DRIVES

Development and Research on Innovative Vocational Education Skills

COMPUTER VISION FUNDAMENTALS

U2.E9. FEATURE DETECTION AND MATCHING

Computer Vision Expert

May 2021, Version 1

Co-funded by the Erasmus+ Programme of the European Union

The Development and Research on Innovative Vocational Educational Skills project (DRIVES) is co-funded by the Erasmus+ Programme of the European Union under the agreement 591988-EPP-1-2017-1-CZ-EPPKA2-SSA-B. The European Commission support for the production of this publication does not constitute endorsement of the contents which reflects the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

LEARNING OBJECTIVES

The student is able to …

CVE.U2.E9.PC1	The student is able to define a feature.
CVE.U2.E9.PC2	The student knows the different types of image features.
CVE.U2.E9.PC3	The student is able to define feature detection and matching and understand their purposes.
CVE.U2.E9.PC4	The student knows some commonly used feature detectors and their classification.

FEATURE

A **feature** can be definied as a piece of information about the content of an image, usually about whether a certain region of the image has certain properties. This can be specific structures in the image such as points, edges or objects.

Image registration need to get correspondence between images.

Basic idea:

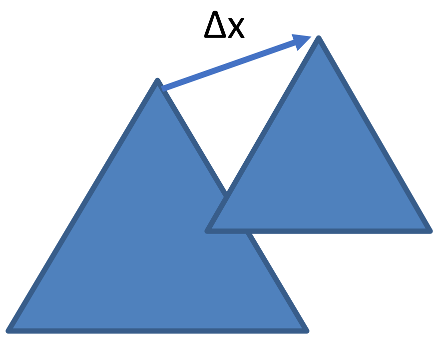
- detect feature points, called keypoints
- match feature points in different images

Want feature points to be detected consistently and matched correctly.

DIFFERENT TYPES OF IMAGE FEATURES

<u>Features available:</u>

- Harris corner
- Tomasi's "good features to track"
- SIFT: Scale Invariant Feature Transform
- SURF: Speeded Up Robust Feature
- GLOH: Gradient Location and Orientation Histogram
- etc.



A shifted corner make some difference in the image.

A shifted uniform region make no difference.

So, look for large difference in shifted image.

Suppose an image patch W at ***x*** is shifted by a small amount Δx. The sum-squared difference at ***x*** is:

$$E(\mathbf{x}) = \sum_{\mathbf{x}_i \in W} [I(\mathbf{x}_i) - I(\mathbf{x}_i + \Delta\mathbf{x})]^2 \quad (1)$$

That is,

$$E(x, y) = \sum_{(x_i, y_i) \in W} [I(x_i, y_i) - I(x_i + \Delta x, y_i + \Delta y)]^2 \quad (2)$$

This is called the **auto-correlation function.**

Apply Taylor's series expansion to $I(x_i + \Delta x)$:

$$\begin{aligned} I(x_i + \Delta x, y_i + \Delta y) &= I(x_i, y_i) + \frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y && (3) \\ &= I(x_i, y_i) + I_x \Delta x + I_y \Delta y && (4) \\ &= I(\mathbf{x}_i) + (\nabla I)^\top \Delta\mathbf{x} && (5) \end{aligned}$$

Where $\nabla I = (I_x, I_y)^T$

Replacing Eq. 5 into Eq. 1, the output is

$$E(\mathbf{x}) = \sum_W [I_x \Delta x + I_y \Delta y]^2 \quad (6)$$

$$= \sum_W [I_x^2 \Delta^2 x + 2 I_x I_y \Delta x \Delta y + I_y^2 \Delta^2 y] \quad (7)$$

$$= (\Delta \mathbf{x})^\top \mathbf{A}(\mathbf{x}) \Delta \mathbf{x} \quad (8)$$

Where the auto-correlation matrix **A** is given by:

$$\mathbf{A} = \begin{bmatrix} \sum_W I_x^2 & \sum_W I_x I_y \\ \sum_W I_x I_y & \sum_W I_y^2 \end{bmatrix} \quad (9)$$

A captures intensity pattern in W.

Response R(*x*) of Harris corner detector is given by:

$$R(\mathbf{x}) = \det \mathbf{A} - \alpha(\operatorname{tr} \mathbf{A})^2 \qquad \textbf{(10)}$$

Two manners to define corners:

(1) Large response The locations *x* with R(x) greater than certain threshold.

