

SIFT – Scale Invariant Feature Transform

scale space pyramid propagated by (Lowe 2004) that also incorporates gradient orientation

- **goal:** extraction of stable features, i.e. keypoints, that are detected in slightly different images (affine transform of view, scale, noise level, partial occlusion, illumination change,...) too.
- SIFT **WAS** protected by US patent → license of University of British Columbia required for commercial use
- a SIFT keypoint is described by position (x,y), scale, local orientation and its direct neighborhood characterized in a 128 value array, denoted as SIFT descriptor.
- conceptual overview at high level:
 - **step1 scale space extrema detection:** keypoint detection on DoG scale space, invariant to scale and orientation
 - **step2 keypoint localization:** for each keypoint candidate, scale, location and stability is determined
 - **step3 orientation assignment:** for each keypoint candidate, the orientation is approximated from local image gradients
 - **step4 keypoint description:** build up a keypoint descriptor based on local gradient distribution

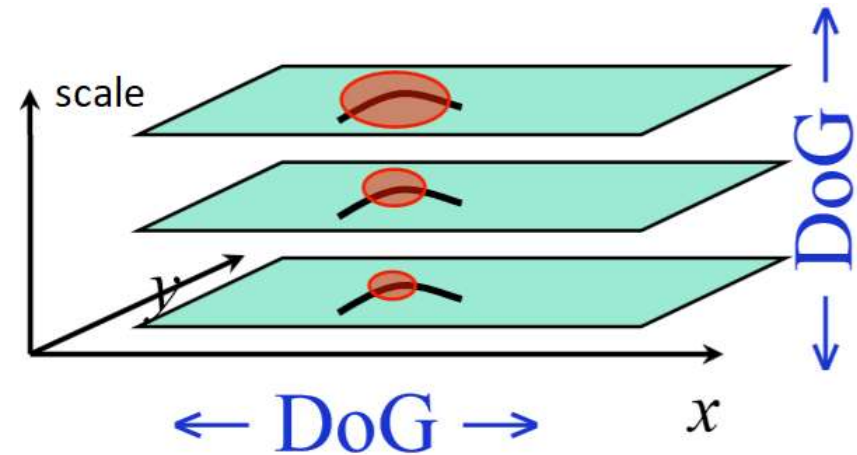
SIFT – STEP1 scale space extrema detection

- application of a DoG pyramid

- calculation of Gauss filtered images, so called scales, that are grouped in octaves of same image size.
- each octave at same scale calculated with $k = 2^{\frac{1}{2}} = \sqrt{2}$. Thus the 4 images of octave #1 in $[\sigma; 2\sigma]$, for octave #2 in $[2\sigma; 4\sigma]$, for octave #3 in $[4\sigma; 8\sigma]$ aso.
- thus octave 1 starting with scale σ , octave 2 with scale 2σ ,...
- a keypoint candidate is a hotspot value in an scale image of an octave showing extremum (minimum or maximum) value in local N_{26} thus incorporating images at three different scales.

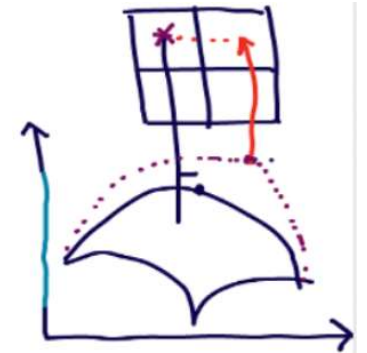


832 DoG extrema



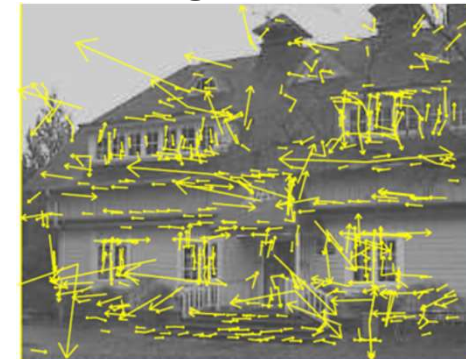
SIFT – STEP2 keypoint localization

- not all of the DoG extrema are useful keypoints, only the stable ones desired
 - remove DoG extrema in low contrast image regions that are very sensitive to noise by assessing the local neighborhood utilizing a quadratic function within each octave:
 - calculate true maximum/minimum position in local surface-shaped neighborhood utilizing Taylor series expansion and interpolation
 - thereby calculation of derivatives for (x, y, σ) leads to 3x3 Hessian matrix
 - reject flats with value $|I(\hat{p})| < 0.03$, point p in scale space as $p = (x, y, \sigma)$



[from <https://devanginiblog.wordpress.com/2016/05/10/sift-scale-invariant-feature-transform/>]

- remove DoG extrema that are placed on edges by assessing the “corneriness” of each keypoint:
 - utilizing Hessian / Harris corner detector. Around corners there is no dominant principal curvature from eigenvalues α and β sorted by magnitude
 - $C = \begin{bmatrix} I_{xx} & I_{xy} \\ I_{xy} & I_{yy} \end{bmatrix}$, $trace(C) = I_{xx} + I_{yy} = \alpha + \beta$, $det(C) = I_{xx} \cdot I_{yy} - I_{xy}^2 = \alpha \cdot \beta$
 - let $r = \frac{\alpha}{\beta}$, thus $\frac{(\alpha+\beta)^2}{\alpha \cdot \beta} = \frac{(r \cdot \beta + \beta)^2}{r \cdot \beta^2} = \frac{(r+1)^2}{r}$
 - with Harris criterion $\frac{(trace(C))^2}{det(C)} < \frac{(r+1)^2}{r}$, removing keypoints showing r below threshold $T=10$.



