsymbol{\mu}_i, \Sigma_i\right) = \frac{1}{(2\pi)^{\frac{K}{2}} \left|\Sigma_i\right|^{\frac{1}{2}}} \cdot e^{-\frac{1}{2}(\mathbf{I}_{s,t} - \boldsymbol{\mu}_i)^\top \Sigma_i^{-1} (\mathbf{I}_{s,t} - \boldsymbol{\mu}_i)}, \tag{15}$$

$$\begin{aligned} \boldsymbol{\mu}_i(t+1) &= (1-\alpha)\,\boldsymbol{\mu}_i(t) + \alpha \mathbf{S}_i(t), \\ \Sigma_i(t+1) &= (1-\alpha)\,\Sigma_i(t) + \alpha\left(\mathbf{S}_i(t) - \boldsymbol{\mu}_i(t)\right)^\top \left(\mathbf{S}_i(t) - \boldsymbol{\mu}_i(t)\right) \end{aligned}$$

- the distance is computed as Mahalanobis distance

$$d\left(\mathbf{I}_{s,t}, \boldsymbol{\mu}_{s,t}\right) = \sqrt{\left(\mathbf{I}_{s,t} - \boldsymbol{\mu}_{s,t}\right)^\top \Sigma_{s,t}^{-1} \left(\mathbf{I}_{s,t} - \boldsymbol{\mu}_{s,t}\right)}, \tag{16}$$

DEPARTMENT OF CYBERNETICS

Gaussian mixture model

- sometimes the background can be dynamic
- the color of pixels can change in time, but still is considered as background (running water, moving trees, ...)
- we will model the multimodal density as GMM

$$P\left(S_i(t)\right) = \sum_{k=1}^{K} \omega_{k,i,t} \cdot \eta\left(\mathbf{S}_i(t), \boldsymbol{\mu}_i(t), \Sigma_i(t)\right), \qquad (17)$$

$$\begin{aligned}
\omega_{k,s,t+1} &= (1-\alpha)\omega_{k,s,t} + \alpha, \\
\boldsymbol{\mu}_i(t+1) &= (1-\alpha)\boldsymbol{\mu}_{i,s,t} + \alpha\mathbf{S}_i(t), \\
\Sigma_i(t+1) &= (1-\alpha)\Sigma_i(t) + \alpha\left(\mathbf{S}_i(t) - \boldsymbol{\mu}_i(t)\right)^\top \left(\mathbf{S}_i(t) - \boldsymbol{\mu}_i(t)\right),
\end{aligned}$$

- in this case we cannot compute distance, but must compute probability

DEPARTMENT OF CYBERNETICS

Modeling using Histogram

- in this case the segment must consist of several pixels
- from these pixels a histogram is computed
- the distance of histograms of segments in consecutive frames is computed via Pearson's correlation

$$d\left(H_1, H_2\right) = 1 - r_{H_1,H_2}, \tag{18}$$

$$r_{H_1,H_2} = \frac{\sum_{i=1}^{N}\left(H_1^i - \bar{H}_1\right)\left(H_2^i - \bar{H}_2\right)}{\sqrt{\sum_{i=1}^{N}\left(H_1^i - \bar{H}_1\right)^2 \cdot \sum_{i=1}^{N}\left(H_2^i - \bar{H}_2\right)^2}} \tag{19}$$

- we can use other metrics: recall Local Binary Patterns histogram comparison

DEPARTMENT OF CYBERNETICS

Usage

- https://www.youtube.com/watch?v=QSfcrbtOaQw

DEPARTMENT OF CYBERNETICS