Image Descriptors: SIFT, SURF & HOG

SIFT
Scale-Invariant Feature Transform

Detects and describes local features

- ✓ Scale invariant
- ✓ Rotation invariant
- ✓ Partial illumination invariance

SURF
Speeded Up Robust Features

Faster approximation of SIFT

- ✓ Improved speed
- ✓ Similar robustness
- ✓ Efficient computation

HOG
Histogram of Oriented Gradients

Captures edge orientation distribution

- ✓ Shape description
- ✓ Edge patterns
- ✓ Object detection

Capabilities
Enable robust object recognition and image matching

Historical Context: Widely used before deep learning era for computer vision tasks

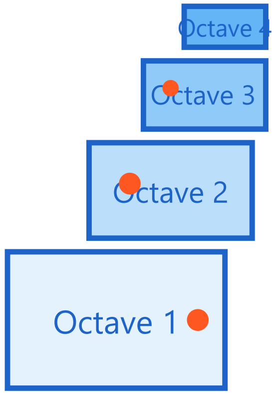
Current Use: Still valuable for lightweight applications and limited data scenarios

🔍 SIFT: Scale-Invariant Feature Transform

💡 Core Principle

SIFT finds feature points in **scale space** and generates descriptors using **gradient orientation histograms** around each keypoint.

DoG Pyramid (Difference of Gaussians)



Multiple Scales

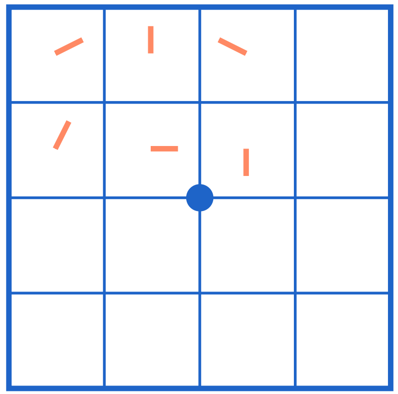
Apply Gaussian blur at each scale

● Keypoints

Detected at extrema locations

Divide image into multiple scales and compute DoG to find scale-invariant features

SIFT Descriptor (128-dimensional)



4×4 = 16 blocks

Each block: 8-bin histogram

16 × 8 = 128

dimensional vector

Divide 16x16 region into 4x4 blocks and compute 8-direction gradient histogram per block

1 Scale-Space Extrema Detection

Find extrema in DoG pyramid → Detect scale-invariant keypoint candidates

↓

2 Keypoint Localization

Refine with Taylor expansion → Remove unstable points

↓

3

Orientation Assignment

Gradient orientation histogram → Assign dominant orientation (rotation invariance)

↓

4

Descriptor Generation

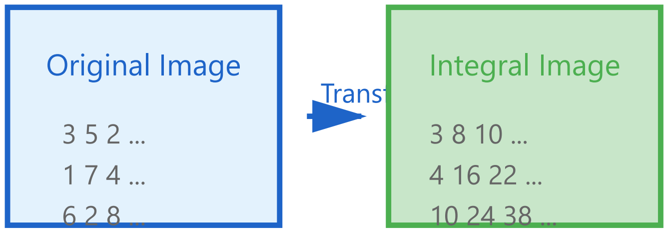
4x4 blocks × 8 orientations = 128-dimensional feature vector

⚡ SURF: Speeded Up Robust Features

💡 Core Principle

SURF uses **Integral Image** and **Box Filters** to approximate SIFT faster. Detects keypoints based on Hessian matrix.

Integral Image

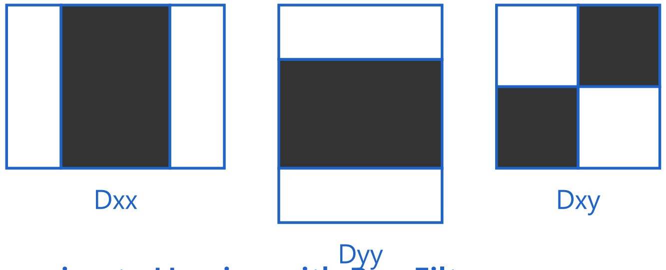


Fast Region Sum

Calculate sum of any rectangle in 4 operations

With Integral Image, box filters can be computed in O(1) time

Fast-Hessian Detector



Approximate Hessian with Box Filters

$\mathrm{Det}(H) \approx Dxx \cdot Dyy - (0.9 \cdot Dxy)^2$

Max value location = Keypoint

Quickly approximate Hessian matrix with 3 box filters to detect blobs

1 **Generate Integral Image**

Original image → Integral Image (preprocessing)

↓

2 **Fast-Hessian Detector**

Calculate Hessian determinant with box filters → Detect keypoints

↓

Orientation via Haar Wavelet

3 Determine dominant orientation using Haar wavelet responses

↓

4 **64D Descriptor**

4x4 sub-regions × 4D Haar response = 64-dimensional vector

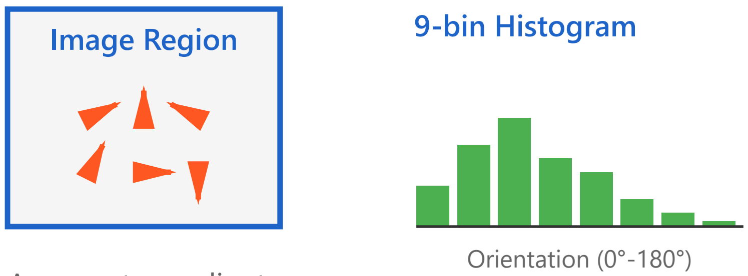
⚡ **Speed Improvement:** 2-3x faster than SIFT, with Integral Image enabling same speed for all scales

📊 HOG: Histogram of Oriented Gradients

💡 Core Principle

HOG divides images into small **cells** and computes **gradient orientation histograms** in each cell. Normalizes by blocks to create illumination-robust shape descriptors.