729 after contrast threshold,
536 after corneriness

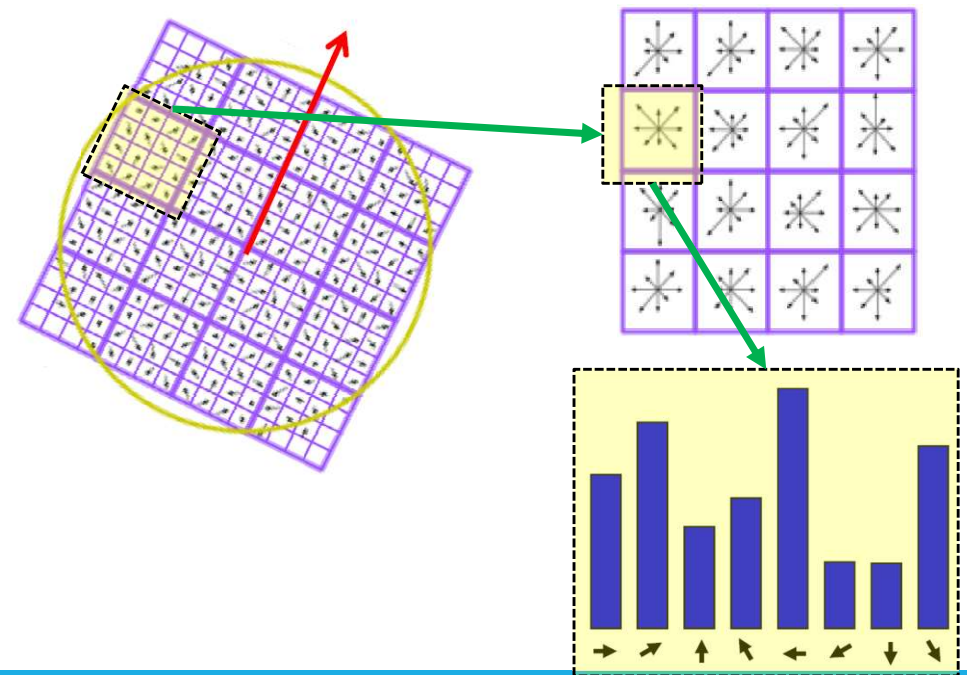
SIFT – STEP3 orientation assignment

- the novelty of SIFT is its robustness in case of rotation, as for each keypoint the local orientation is assessed.
 - aim: assign constant orientation to each keypoint. If more orientations present in local neighbourhood, then split the keypoint → rotational invariance is achieved
 - for each keypoint, gradient magnitude and gradient direction are calculated for all considered scales.
 - the directions are stored in a histogram with 36 bins, covering full 360° at 10° granularity.
 - the orientation are thereby weighted according to a Gauss function and primarily their magnitude
 - the orientation histogram peak/maximum determines the keypoint orientation.
 - if there are other orientations within quantile $q_{80}(histo)$, then the keypoint is copied with same position, same scale but divergent orientation.
 - gradient magnitude and orientation calculated by $G = \sqrt{G_x^2 + G_y^2}$ and $\theta = \arctan(G_y/G_x)$ respectively, with $G_x \approx I(x+1, y) - I(x-1, y)$



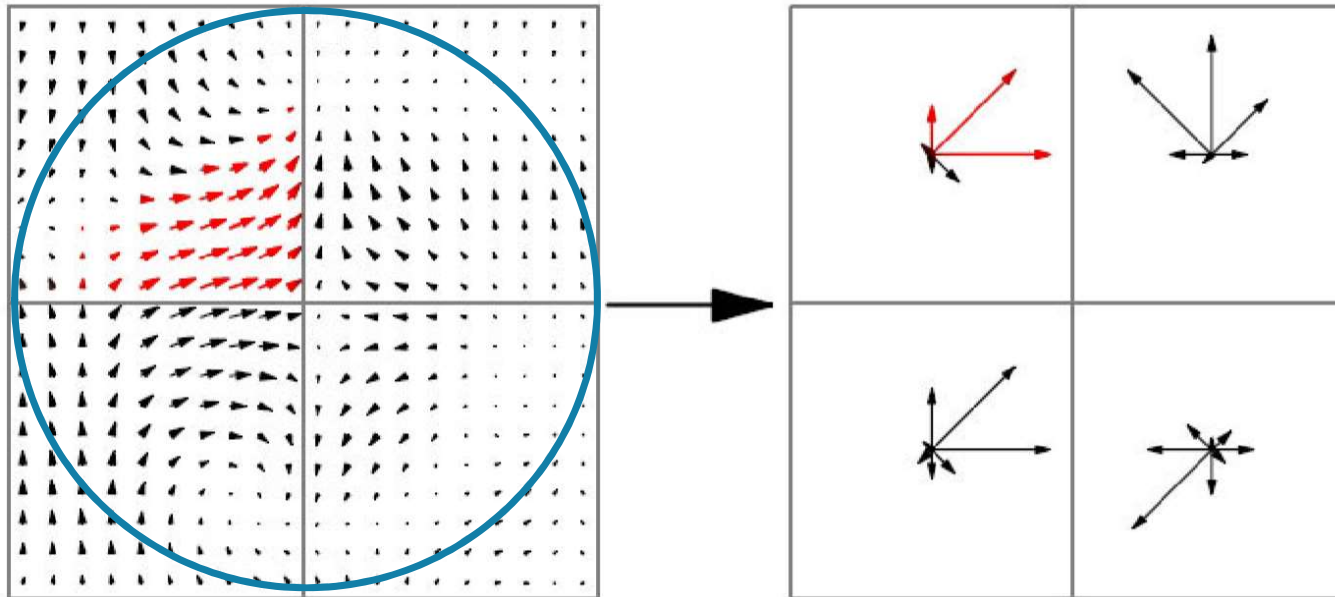
SIFT – STEP4 keypoint description


- the first three steps lead to detected keypoints that are robust with respect to affine transformations (position, rotation and scale) and are already assigned an orientation
- to further achieve robustness towards local illumination and changes in the viewing direction, the local region characteristics need to be incorporated too, introducing the *keypoint descriptor*:
 - in local 16x16 neighborhood, orientation and gradient magnitude is calculated for all discrete positions.
 - these 256 values are aggregated in 16 regions of size 4x4.
 - For each of the 16 regions a histogram is calculated
 - the histograms show granularity of 8 bins (classes)
 - orientation values are thereby weighted by Gaussian and gradient magnitude too
 - the keypoint descriptor finally comprises 16 histograms with 8 bins each as a 128-value vector.
 - to achieve robustness towards illumination and changing contrast, the vector is finally normalized (unit vector)



