
Computer Vision Techniques to Identify Rice Pests
Christopher Harris
CS 395 – Emerging Topics in CS
Spring 2024

Overview

- Introduce computer vision techniques
  - History of computer vision
  - The logic behind different techniques
    - Local appearance-based methods
    - Interest point-based methods
  - Examples
- Apply these technique to rice pests
  - Describe the problem
  - Describe the experimental process
  - Results and Analysis

A Brief History of Feature Detection Methods

- **1970s**: Researchers begin to develop algorithms for detecting and tracking features in images.
- **1988**: Chris Harris (not me) and Mike Stephens develop the Harris corner detector.
- **1999**: David Lowe develops the Scale-Invariant Feature Transform (SIFT).
- **2006**: Herbert Bay, Tobias Tuytelaars, and Luc Van Gool develop the Speeded-Up Robust Features (SURF) detector.
- **2010**: Michael Calonder, Michael Mikolajczyk, and Andrew Torralba develop the Binary Robust Independent Elementary Features (BRIEF) detector.

Types of Feature Extraction Methods

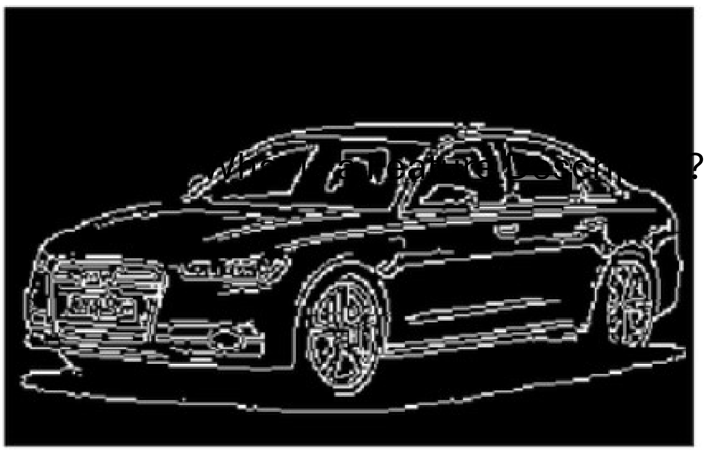
- **Local appearance-based methods**
  - BRIEF (Binary Robust Independent Elementary Features)
  - HOG (Histogram of Oriented Gradients)
  - LBP (Local Binary Pattern)
- **Interest point-based methods**
  - SIFT (Scale-Invariant Feature Transform)
  - SURF (Speeded-Up Robust Features)
- **Hybrid**
  - ORB (Oriented FAST and Rotated BRIEF)

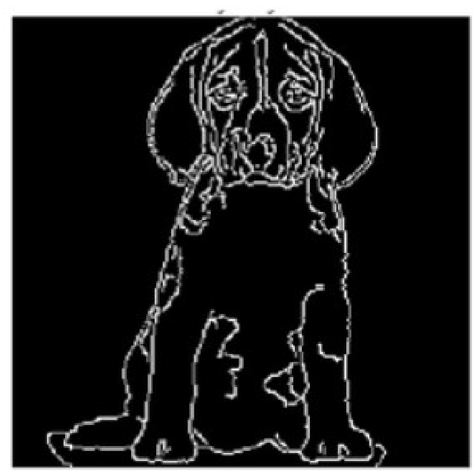
Histogram of Oriented Gradients (HOG)

- A popular feature descriptor technique in computer vision and image processing.
- It analyzes the distribution of edge orientations within an object to describe its shape and appearance.
- The HOG method involves computing the gradient magnitude and orientation for each pixel in an image and then dividing the image into small cells.

What is a Feature Descriptor?

What is a Feature Descriptor?





How does HOG work?

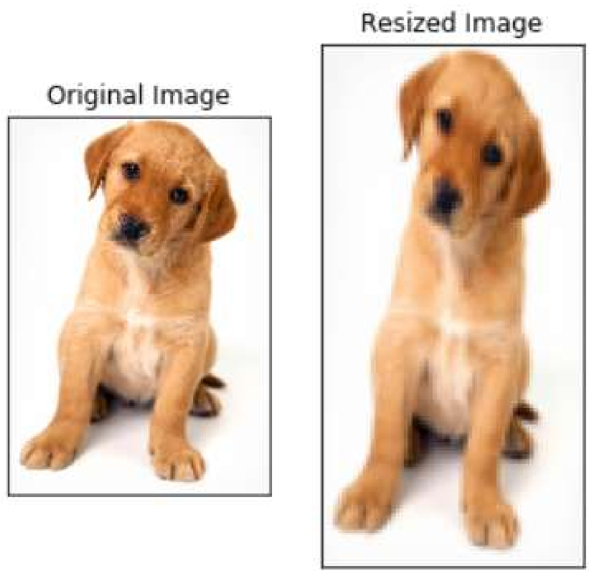
- The HOG descriptor focuses on the structure or the shape of an object.
- In the case of edge features, we only identify if the pixel is an edge or not. HOG is able to provide the edge direction as well.
- This is done by extracting the **gradient and orientation** of the edges

How does HOG work?

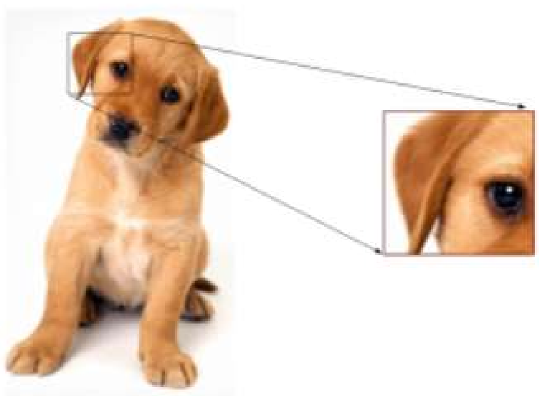
- These orientations are calculated in **‘localized’ portions**.
  - This means that the complete image is broken down into smaller regions and for each region, the gradients and orientation are calculated.
- Finally, the HOG generates a **Histogram** for each of these regions separately.
  - The histograms are created using the gradients and orientations of the pixel values, hence the name ‘Histogram of Oriented Gradients’

Resize the Image

The first step of calculation in many feature detectors in image pre-processing is to ensure normalized color and gamma values.



Examine the Pixel Contrast with Neighboring Pixels



121	10	78	96	125
48	152	68	125	111
145	78	85	89	65
154	214	56	200	66
214	87	45	102	45

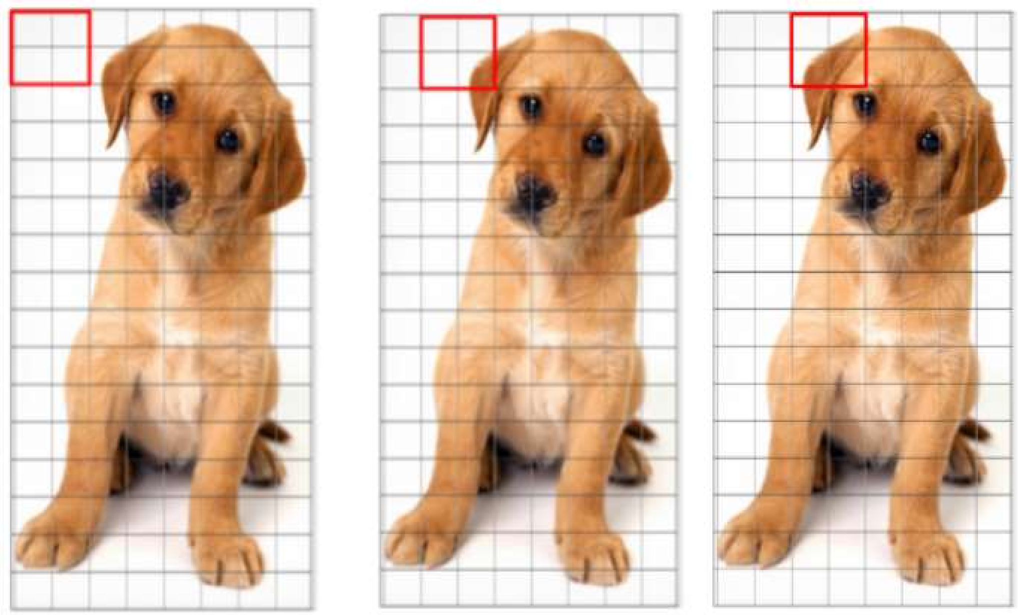
Determine Gradients for X, Y

121	10	78	96	125
48	152	68	125	111
145	78	85	89	65
154	214	56	200	66
214	87	45	102	45