(2) Local maximum The locations *x* where R(x) are greater than those of their neighbors, i.e., apply non-maximum suppression.

Sample result (large response):

Many corners are detected.

Shi and Tomasi considered weighted auto-correlation:

$$E(\mathbf{x}) = \sum_{\mathbf{x}_i \in W} w(\mathbf{x}_i)\left[I(\mathbf{x}_i) - I(\mathbf{x}_i + \Delta\mathbf{x})\right]^2 \qquad \textbf{(11)}$$

Where $w(x_i)$ is the weight.

Then, **A** becomes

$$\mathbf{A} = \begin{bmatrix} \sum_W w I_x^2 & \sum_W w I_x I_y \\ \sum_W w I_x I_y & \sum_W w I_y^2 \end{bmatrix} \qquad \textbf{(12)}$$

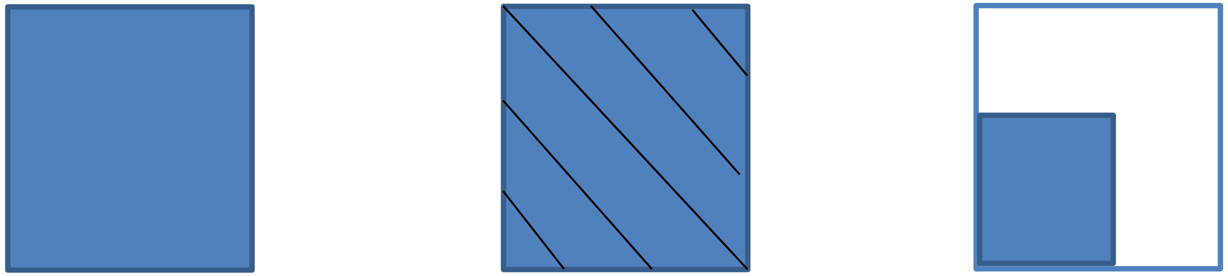
A is a 2×2 matrix. This means there exist scalar values λ_1, λ_2 and vectors v_1, v_2 like that

$$\mathbf{A}\,\mathbf{v}_i = \lambda_i \mathbf{v}_i \;, \quad i = 1, 2 \tag{13}$$

- v_i are the orthonormal eigenvectors,

$$\mathbf{v}_i^\top \mathbf{v}_j = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \tag{14}$$

λ_i are the eigenvalues; expect $\lambda_i \geq 0$.



1. If both λ_i are small, then feature does not vary much in any direction. ⇒ uniform region (bad feature).
2. If the larger eigenvalue $\lambda_1 \gg \lambda_2$, then the feature varies mainly in the direction of v_1. ⇒ edge (bad feature).
3. If both eigenvalues are large, then the feature varies significantly in both directions. ⇒ corner or corner-like (good feature).
4. In practice, *I* has a maximum value (e.g., 255). So, λ_1, λ_2 also have an upper bound. Then, only have to check that min(λ_1, λ_2) is large enough.

TOMASI'S GOOD FEATURE

Sample results (large maximum):

Detected corners are more spread out with non-maximum suppression.

- Tomasi's good feature uses smallest eigenvalue $\min(\lambda_1, \lambda_2)$.
- Harris corner uses $\det \mathbf{A} - \alpha(\mathrm{tr}\,\mathbf{A})^2 = \lambda 1 \lambda 2 - \alpha(\lambda_1 + \lambda_2)^2$.
- Brown et al. use the harmonic mean

$$\frac{\det \mathbf{A}}{\mathrm{tr}\,\mathbf{A}} = \frac{\lambda_1 \lambda_2}{\lambda_1 + \lambda_2}. \qquad \textbf{(15)}$$

SUBPIXEL CORNER LOCATION

- Locations are detected keypoints are usually at integer coordinates.
- To gain more accurate real-number coordinates, need to run subpixel algorithm.
- General idea: starting with an approximate location of a corner, find the accurate location which lies at the intersections of edges.