SIFT – STEP4 keypoint description cont'd

for better illustration, the following figures show the feature descriptor calculated at 2x2 segments only:



- with 4x4 descriptors over 16x16 samples we get a feature descriptor vector with 128 values (4x4x8)
- finally the 128-element vector is normalized 
- array might be truncated to [0.0;0.2] and re-normalized to handle non-linear change in illumination and reducing dominant influence of the larger gradients

SIFT-Feature Trajectory

- for object detection and object tracking it is essential to match the keypoints and their feature descriptors available from the two images A (reference) and image B (test).
- SIFT utilizes the nearest neighbor algorithm [Lowe 2001] to determine matching keypoints according to Euclidean feature distance between the feature descriptor vectors as $\sqrt{\sum_{i=1}^N (V_{1i} - V_{2i})^2}$, $N = 128$
- confidently matching SIFT feature from both images are thereby represented as feature trajectory
- ratio test: to introduce a higher level of noise invariance and prevent from false matches, trajectories and keypoints are rejected respectively, if the second best matching neighbor is within 0.8 of the best match distance.
- object features still might get misclassified. Thus, features of separated objects are interpreted as cluster, that need trajectory representatives of similar cluster shape. Therefore, Hough transform besides common registration is applicable as search strategy.
 - the keypoints thereby vote for different object poses, i.e. affine transformation parameters
 - the object pose with most votes leading to clusters in the Hough space represents the most probable result.
 - due to incorporation of the features as cluster, a significantly higher level of correlation is achievable compared to single keypoint matching

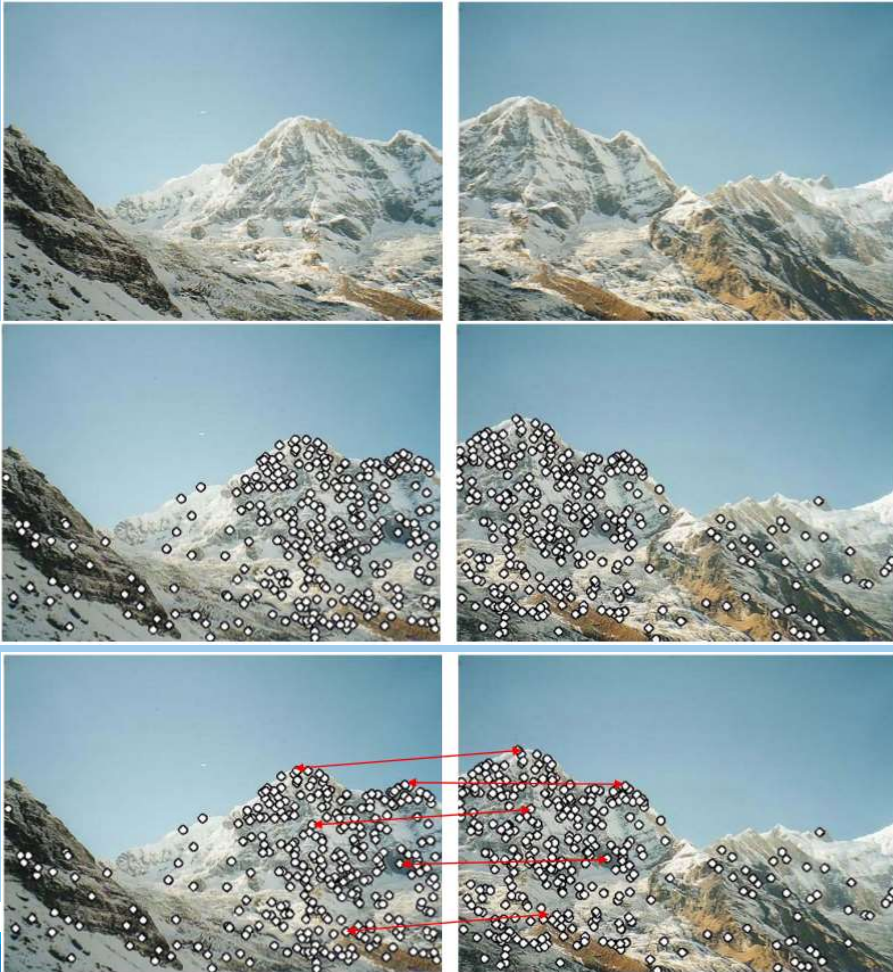
SIFT – Results

- local stable oriented features that are invariant to noise, position, affine transformations and scale.
- scale (from detection) is indicated by circle size

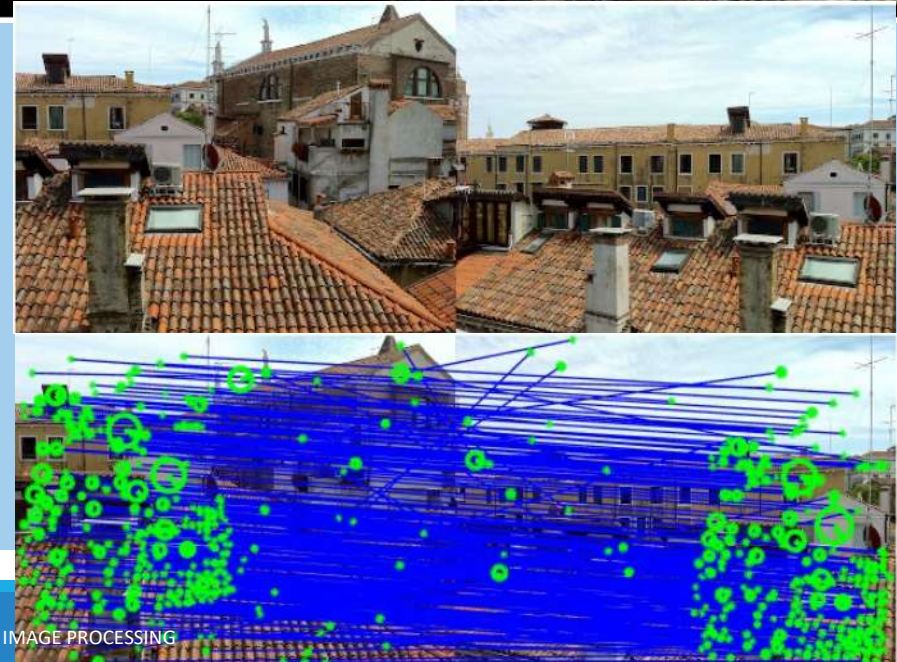


SIFT – Field of Application

■ Image stitching



5/16/2021

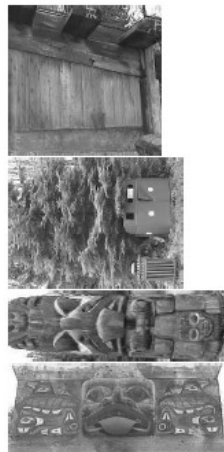


BVA2- ADVANCED IMAGE PROCESSING

49

SIFT – Field of Application con'd

- matching objects via feature trajectory
- works even in case of affine transformation