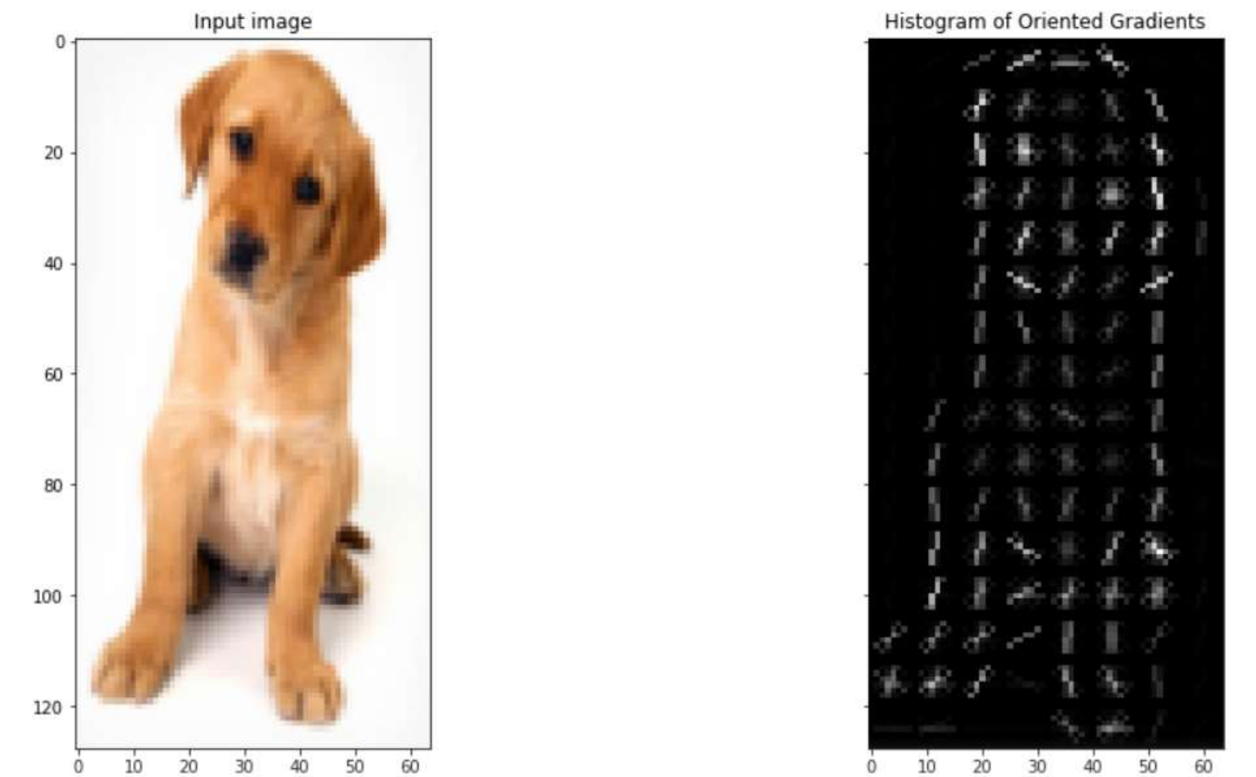
- Now, to determine the gradient (or change) in the x-direction, we need to subtract the value on the left from the pixel value on the right.
- Same with the y-direction
- Hence the resultant gradients in the x and y direction for this pixel are:
- Change in X direction(G_x) = 89 – 78 = 11
- Change in Y direction(G_y) = 68 – 56 = 8
- This process will give us two new matrices – one storing gradients in the x-direction and the other storing gradients in the y direction.
- **The magnitude would be higher when there is a sharp change in intensity, such as around the edges.**

Build a Histogram of Gradients

- A **histogram** is a plot that shows **the frequency distribution** of **a set of continuous data.**



Result



Input image
Histogram of Oriented Gradients
0
20
40
60
80
100
120
0
10
20
30
40
50
60

Steps: First… resize the image and remove saturation

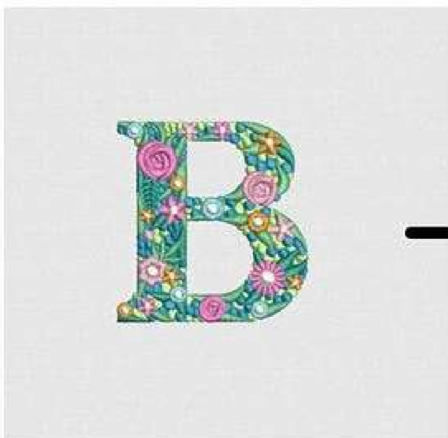


Figure 1

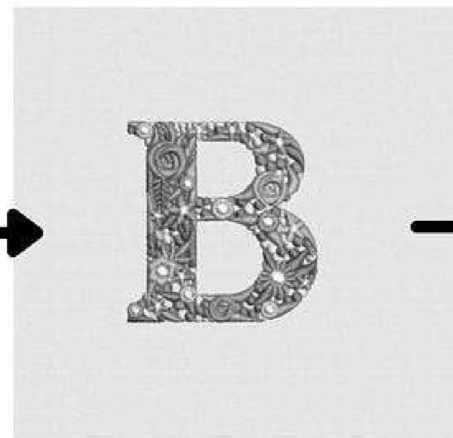


Figure 2

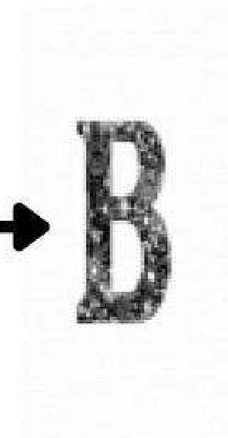


Figure 3

Next: Calculate Magnitude and Direction

$$Magnitude(\mu) = \sqrt{G_x^2 + G_y^2} \quad Angle(\theta) = |\tan^{-1}(G_y/G_x)$$

Figure 4

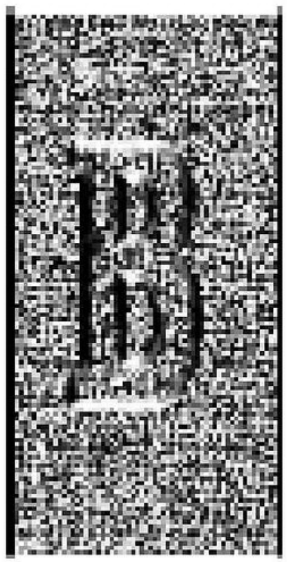


Figure 5

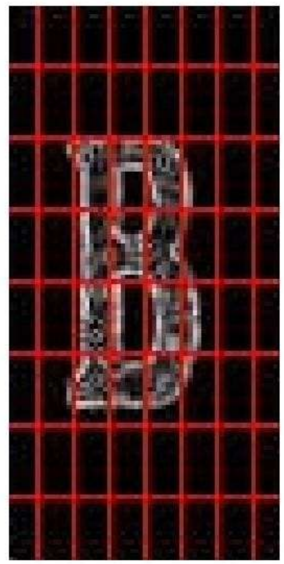


Figure 6

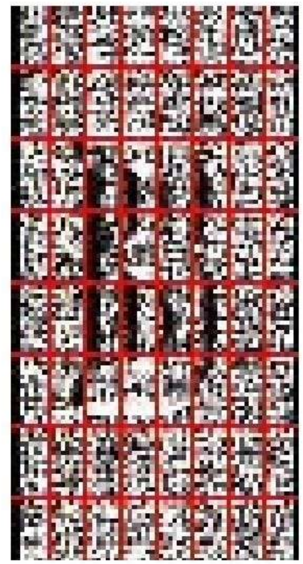
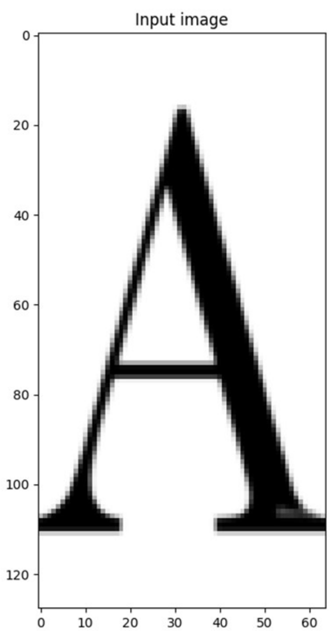
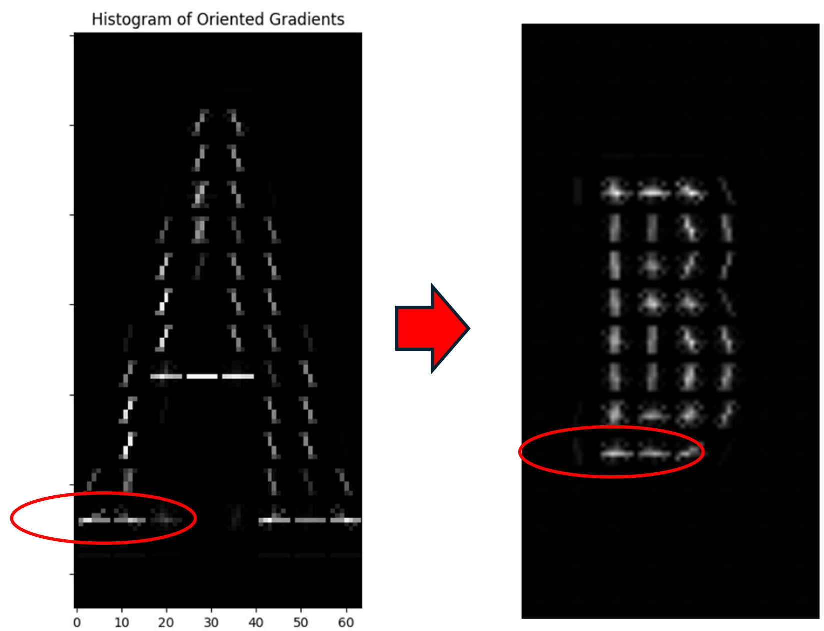


Figure 7

HOG

Compare with A… Where is the Overlap?