ADAPTIVE NON-MAXIMAL SUPPRESSION

- Non-maximal suppression: look for local maximal as keypoints.
- Can lead to uneven distribution of detected keypoints.
- Brown et al. used adaptive non-maximal suppression:
  - local maximal
  - response value is significantly larger than those of its neighbors

Strongest 250

Strongest 500

ANMS 250, $r = 24$

ANMS 500, $r = 16$

Measure difference as Euclidean distance between feature vectors:

$$d(\mathbf{u}, \mathbf{v}) = \left(\sum_i (u_i - v_i)^2 \right)^{1/2} \qquad \textbf{(23)}$$

Some possible matching strategies:

- Return all feature vectors with *d* smaller than a threshold.
- Nearest neighbor: feature vector with smallest *d*.
- Nearest neighbor distance ratio:

$$\mathrm{NNDR} = \frac{d_1}{d_2} \qquad \textbf{(24)}$$

d_1, d_2: distances to the nearest and 2nd nearest neighbors.

Nearest neighbor is a good match, if NNDR is small.

SCALE INVARIANCE

The scale of the object of interest may vary in different images.

Inefficient solution:

- Extract features at many different scales.
- Combine them to the object's known features at a particular scale.

Efficient solution:

- Extract features that are invariant to scale.

Scale Invariant Feature Transform (SIFT).

Convolve input image *I* with Gaussian *G* of various scale *σ*:

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \qquad \textbf{(16)}$$

Where,

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \qquad \textbf{(17)}$$

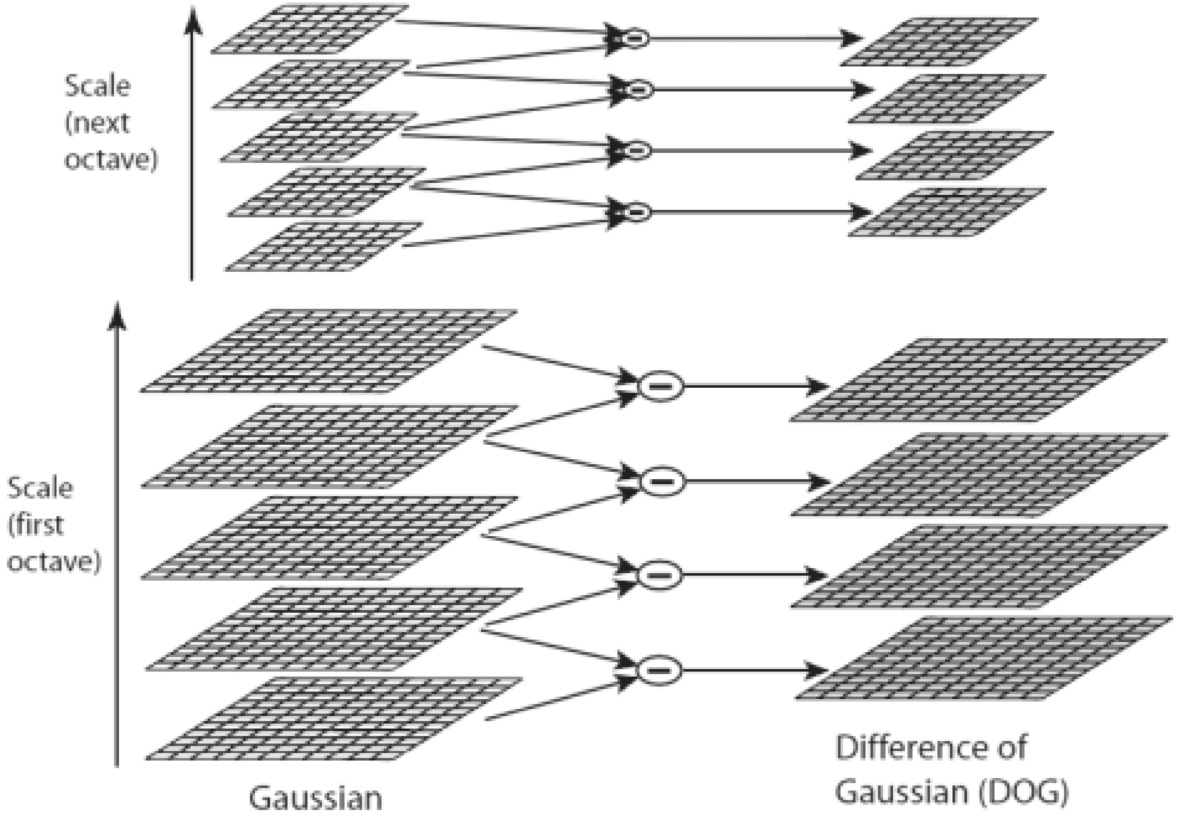
This produces *L* at different scales.