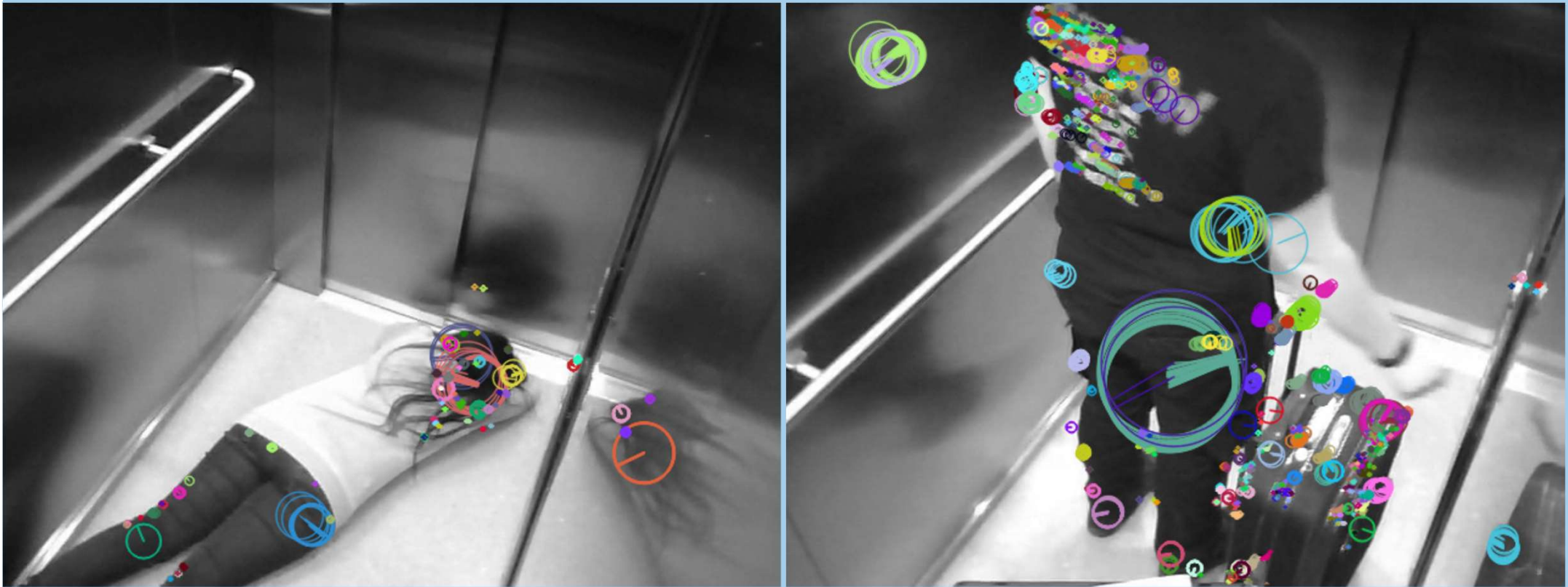
SIFT – Field of Application con'd

- matching objects via feature trajectory
- works even in case of partial occlusion



SIFT – Field of Application con'd

- SIFT Keypoint trajectory to track moving objects in elevators (Winkler 2016) for emergency detection



Histogram of Oriented Gradients (HOG)

Overview

- first introduced in the paper “*Histogram of Oriented Gradients for Human Detection*” of [Dalal and Triggs 2005].
- fields of application:
 - pedestrian detection as initial classification domain
 - general object recognition
 - face recognition
- feature vector calculated similar to the keypoint descriptor of SIFT, but evaluated for all regions in 8×8 segments
- HoG feature vectors perfectly applicable to support vector machine (SVM) classification in the field of image recognition
- applicable to RGB input images (3 channels)
- feature vector as code-transformation, i.e. from $width \cdot height \cdot 3$ for RGB images with 3 channels to local 9-bin histograms per 8×8 pixel cells comprising gradient orientation/magnitude information.
- pre-processing required, mainly to scale to normalized image size

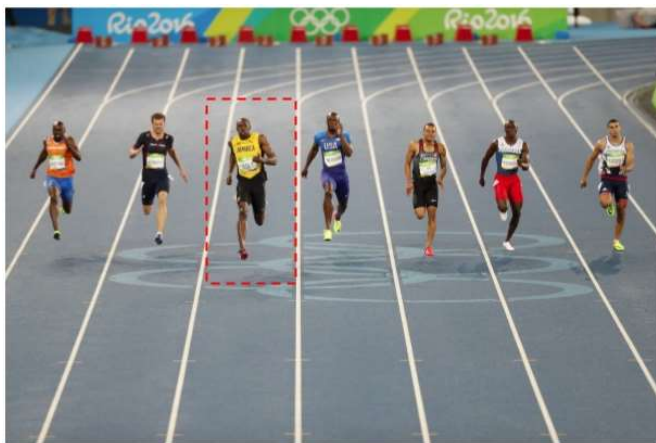
Histogram of Oriented Gradients (HOG) cont'd

■ (1)Pre-Processing

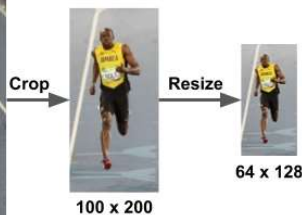
- patch-size needs to be pre-defined, e.g. 64×128 leading to an aspect ratio of $r = \frac{1}{2}$. input image needs to be cropped to pre-defined aspect ratio and then scaled to match the target patch size utilizing proper interpolation concepts such as *Lanczos* or *Cubic* interpolation.
- optionally gamma correction to be applied