HOG

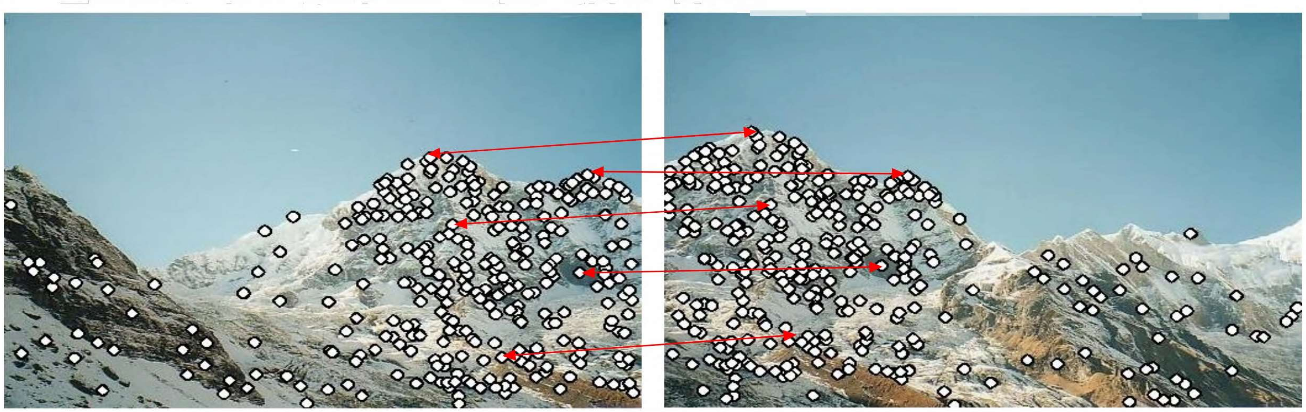
Overlap is only 0.132

SIFT (Scale Invariant Feature Transform)

- A type of **interest-point based method.**
- Designed to detect, describe, and match local features in images, invented by David Lowe in 1999.
- Applications include object recognition, robotic mapping and navigation, image stitching, 3D modeling, gesture recognition, video tracking, individual identification of wildlife and match moving.
- Patented… but patent expired in 2020.

Correspondence

- Fundamental to many of the core vision problems
  - Recognition Motion
  - tracking Multiview
  - geometry
- Local features are the key



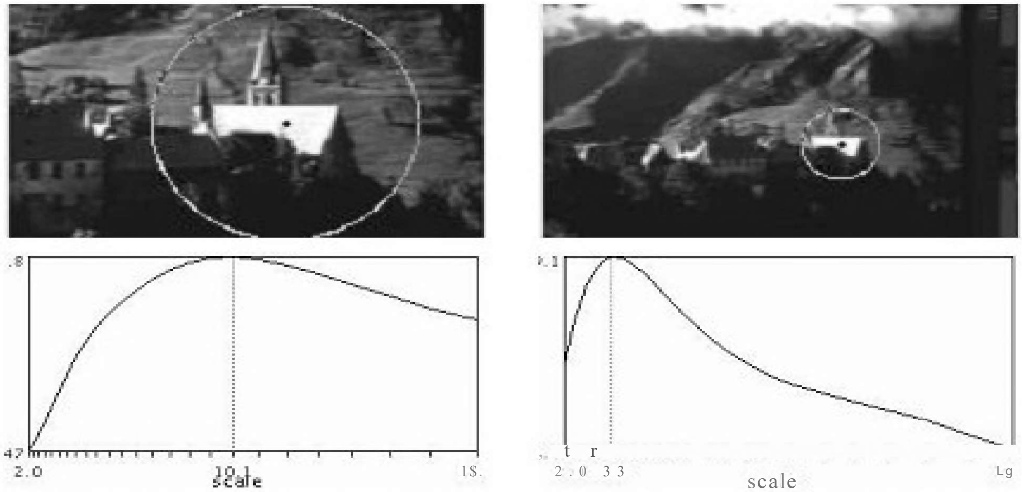
Ideal Interest Points/Regions

- Lots of them
- Repeatable
- Representative orientation/scale
- Fast to extract and match



Scale Selection

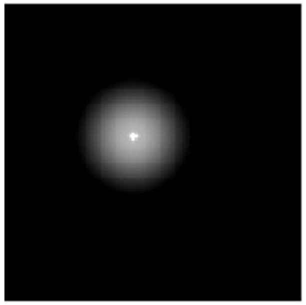
- Experimentally, Maxima of Laplacian-of-Gaussian gives best notion of scale:



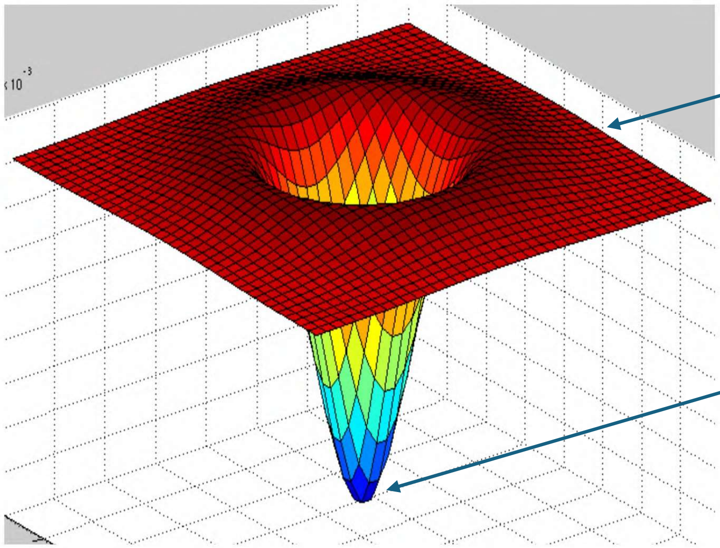
- However, it is expensive to calculate

See https://www.cse.psu.edu/~rtc12/CSE486/lecture11.pdf for more about this topic

DoG Efficiency



- The smoothed images need to be computed in any case for feature description.
- We need only to subtract two images.



Biggest difference

Smallest difference

"Difference of Gaussians" algorithm works by performing two different Gaussian blurs on the image, with a different blurring radius for each, and subtracting them to yield the result.

DoB Filter (Difference of Boxes')

- Even faster approximation is using box filters (by integral image)

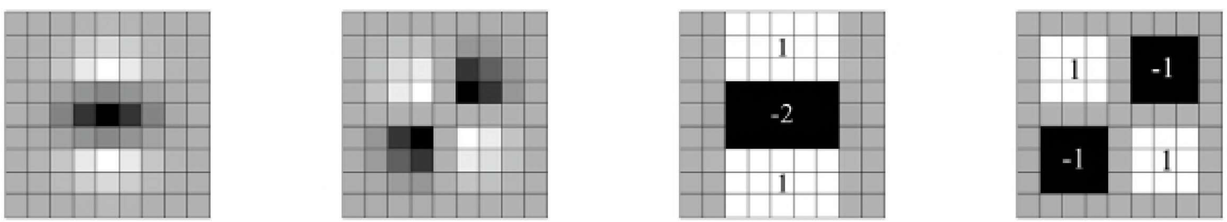
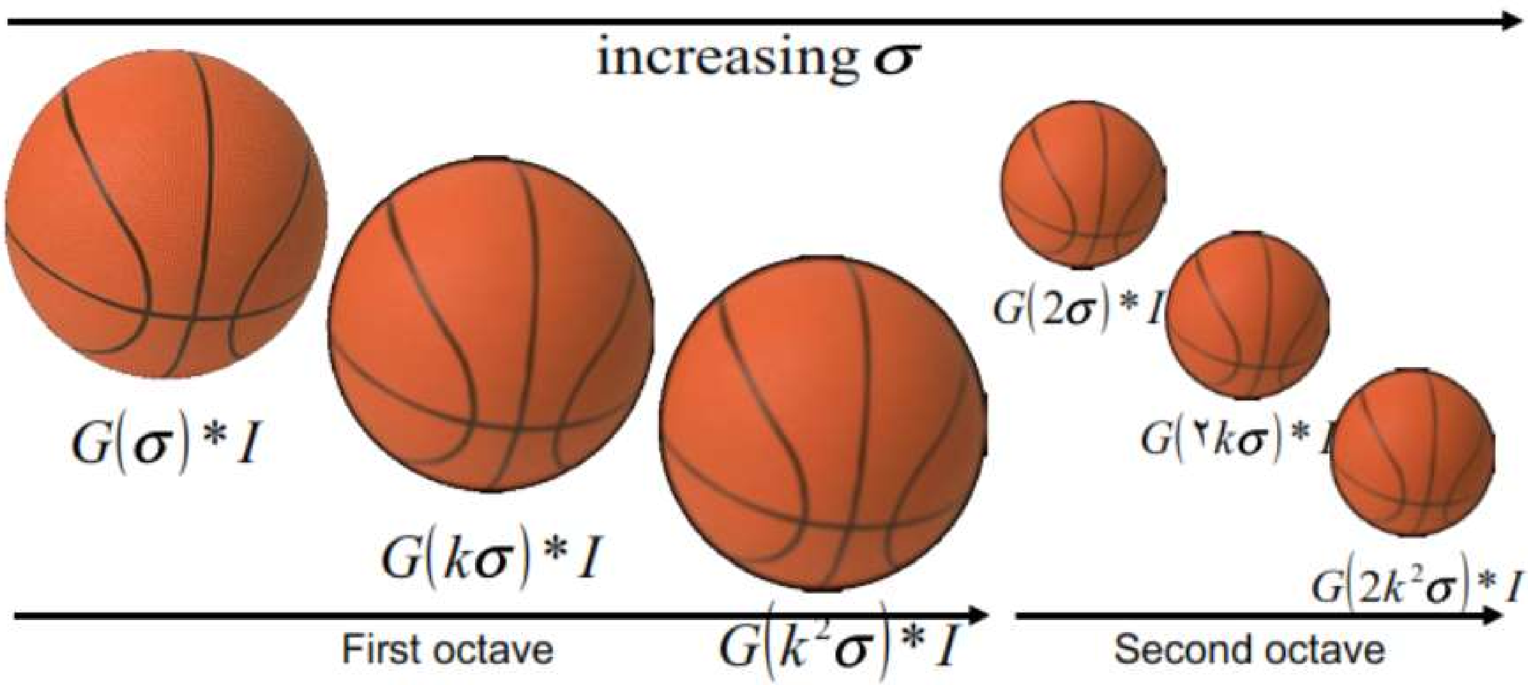