To detect stable keypoint, convolve image *I* with difference of Gaussian:

$$\begin{aligned} D(x, y, \sigma) &= (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma). \end{aligned} \qquad \textbf{(18)}$$

SIFT

- Have 3 different scales within each octave (doubling of σ).
- To produce *D,* successive DOG images are subtracted.
- *D* images in a lower octave are downsampled by factor of 2.



Find local maximum and minimum of D(*x, y, σ*):

- Compare a sample point with its 8 neighbors in the same scale and 9 neighbors in the scale above and below.
- Choose it if it is larger or smaller than all neighbors.
- Get position *x, y* and scale *σ* of keypoint.

Orientation of keypoint:

$$\theta(x,y) = \tan^{-1}\frac{L(x,y+1)-L(x,y-1)}{L(x+1,y)-L(x-1,y)}. \qquad \textbf{(19)}$$

Gradient magnitude of keypoint:

$$m(x,y) = \sqrt{(L(x+1,y)-L(x-1,y))^2+(L(x,y+1)-L(x,y-1))^2}. \qquad \textbf{(20)}$$

Keypoints found may contain edge points.

Edge points are not good since different edge points along an edge may look the same.

To discard edge points, form the Hessian **H** for each keypoint

$$\mathbf{H} = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} \quad (21)$$

and discard those for which

$$\frac{\operatorname{tr} \mathbf{H}^2}{\det \mathbf{H}} > 10 \quad (22)$$

BETTER INVARIANCE

Rotation invariance

- Find dominant orientation of keypoint.
- Normalize orientation.

Affine invariance

- Adjust ellipse to auto-correlation function or Hessian.
- Apply PCA to determine principal axes.
- Normalize according to principal axes.

FEATURE DESCRIPTORS

Why need feature descriptors?

- Keypoints provide just the positions of strong features.
- To combine them across different images, have to describe them by extracting feature descriptors.

Type of feature descriptors:

- Able to match corresponding points across images accurately.
- Invariant to scale, orientation, or even affine transformation.
- Invariant to lighting difference.

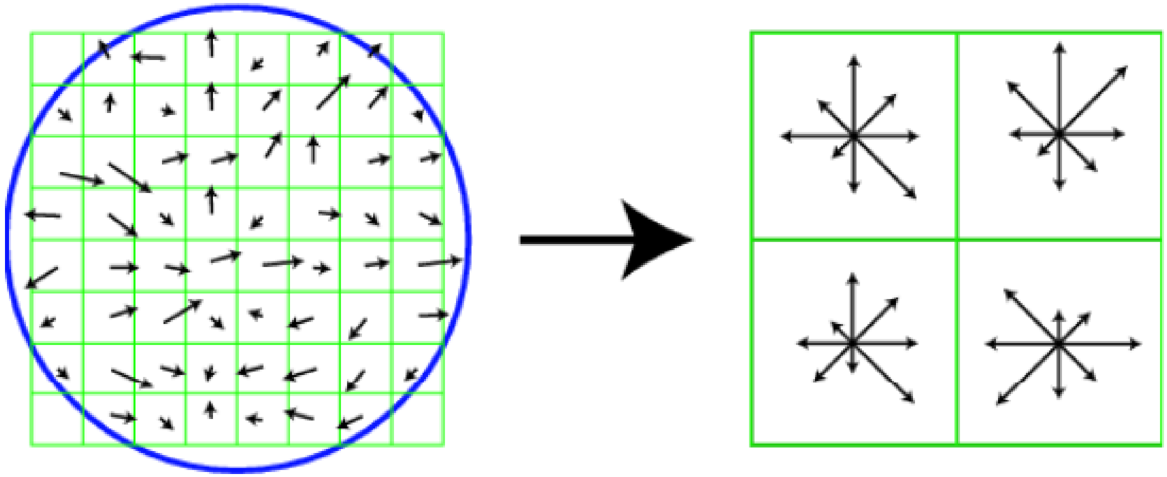


Image gradients **Keypoints descriptor**

- Take 16x16 square window around detected interest point
- Coordinates and gradient orientations are measured relative to keypoint orientation to achieve orientation invariance.
- Weighted by Gaussian window.
- Collect into 4×4 orientation histograms with 8 orientation bins. Create histogram of surviving edge orientations.