■ (2) Gradient Image Calculation

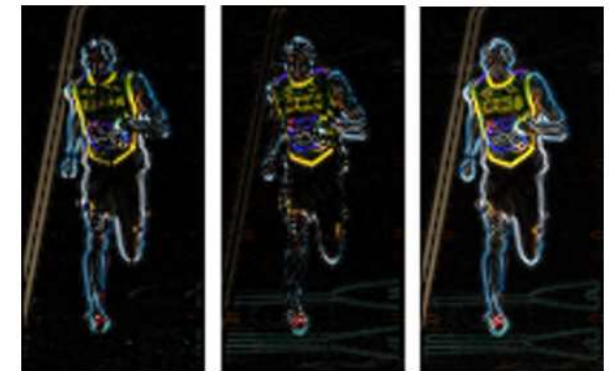
- calculation of horizontal and vertical gradients utilizing Sobel edge detection.
- calculation of gradient magnitude and gradient orientation by converting from cartesian g_x, g_y to polar representation
- in case of color images, the max magnitude over all channels with associated direction is utilized



Original Image : 720 x 475



[from <https://www.learnopencv.com/histogram-of-oriented-gradients/>]

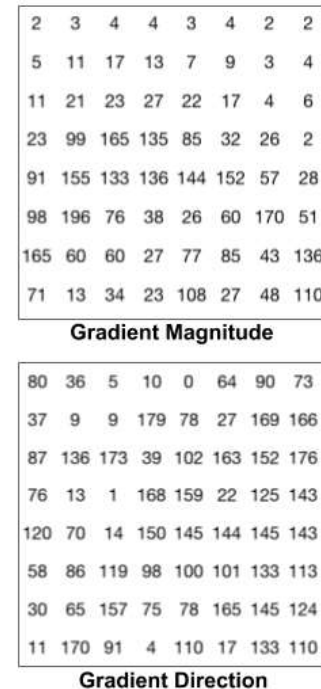
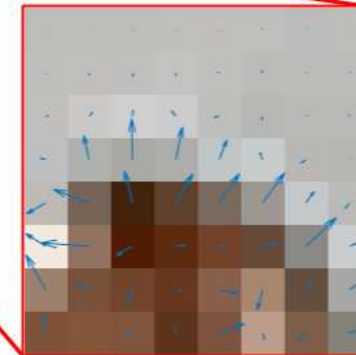
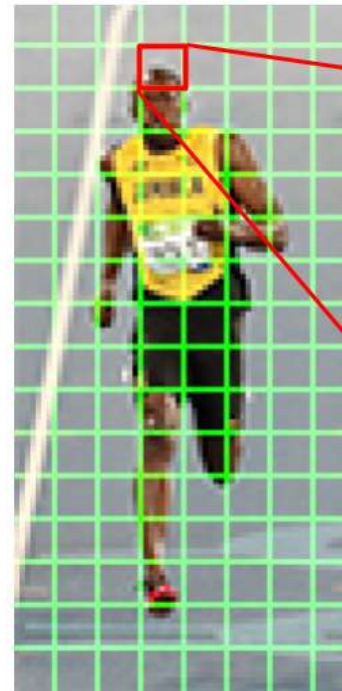
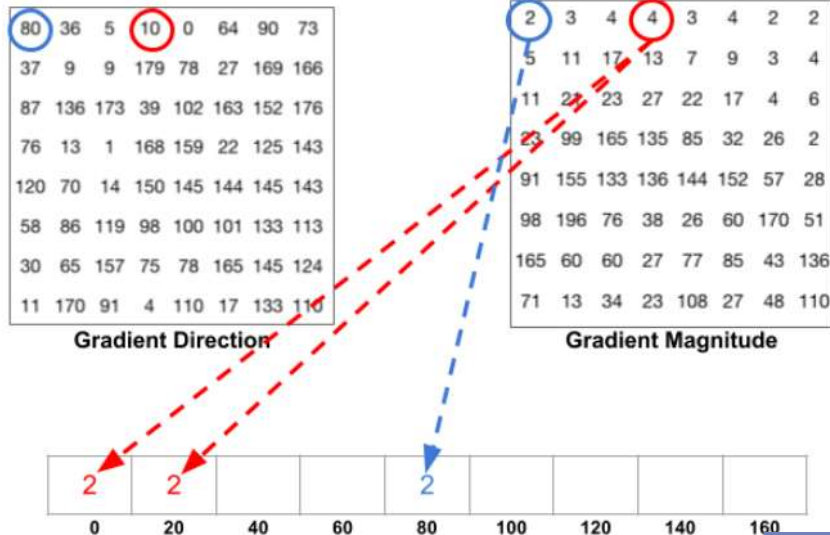