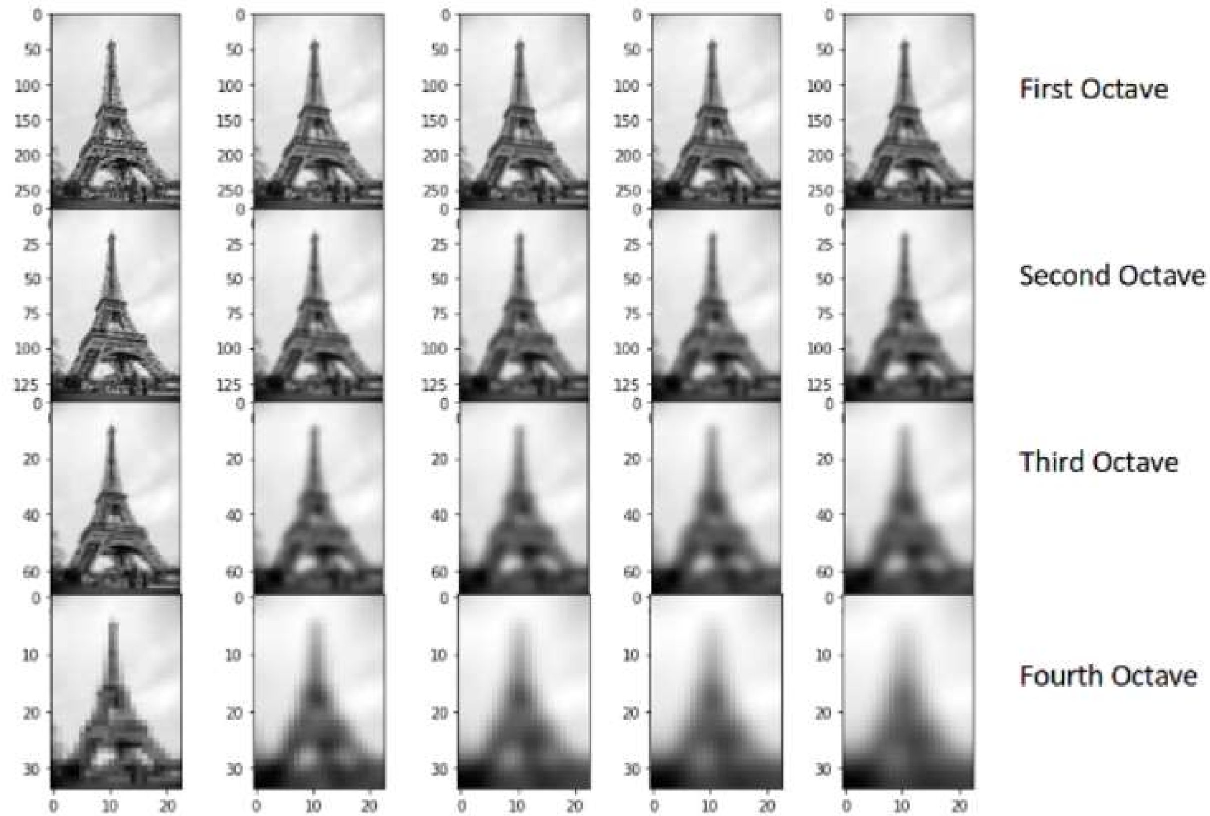
Fig.1. Left to right: the (discretised and cropped) Gaussian second order partial derivatives in y-direction and y-direction, and our approximations thereof using box filters. The grey regions are equal to zero.

Bay et al., ECCV 2006

Scale-Space Construction

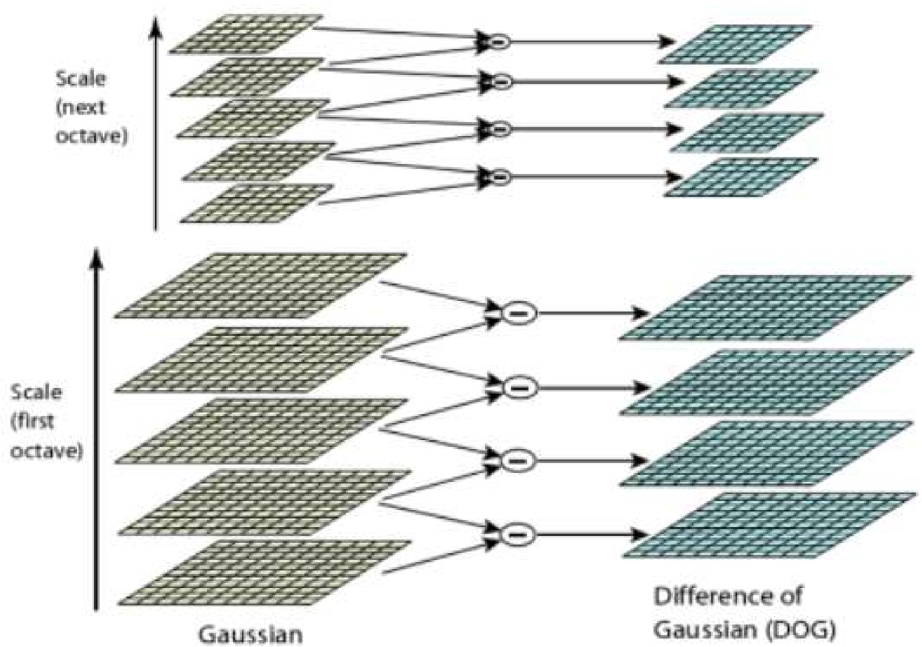
- **First construct scale-space:**

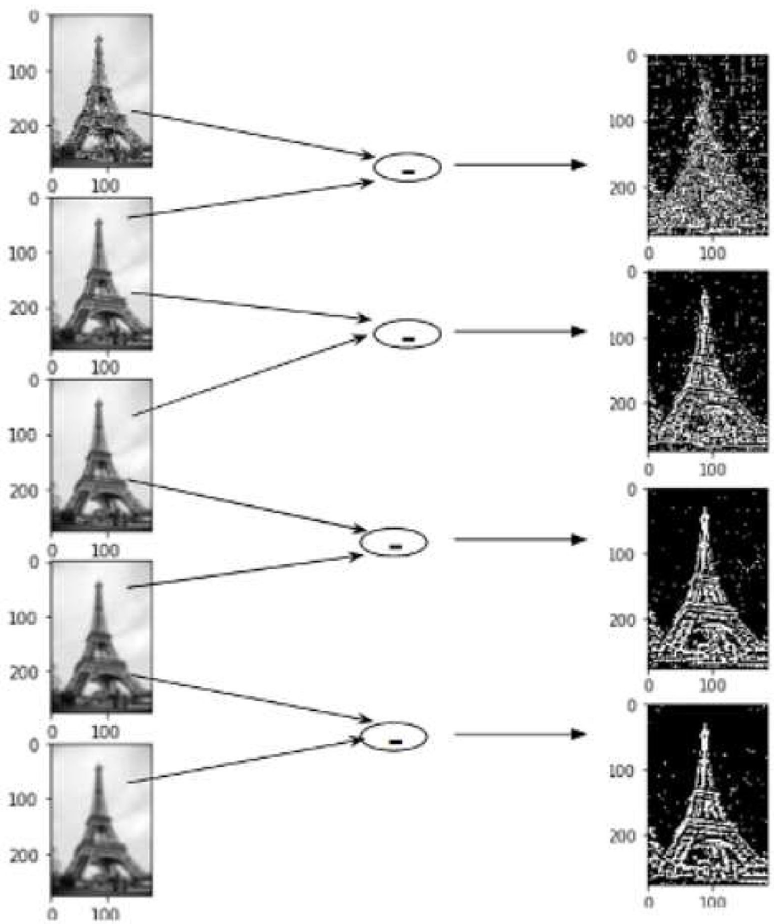



First Octave
Second Octave
Third Octave
Fourth Octave

Difference-of-Gaussianss

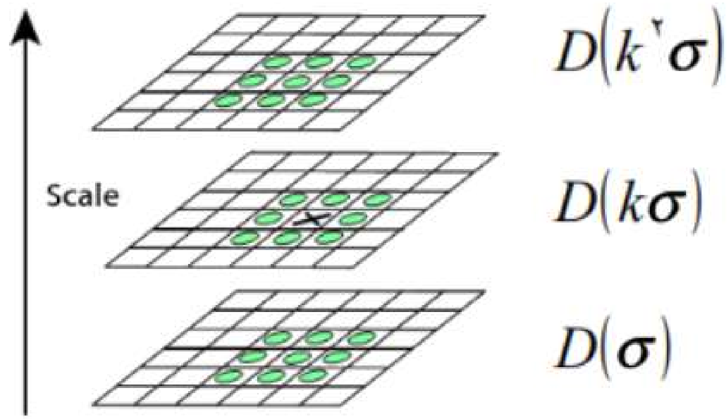
- Now take differences:





Scale-Space Extrema

- Choose all extrema within 3x3x3 neighborhood.
- Low cost – only several usually checked

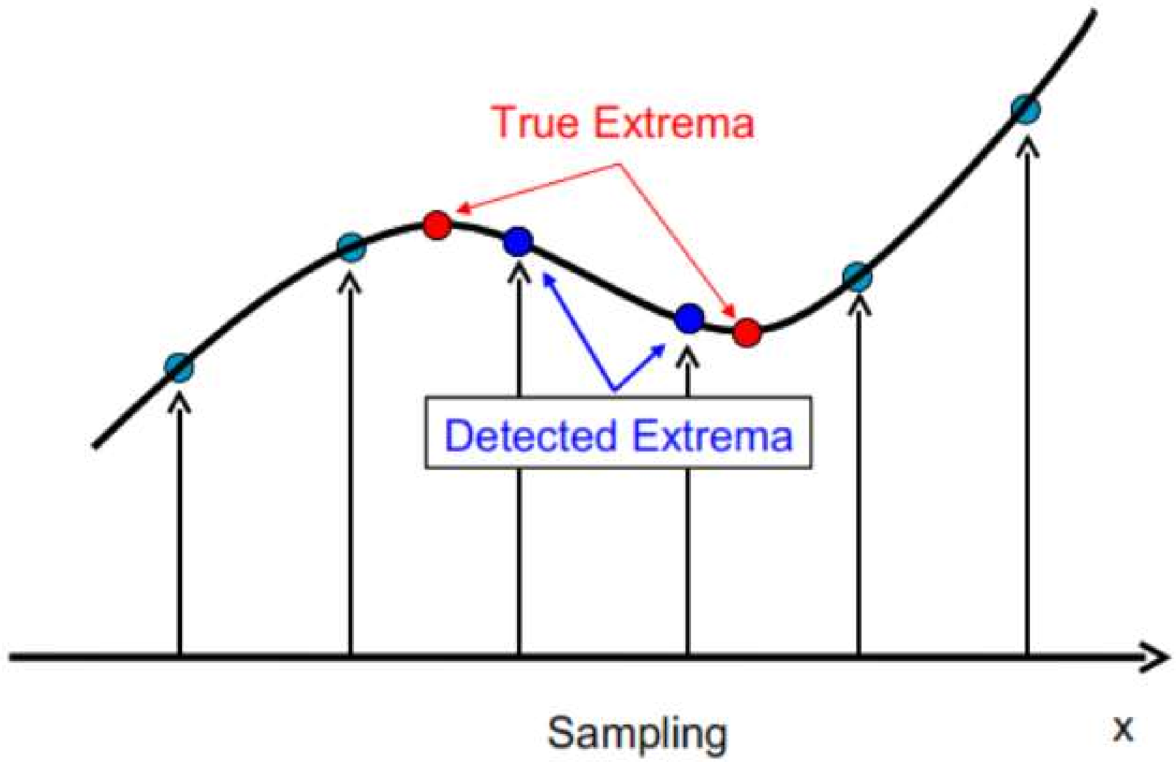


Keypoint Localization & Filtering

- Now we have much less points than pixels.
- However, still lots of points (~1000s)…
  - With only pixel-accuracy at best
    - At higher scales, this corresponds to several pixels in base image
  - And this includes many bad points

Keypoint Localization

- The problem:



Keypoint Localization

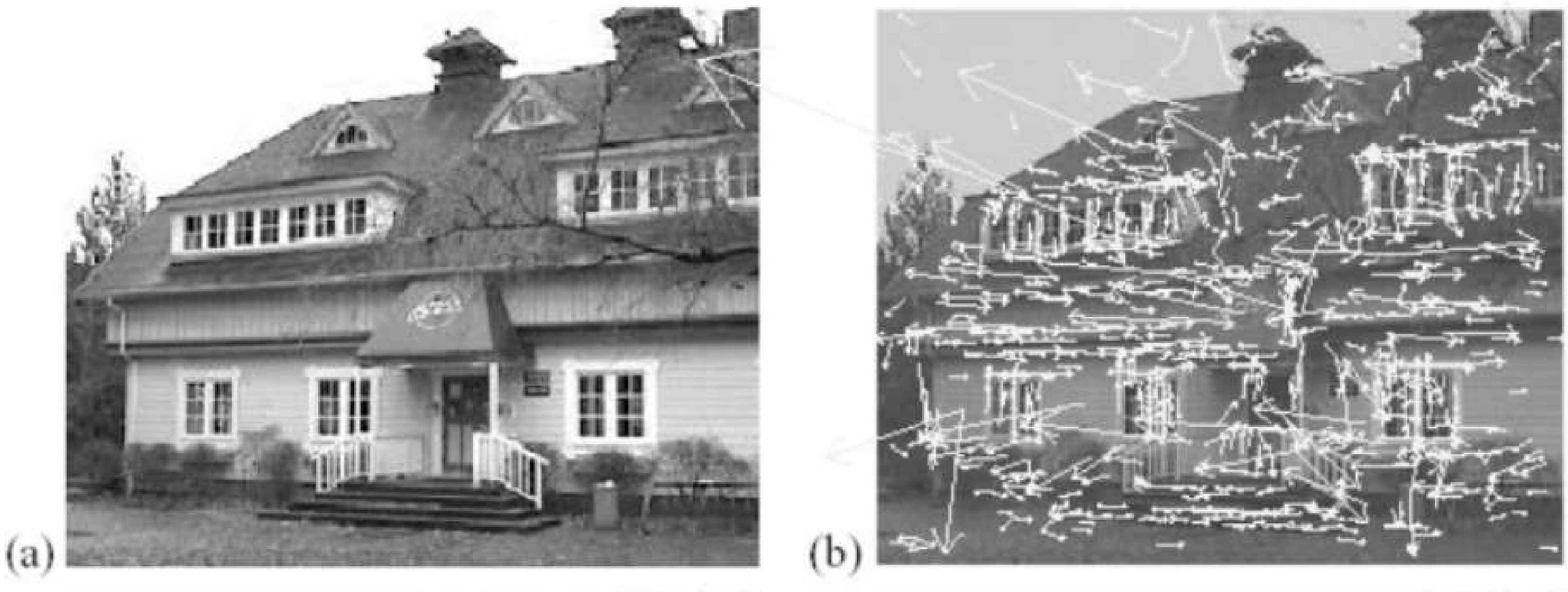
- The Solution:
  - Take Taylor series expansion:

$$D(\vec{x}) = D + \frac{\partial D^T}{\partial \vec{x}}\vec{x} + \frac{1}{2}\vec{x}^T \frac{\partial^2 D^T}{\partial \vec{x}^2}\vec{x}$$

  - Minimize to get true location of extrema:

$$\hat{x} = -\frac{\partial^2 D}{\partial \vec{x}^2}^{-1} \frac{\partial D}{\partial \vec{x}}$$