FEATURE DESCRIPTORS

- Bin value = sum of gradient magnitudes near that orientation.
- 16 cells * 8 orientations = 128 dimensional descriptor.
- Normalize feature vector to unit length to lower the effect of linear illumination change.
- To lower the effect of nonlinear illumination change, threshold feature values to 0.2 and renormalize feature vector to unit length.

Alternatives of **SIFT**:

- PCA-SIFT - Use PCA to lower the dimensionality.
- SURF (Speeded Up Robust Features) - Use box filter to approximate derivatives.
- GLOH (Gradient Location-Orientation Histogram) - Use log-polar binning structure.

GLOH performs the best, followed by SIFT.

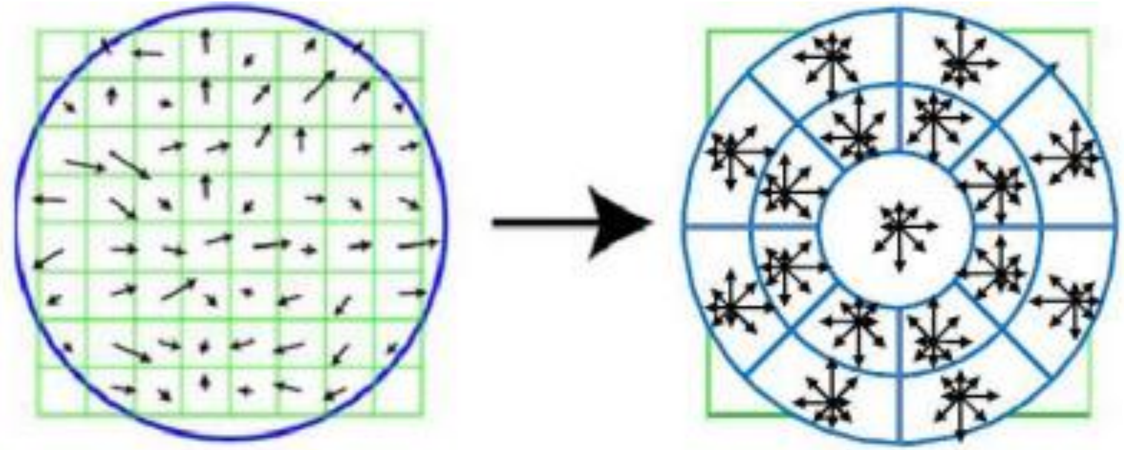
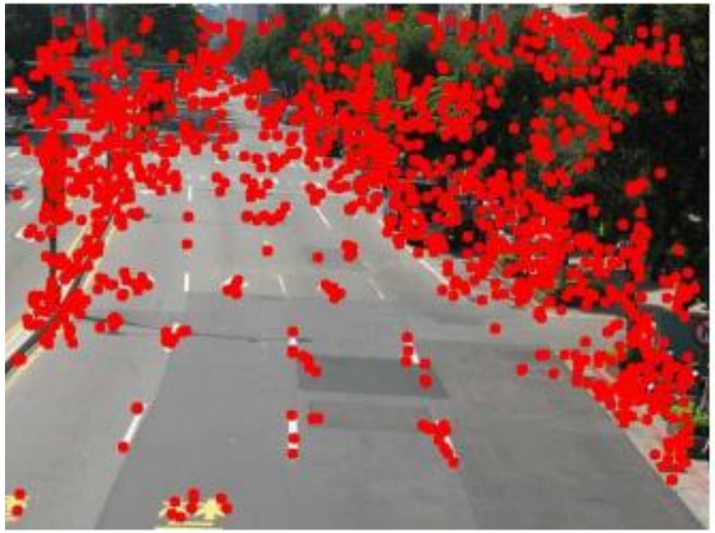
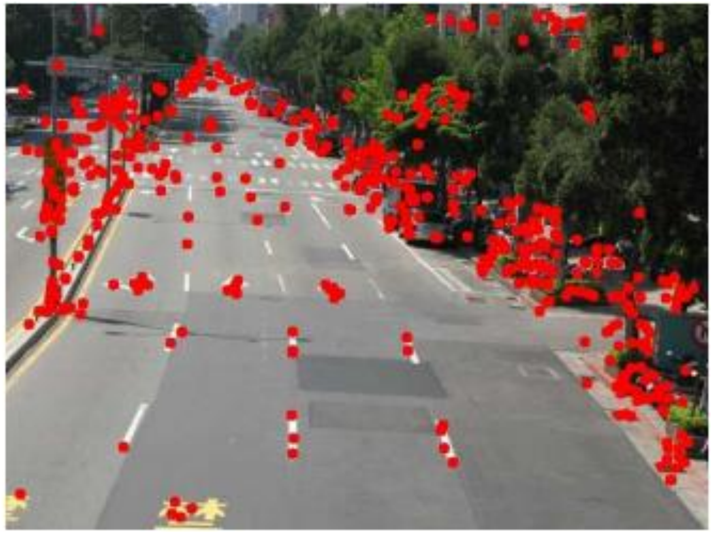