$$g_x, g_y \text{ and magnitude } m = \sqrt{g_x^2 + g_y^2}$$

Histogram of Oriented Gradients (HOG) cont'd

■ (3) Calculation of the gradient histogram

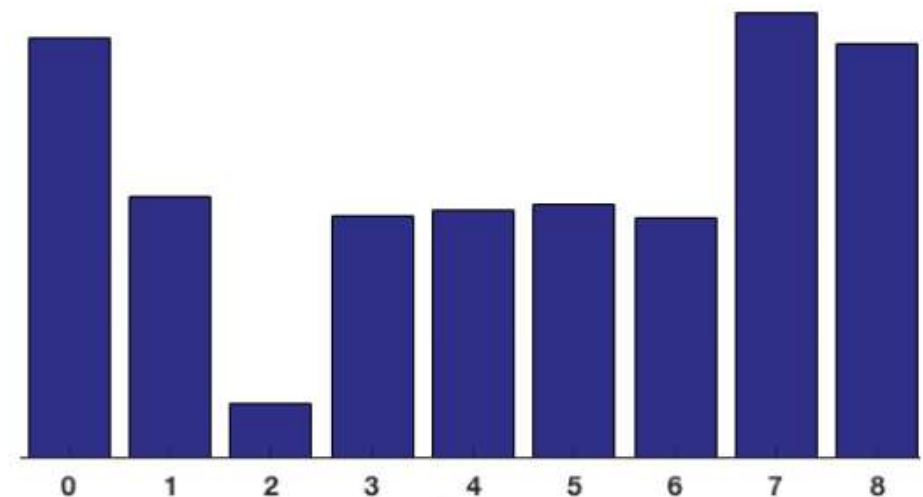
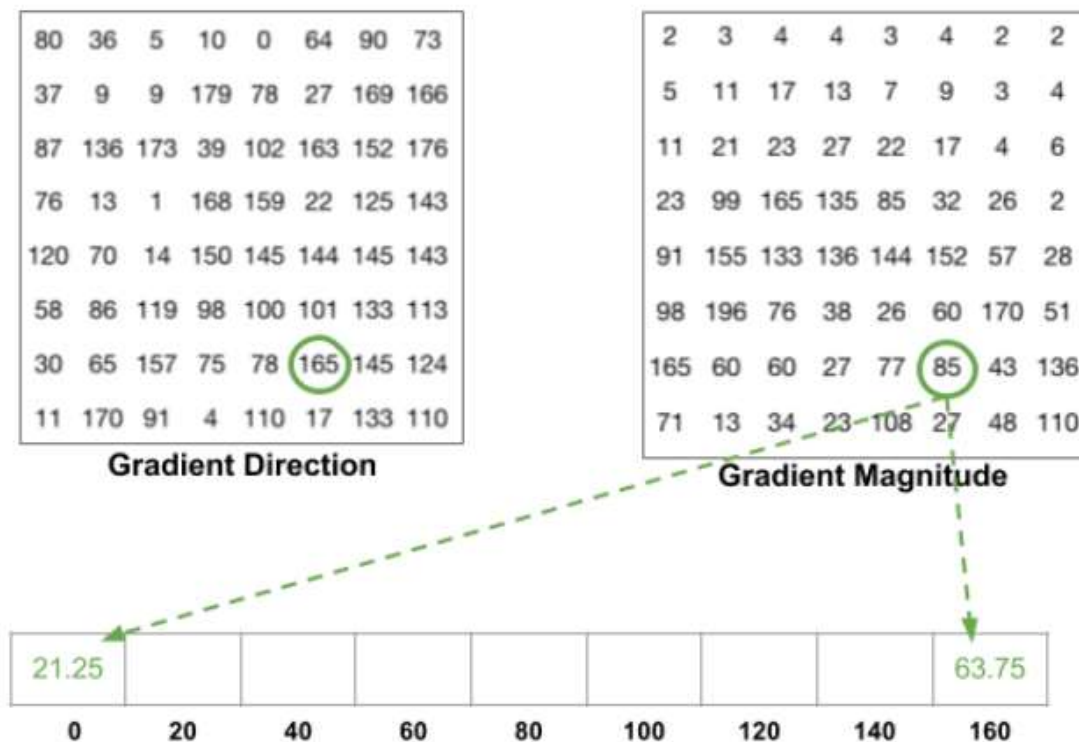
- HoG calculated for 8×8 pixel cells, c.f. DCT for JPEG compression
- 8×8 pixel cells show higher noise robustness compared to calculating gradient metrics per pixel while still preserving fine image details w.r.t. chosen constant scale.
- HoG histogram with $n = 9$ bins, representing gradient orientations of $\theta = (n - 1) \cdot 20$, thus covering angles of $0, 20, \dots, 160$ degrees.
 - thus, unsigned gradients are used with $0 \equiv 180$
 - interpolation utilized for intermediate angles



Histogram of Oriented Gradients (HOG) cont'd

■ (3) Calculation of the gradient histogram cont'd

- calculation of bins:
 - how to interpolate if angle $160 < \theta < 180$? Then the values are assigned the bins 0 and 160!
 - finally, per 8×8 pixel cell a filled histogram is given



[from <https://www.learnopencv.com/histogram-of-oriented-gradients/>]

Histogram of Oriented Gradients (HOG) cont'd

■ (4) Block Normalization

- gradient magnitude is sensitive to local lighting in the image, thus also affect histogram characteristics
- normalization is required to balance the local lighting variations
- to better incorporate local neighborhood, four *overlapping* blocks within 16×16 are normalized together, constructing a 36 component vector (from four times 9 bins) $\vec{v} = (b_{1,1}, b_{1,2}, \dots, b_{1,9}, b_{2,1}, \dots, b_{4,9})$

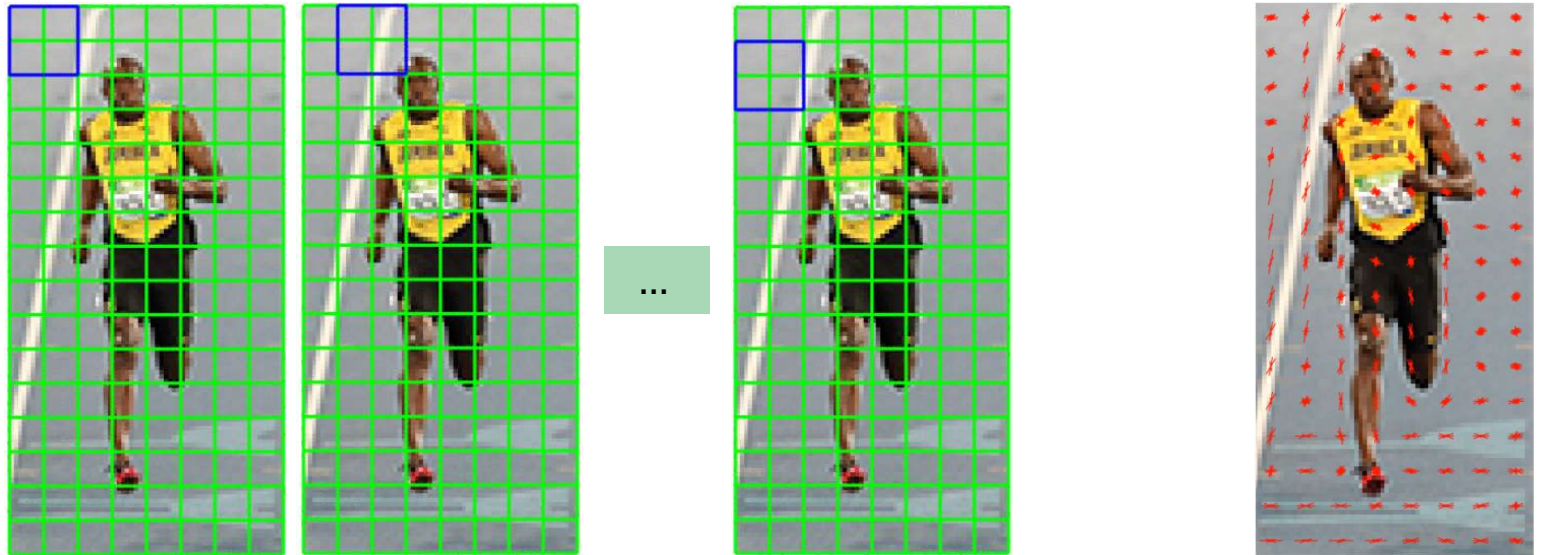
– the 36 component vector is then per component normalized by L2 norm, i.e. length of the vector as $|\vec{v}| =$