Keypoints



(a) 233x189 image
(b) 832 DOG extrema

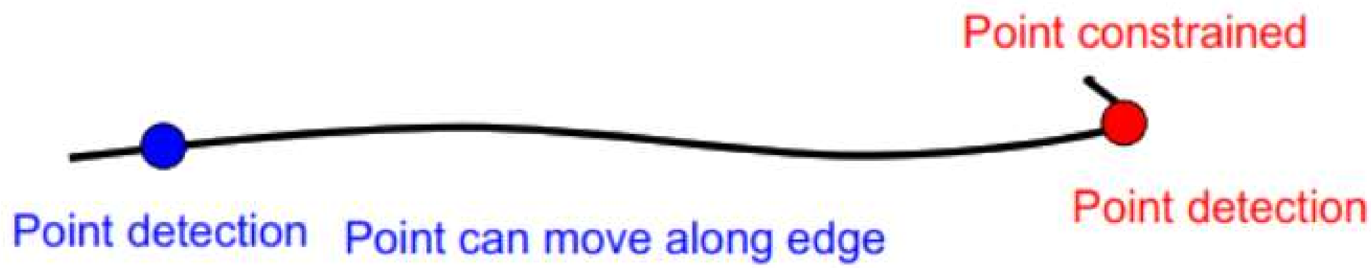
Keypoint Filtering - Low Contrast

- Reject points with bad contrast

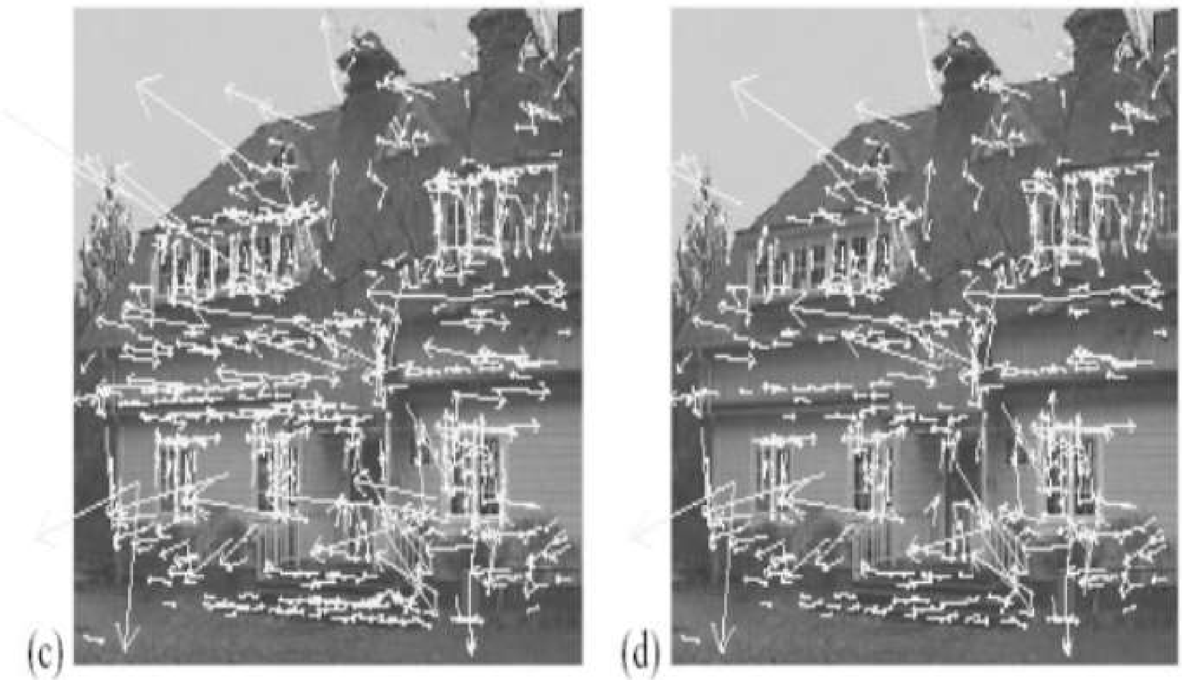
  $D(\hat{x})$ is smaller than 0.03 (image values in [0,1])

Keypoint Filtering - Edges

- Reject points with strong edge response in one direction only
- Like Harris - using Trace and Determinant of Hessian



Keypoint Filtering



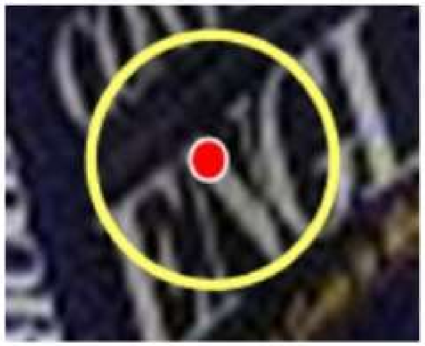
(c) 729 left after peak value threshold (from 832)
(d) 536 left after testing ratio of principle curvatures

Ideal Descriptors

- Robust to:
  - Affine transformation
  - Lighting
  - Noise
- Distinctive
- Fast to match
  - Not too large

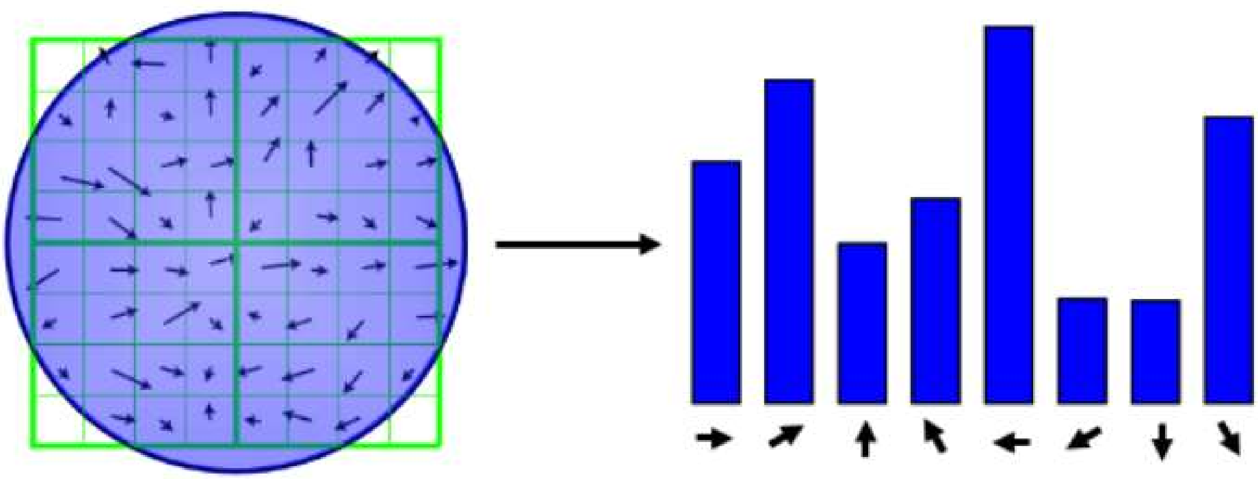
Orientation Assignment

- Now we have set of good points
- Choose a region around each point
  - Remove effects of scale and rotation



Orientation Assignment

- Create gradient histogram (36 bins)
  - Weighted by magnitude and Gaussian window (σ is 1.5 times that of the scale of a keypoint)



Orientation Assignment

- Any peak within 80% of the highest peak is used to create a keypoint with that orientation
- ~15% assigned multiplied orientations, but contribute significantly to the stability
- Finally a parabola is fit to the 3 histogram values closest to each peak to interpolate the peak position for better accuracy

SIFT Descriptor

- 4x4 Gradient window
- Histogram of 4x4 samples per window in 8 directions
- Gaussian weighting around center(σ is 0.5 times that of the scale of a keypoint)
- 4x4x8 = 128 dimensional feature vector