Image gradients **Keypoints descriptor**

OTHERS FEATURE DESCRIPTORS

Sample detected SURF keypoints (without non-maximal suppression):

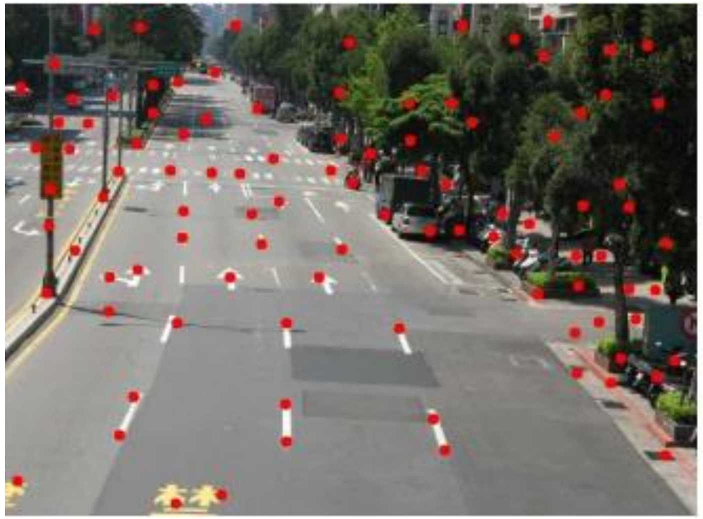




Low threshold provides many cluttered keypoints.

Higher threshold provides fewer keypoints, yet cluttered.

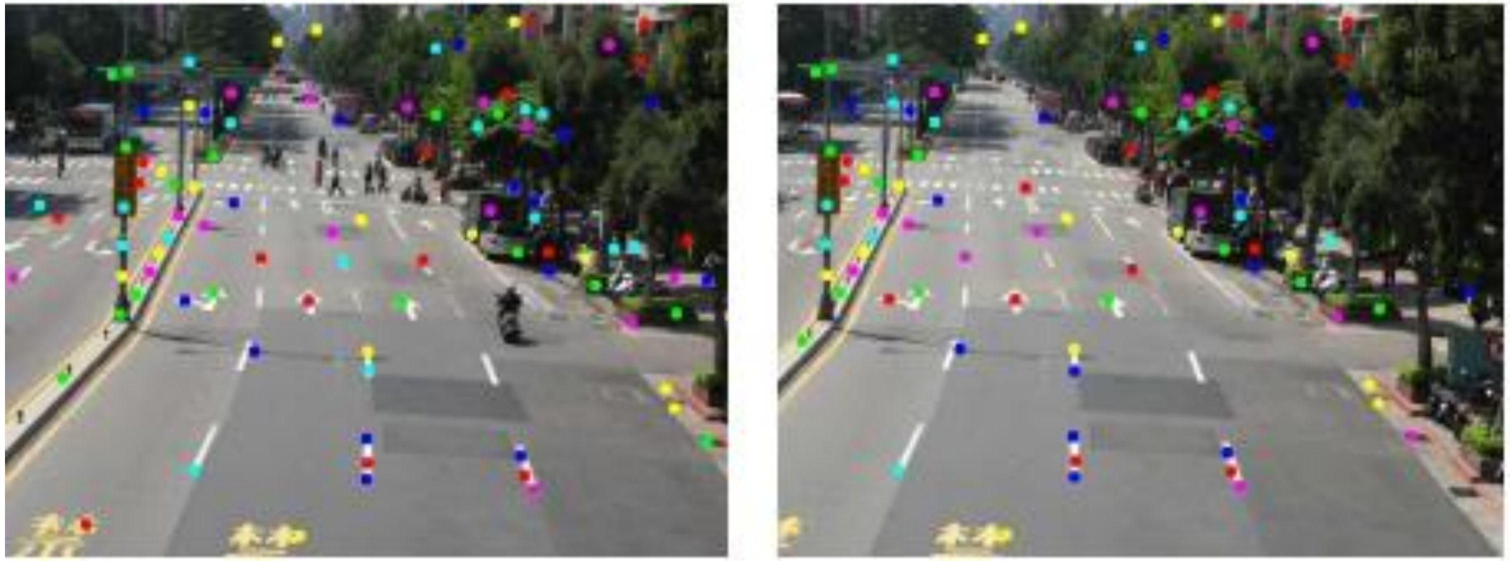
With adaptive non-maximal suppression, keypoints are well spread out:



Top 100 keypoints.

Top 200 keypoints.

Sample matching results: SURF, nearest neighbors with min. distance.



Some matches are correct, others are not.

Can include other info like color to improve match accuracy.

Generally, no perfect matching results.

FEATURE MATCHING

- Feature matching methods can provide false matches.
- Manually select good matches.
- Or use robust method to remove false matches:
    - True matches are consistent and have small errors.
    - False matches are inconsistent and have large errors.
- Nearest neighbor search is computationally expensive.
    - Require efficient algorithm, e.g., using *k*-D Tree.
    - *k*-D Tree is not more efficient than exhaustive search for large dimensionality, e.g., > 20.

SUMMARY

- Harris corner detector and Tomasi’s algorithm find corner points.
- SIFT keypoint: invariant to scale.
- SIFT descriptors: invariant to scale, orientation, illumination change.
- Variants of SIFT: PCA-SIFT, SURF, GLOH.

REFERENCES

Brown, M. (n.d.). *Suppose you want to create a panorama*.

Wee Kheng, L. (n.d.). *Feature Detection and Matching CS4243 Computer Vision and Pattern Recognition*.

DRIVES
Development and Research on Innovative Vocational Education Skills

Ana Luísa Sousa

- PhD student in Information System and Tecnologies
- Research Collaborator of the Algoritmi Research Center

iD 0000-0001-5731-3583

Regina Sousa

- PhD student in Biomedical Engineering
- Research Collaborator of the Algoritmi Research Center

iD 0000-0002-2988-196X

Diana Ferreira

- PhD student in Biomedical Engineering
- Research Collaborator of the Algoritmi Research Center

iD 0000-0003-2326-2153



António Abelha
- Assistant Professor at the University of Minho
- Integrated Researcher of the Algoritmi Research Center

0000-0001-6457-0756

José Machado
- Associate Professor with Habilitation at the University of Minho
- Integrated Researcher of the Algoritmi Research Center

0000-0003-4121-6169

Victor Alves
- Assistant Professor at the University of Minho
- Integrated Researcher of the Algoritmi Research Center

0000-0003-1819-7051

REFERENCE TO AUTHORS

This Training Material has been certified according to the rules of **ECQA – European Certification and Qualification Association.**

The Training Material was developed within the international job role committee **“Computer Vision Expert”:**

UMINHO – University of Minho (https://www.uminho.pt/PT)

The development of the training material was partly funded by the EU under Blueprint Project DRIVES.

DRIVES

Development and Research on Innovative Vocational Education Skills

Thank you for your attention

DRIVES project is project under **The Blueprint for Sectoral Cooperation on Skills in Automotive Sector**, as part of New Skills Agenda.

The aim of the Blueprint is **to support an overall sectoral strategy and to develop concrete actions to address short and medium term skills needs.**

Follow DRIVES project at:

More information at:

www.project-drives.eu

Co-funded by the Erasmus+ Programme of the European Union

The Development and Research on Innovative Vocational Educational Skills project (DRIVES) is co-funded by the Erasmus+ Programme of the European Union under the agreement 591988-EPP-1-2017-1-CZ-EPPKA2-SSA-B. The European Commission support for the production of this publication does not constitute endorsement of the contents which reflects the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.