$$\sqrt{b_{1,1}^2 + b_{1,2}^2 + \dots + b_{1,9}^2 + b_{2,1}^2 + \dots + b_{4,9}^2} \text{ thus leading to normalized } n(\vec{v}) = \left(\frac{b_{1,1}}{|\vec{v}|}, \frac{b_{1,2}}{|\vec{v}|}, \dots, \frac{b_{1,9}}{|\vec{v}|}, \frac{b_{2,1}}{|\vec{v}|}, \dots, \frac{b_{4,9}}{|\vec{v}|} \right)$$

– thus, $n \times m$ vectors of size 9 are replaced by $(n - 1) \times (m - 1)$ vectors of size 36.

– entire HoG feature vector as concatenation of the 36-element vectors

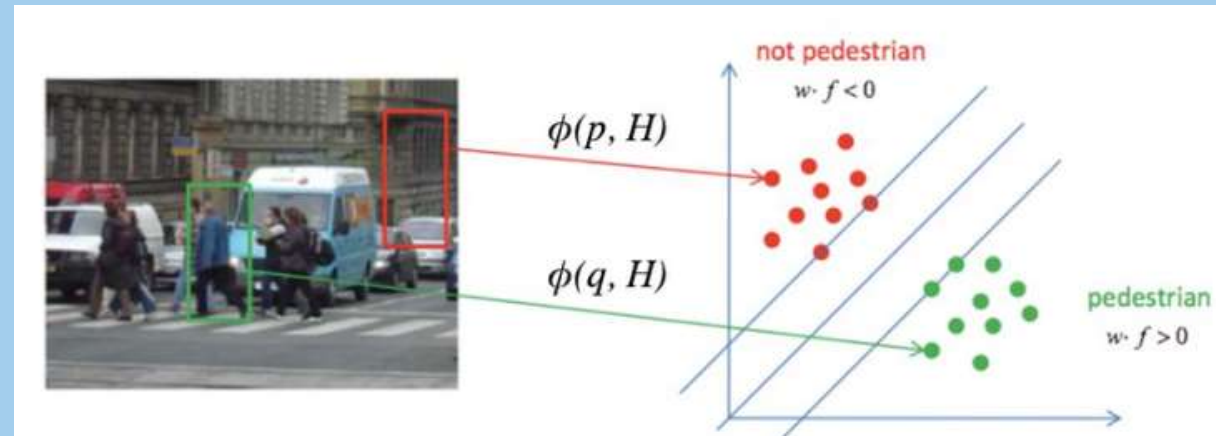
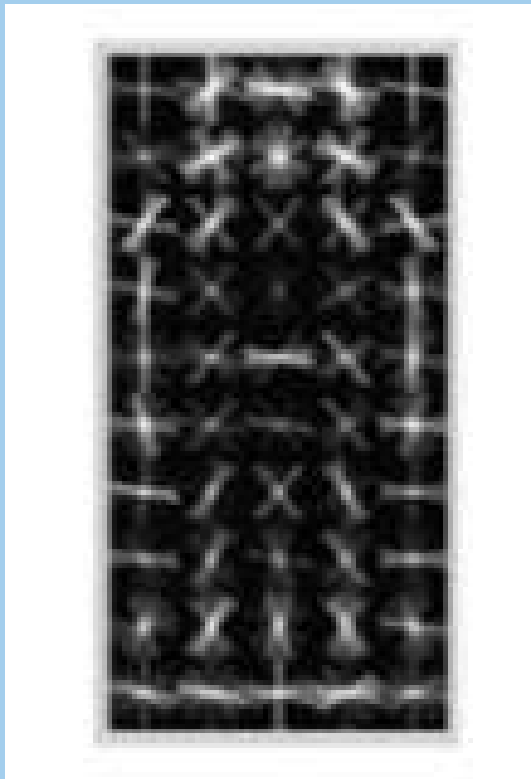
– for visualization purpose, per 8×8 patch only the normalized 9-bin-vector is utilized to prevent from redundant positions



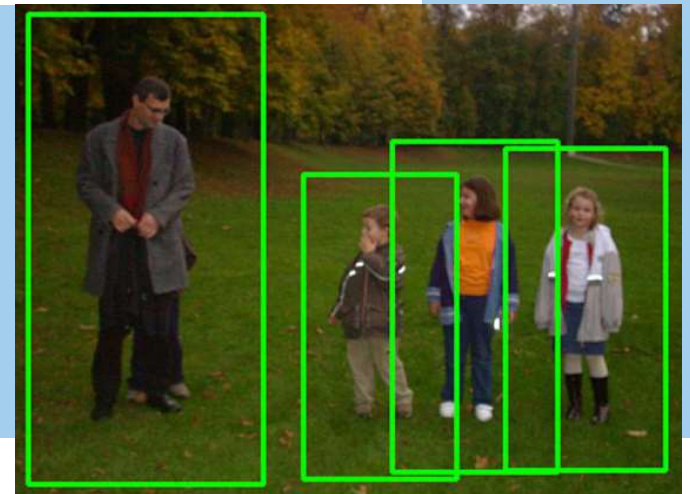
[from <https://www.learnopencv.com/histogram-of-oriented-gradients/>]