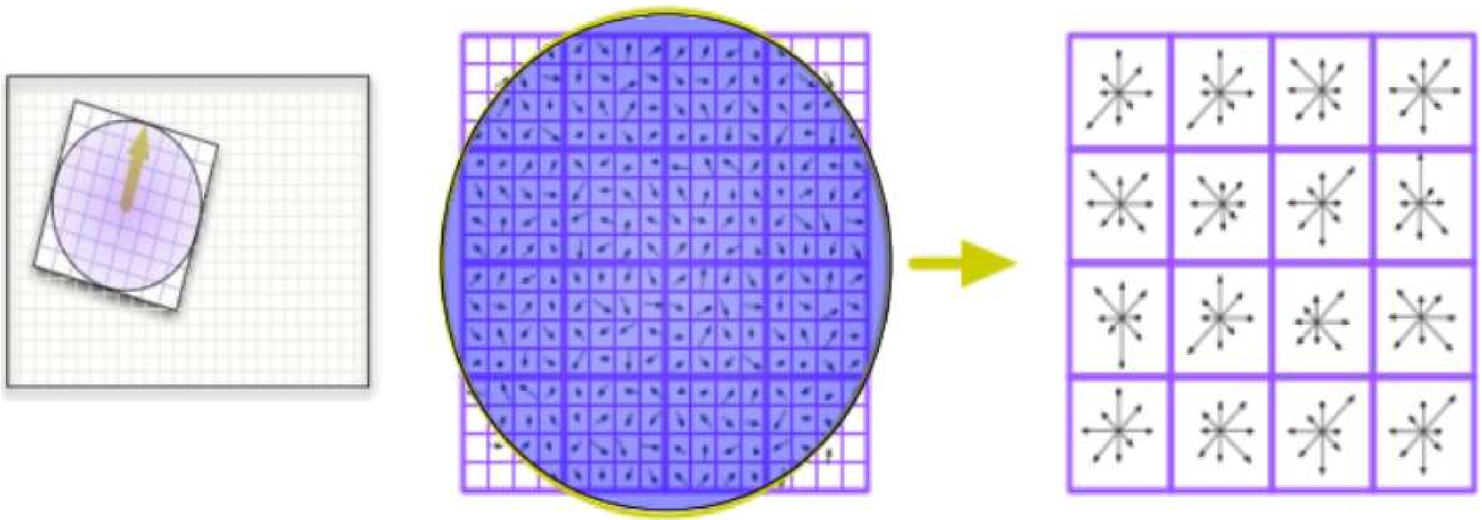


Image gradients Keypoint descriptor

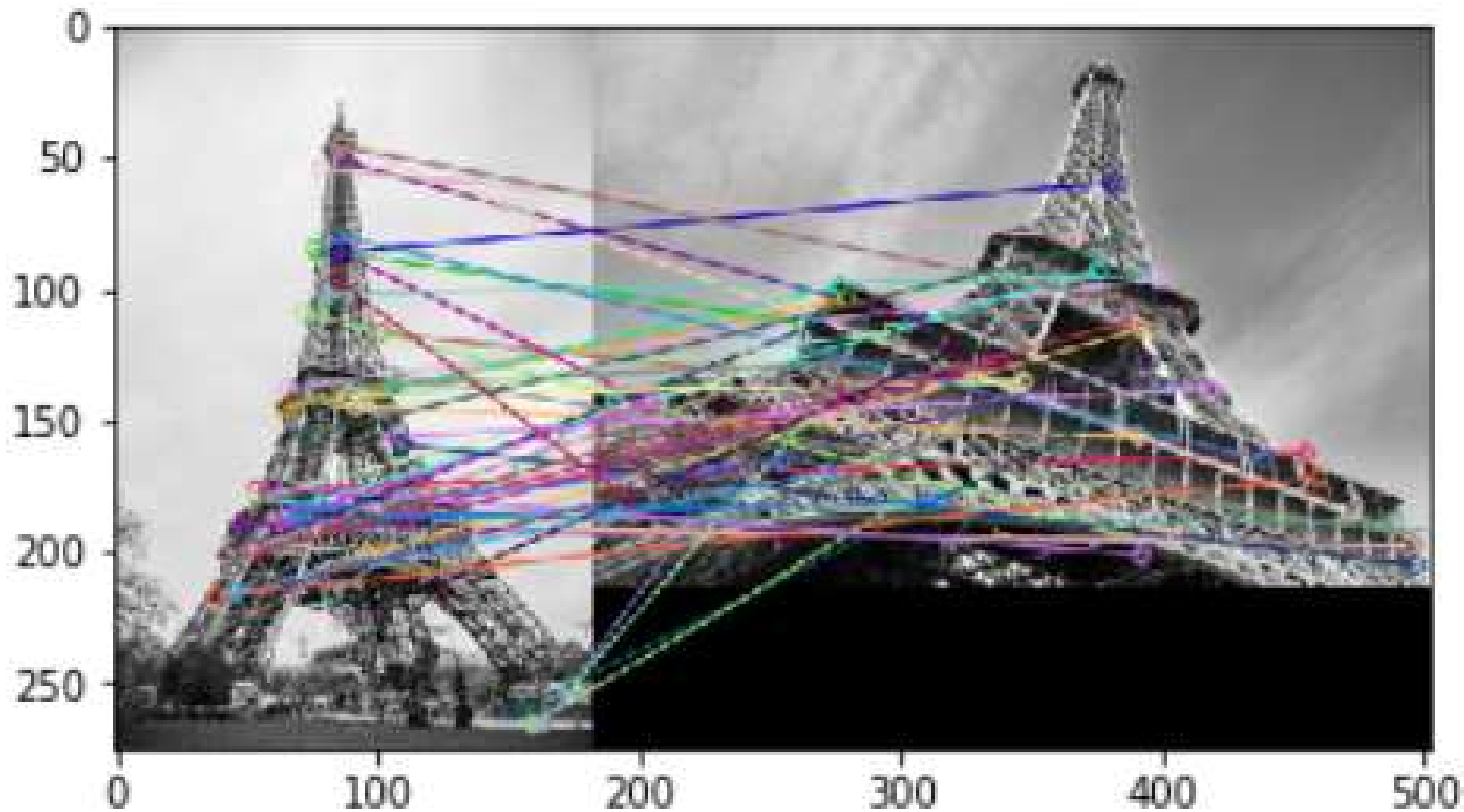
Image from: Jonas Hurreımann

SIFT Descriptor – Lighting changes

- Gains do not affect gradients
- Normalization to unit length removes contrast
- Saturation affects magnitudes much more than orientation
- Threshold gradient magnitudes to 0.2 and renormalize

Typical Usage

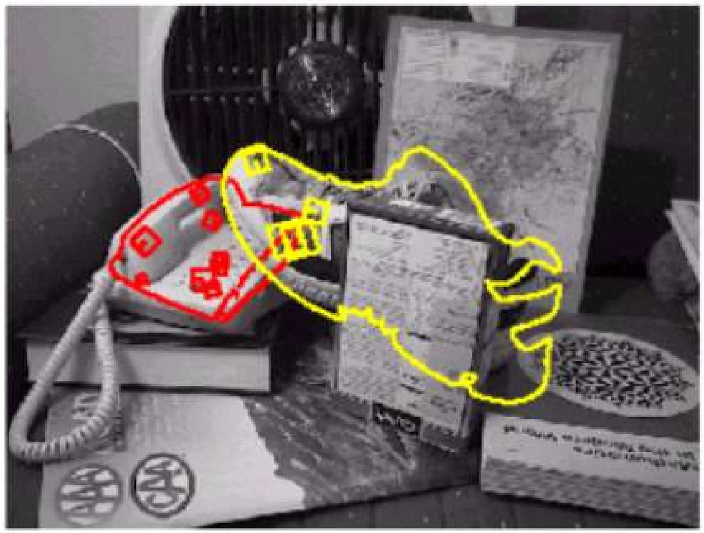
- For set of database images:
  1. Compute SIFT features
  2. Save descriptors to database
- For query image:
  1. Compute SIFT features
  2. For each descriptor:
     - Find a match
  3. Verify matches
     - Geometry
     - Hough transform

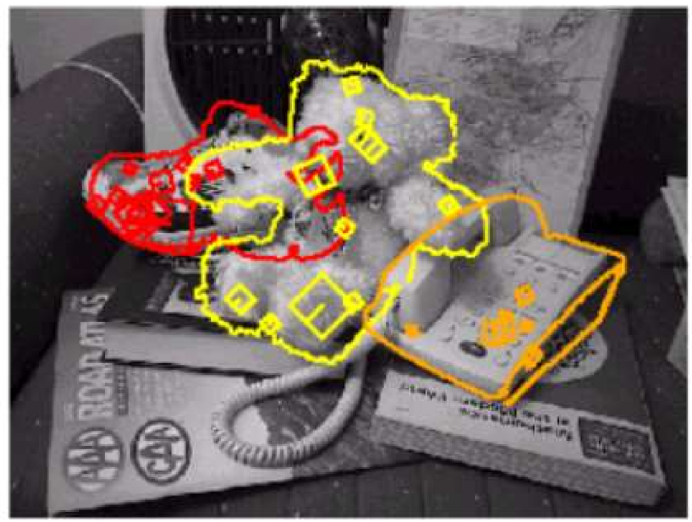

0
50
100
150
200
250
0
100
200
300
400
500

3D Object Recognition

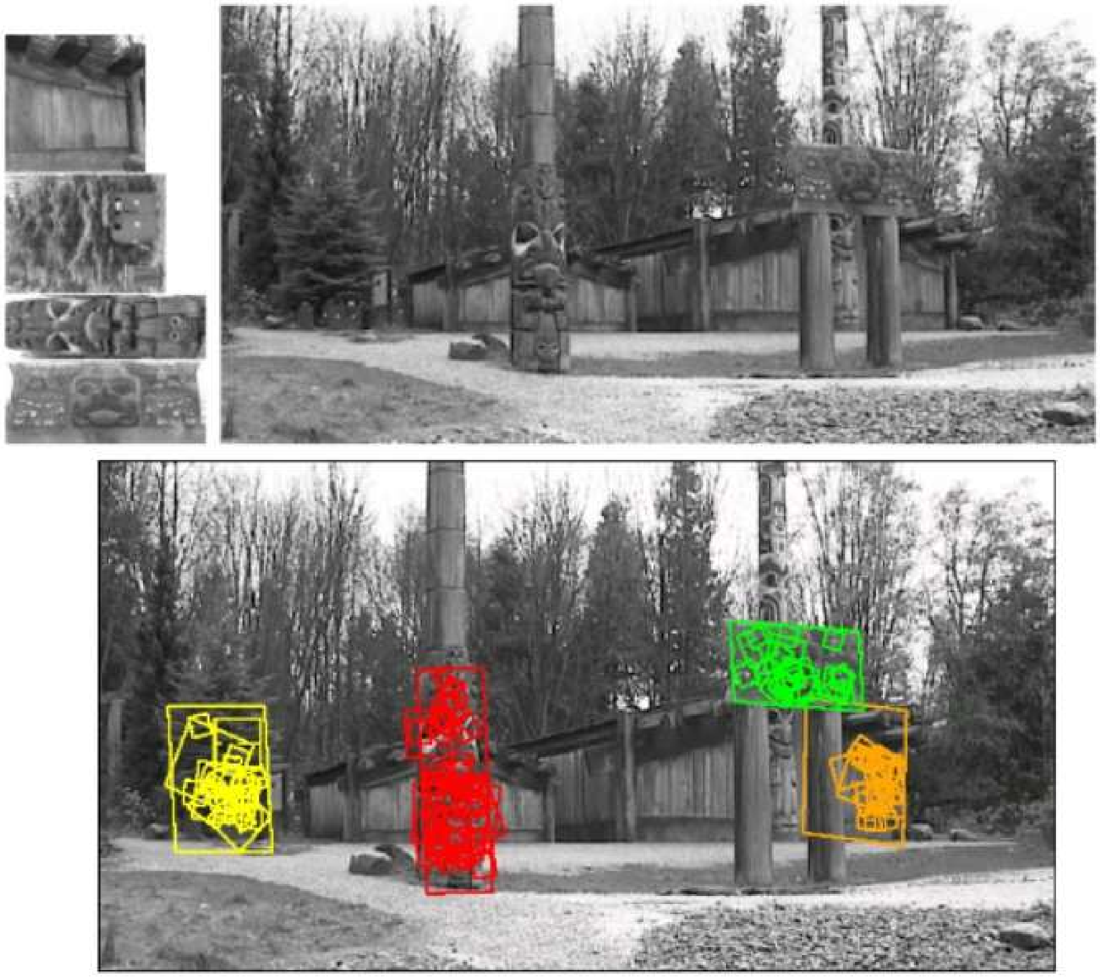
- Only 3 keys are needed for recognition, so extra keys provide robustness