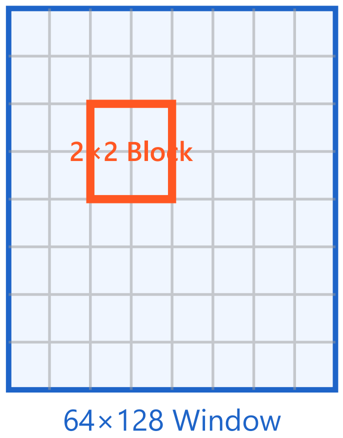
Gradient Computation & Orientation



Aggregate gradient orientations in each cell (8x8) to 9 bins

Calculate gradient magnitude and direction at each pixel and aggregate into histogram

Cell & Block Structure



Structure:

- Cell: 8×8 pixels
- Block: 2×2 cells
- Stride: 8 pixels

Dimension:

7 blocks (H) × 15 blocks (V) × 4 cells × 9 orientations **= 3,780 dimensions**

Group 2×2 cells into blocks and normalize to make robust against illumination changes

1 **Gradient Computation**
Calculate x, y direction gradients at each pixel (using [-1,0,1] filter)

↓

2 **Orientation Binning**
Divide into 8×8 cells, generate 9-bin orientation histogram in each cell

↓

3 Block Normalization

Group into 2×2 cell blocks and apply L2 normalization (50% overlap)

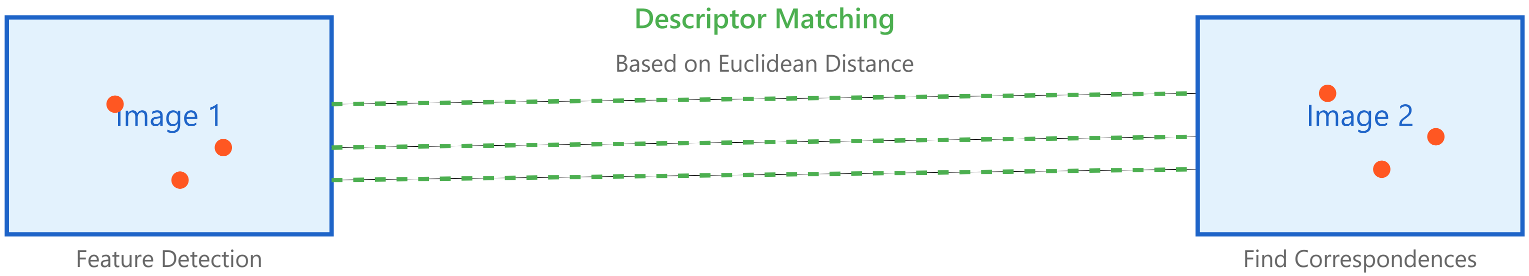
↓

4 Feature Vector

Concatenate all block histograms → 3,780-dimensional descriptor

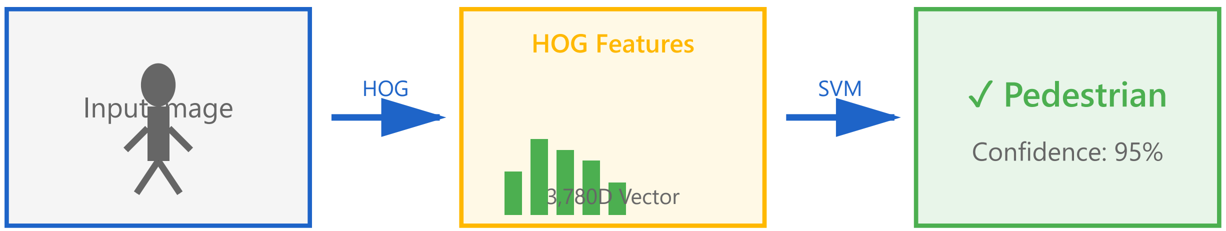
🎯 Real-world Applications

Image Matching (SIFT/SURF)



Extract features with SIFT/SURF from two images and match descriptors to find correspondences

Object Detection (HOG + SVM)



Extract shape features with HOG → Classify with SVM

Extract HOG features from sliding windows and detect pedestrians with pre-trained SVM

📋 Detailed Algorithm Comparison

Characteristic	SIFT	SURF	HOG
Core Method	DoG (Difference of Gaussians)	Fast-Hessian (Box Filter)	Gradient Orientation Histogram
Feature Detection	Scale-space extrema	Hessian determinant maxima	Dense sampling (entire window)
Descriptor Dimension	128-dimensional	64-D (standard) / 128-D	3,780-D (64x128 window)
Computation Speed	Slow (baseline)	Fast (2-3x faster than SIFT)	Medium (depends on window size)
Scale Invariance	Excellent	Excellent	Limited (fixed window)
Rotation Invariance	Excellent	Excellent	Limited
Main Applications	Image matching, panorama, 3D reconstruction	Real-time tracking, video processing	Pedestrian detection, object recognition
Patent Status	Expired (2020)	Patent protected	Free to use

🎯 Selection Guide

Need high accuracy → SIFT (image stitching, 3D reconstruction, object recognition)

Need real-time processing → SURF (video tracking, AR applications, real-time matching)

Object detection goal → HOG + SVM (pedestrian, vehicle, face detection)

🎓 **Key Difference:** SIFT/SURF are keypoint-based, HOG is dense descriptor. Choose based on your application.

💡 **Deep Learning Era:** These algorithms are still useful for limited data or lightweight systems.