Histogram of Oriented Gradients (HOG) – Field of Application

- pedestrian detection in the work of [Dalal and Triggs 2005] discriminating between pedestrians and non-pedestrians by utilizing SVM-classifier

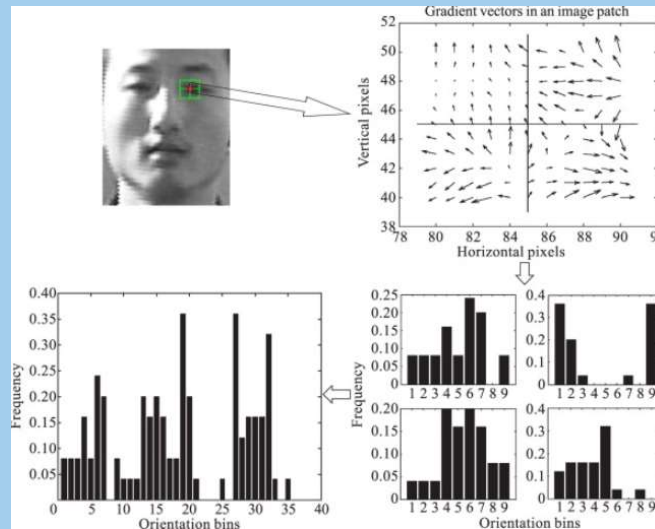


pedestrian HOG model robust w.r.t.
scale and partial occlusions

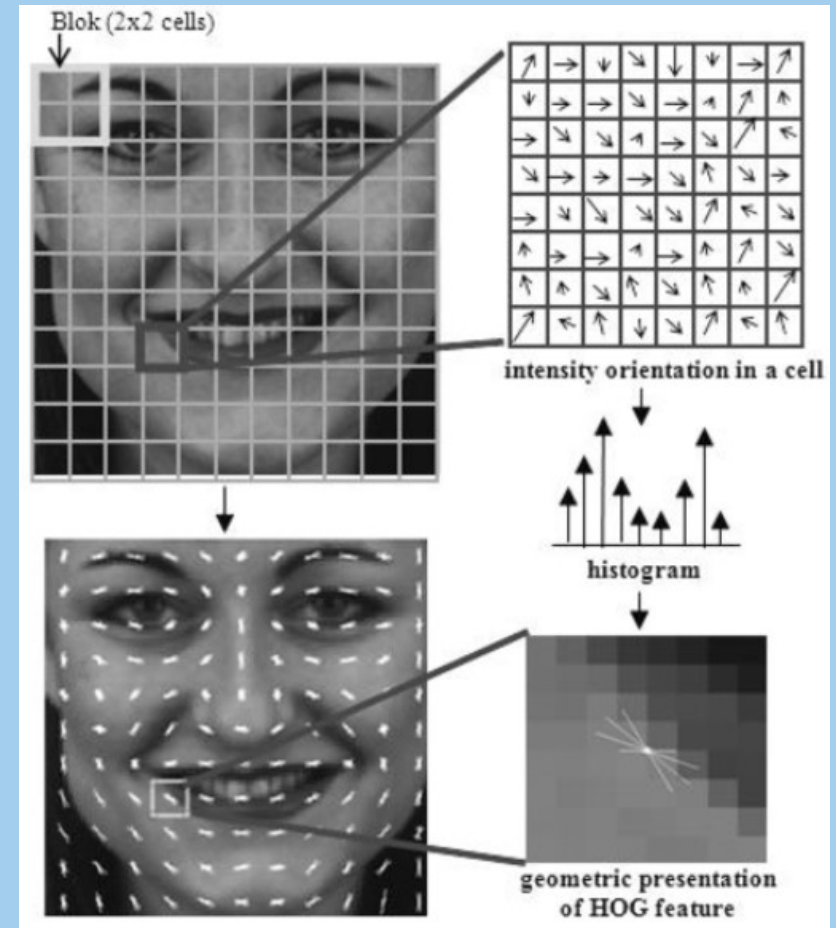


Histogram of Oriented Gradients (HOG) – Field of Application

■ face recognition



Automatic-system-for-facial-expression-
recognition-Greche-Es-Sbai



References

- [Beaudet 1978] P.R. Beaudet, 1978. “*Rotationally invariant image operators*”. In: International Joint Conference on Pattern Recognition, pp. 579-583.
- [Dalal and Triggs 2005] N. Dalal and B. Triggs, 2005. “Histogram of Oriented Gradients for Human Detection”. In: International Conference on Computer Vision & Pattern Recognition (CVPR '05), San Diego, USA.
- [Fries 2010] Carsten Fries, 2010. “Objekterkennung mit SIFT-Merkmalen”, Hochschule für Angewandte Wissenschaften Hamburg.
- [Harris and Stevens 1988] C. Harris and M. Stephens, 1988. “*A Combined Corner and Edge Detector*.” In: Proceedings of the 4th Alvey Vision Conference: pages 147--151.
- [Lindeberg 1998] Tony Lindeberg, 1998. Principles for Automatic Scale Selection. In: Jähne et al., eds. “Handbook on Computer Vision and Applications”, Academic Press.
- [Lowe 2001] David G. Lowe, 2001. “*Local Feature View Clustering for 3D Object Recognition*”. In: Proc. of the 2001 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, pp. 682—688.
- [Lowe 2004] David G. Lowe, 2004. „Distinctive Image Features from Scale-Invariant Keypoints“. Int. J. of Computer Vision 60(2), pp. 91–110, available from <http://www.cs.ubc.ca/~lowe/papers/cvpr01.pdf>
- [Winkler 2016] Sabine Winkler, 2016. „*Visuelle Notfalldetektion in Aufzugsanlagen durch Anwendung von Hintergrundsubtraktion und SIFT Feature Detection*“, Master Thesis, University of Applied Sciences Upper Austria. School of Informatics, Communications and Media, Campus Hagenberg.

References

- [Geiger et al. 2012] Geiger, A., Lenz, P., and Urtasun, R., 2012. *Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite*. In: Proc. of the Conference on Computer Vision and Pattern Recognition (CVPR).