Recognition under occlusion

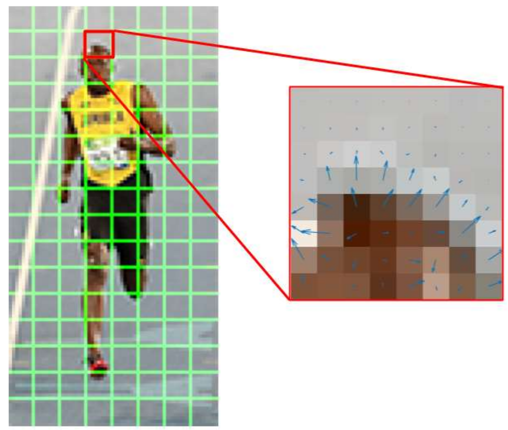




Location recognition



Examples

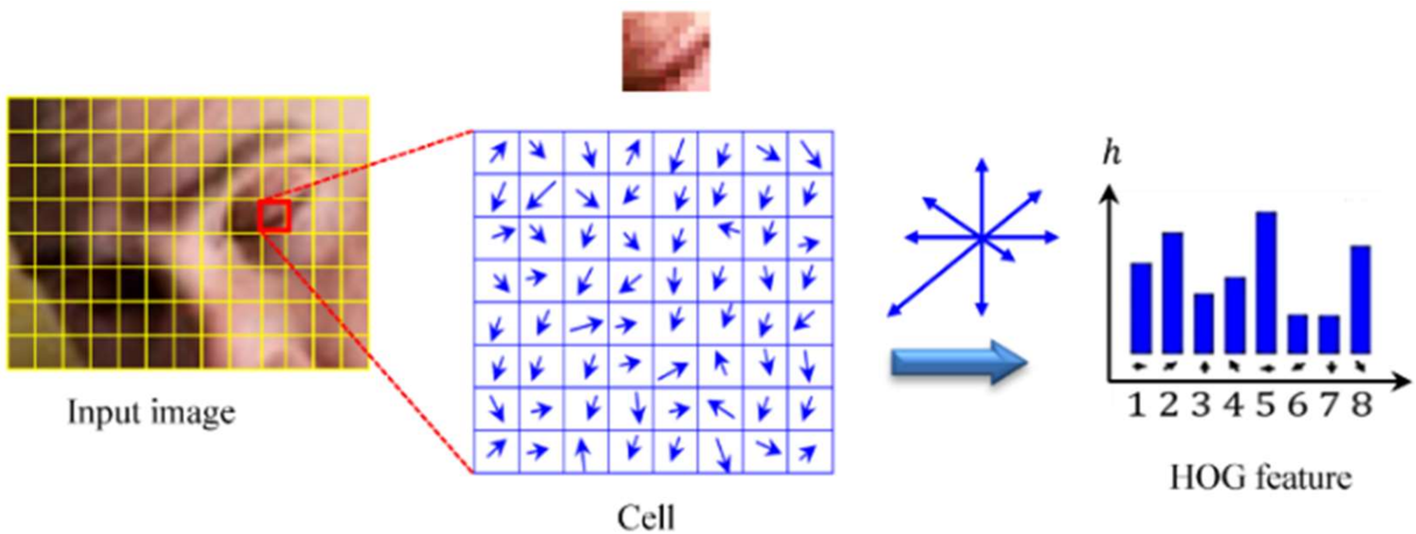


2	3	4	4	3	4	2	2
5	11	17	13	7	9	3	4
11	21	23	27	22	17	4	6
23	99	165	135	85	32	26	2
91	155	133	136	144	152	57	28
98	196	76	38	26	60	170	51
165	60	60	27	77	85	43	136
71	13	34	23	108	27	48	110

Gradient Magnitude

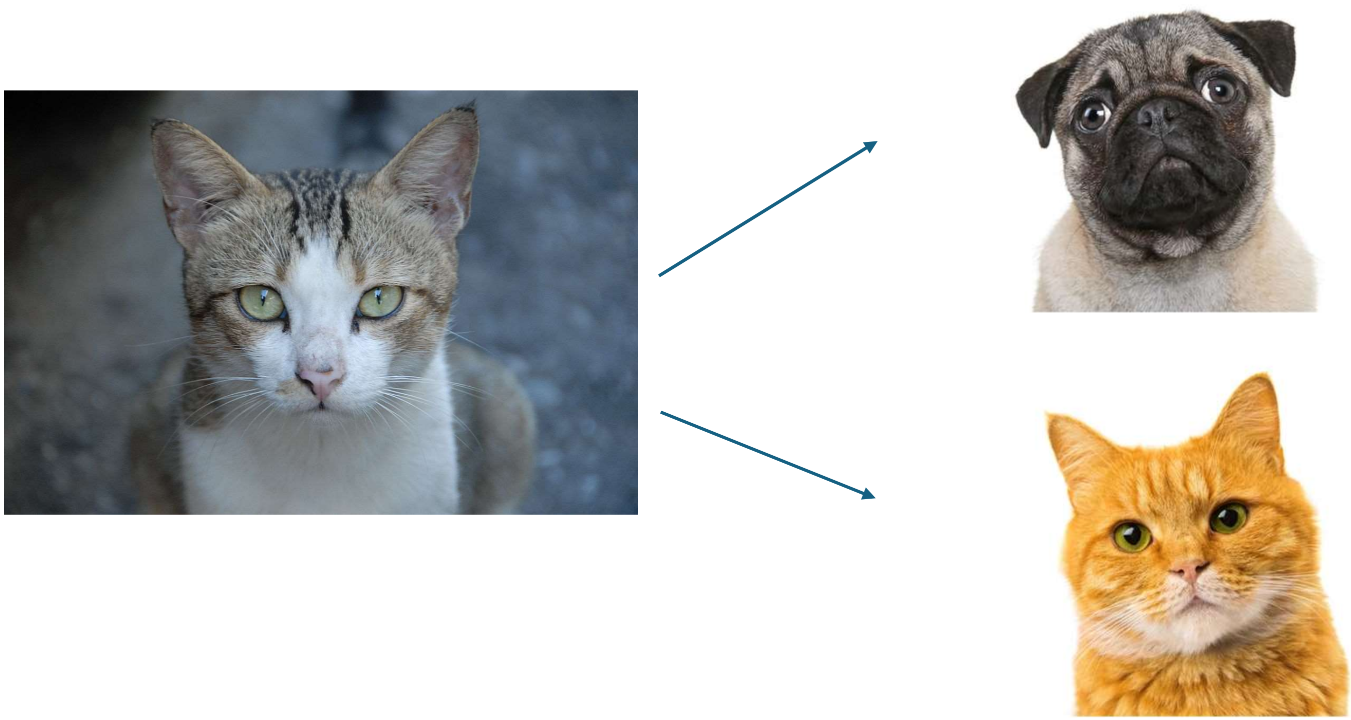
80	36	5	10	0	64	90	73
37	9	9	179	78	27	169	166
87	136	173	39	102	163	152	176
76	13	1	168	159	22	125	143
120	70	14	150	145	144	145	143
58	86	119	98	100	101	133	113
30	65	157	75	78	165	145	124
11	170	91	4	110	17	133	110

Gradient Direction



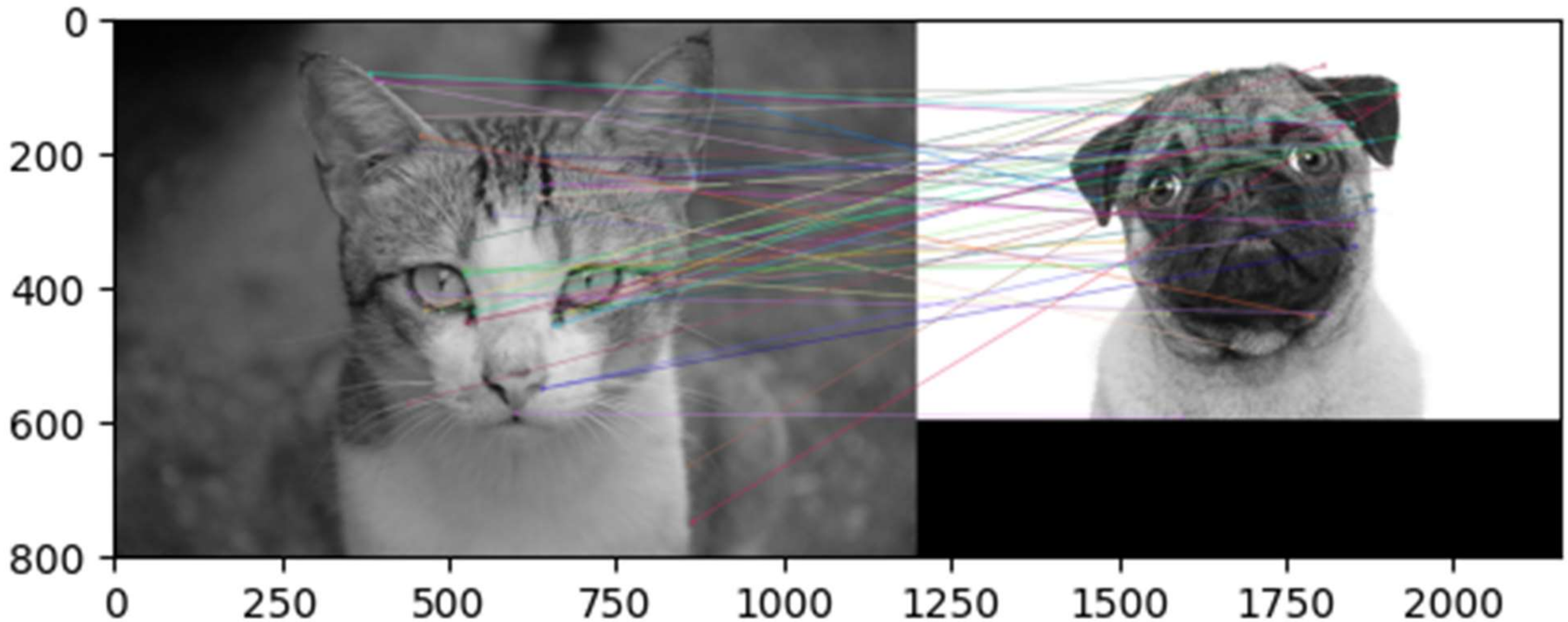
Cat vs dog

New Image Introduced



First, take the images

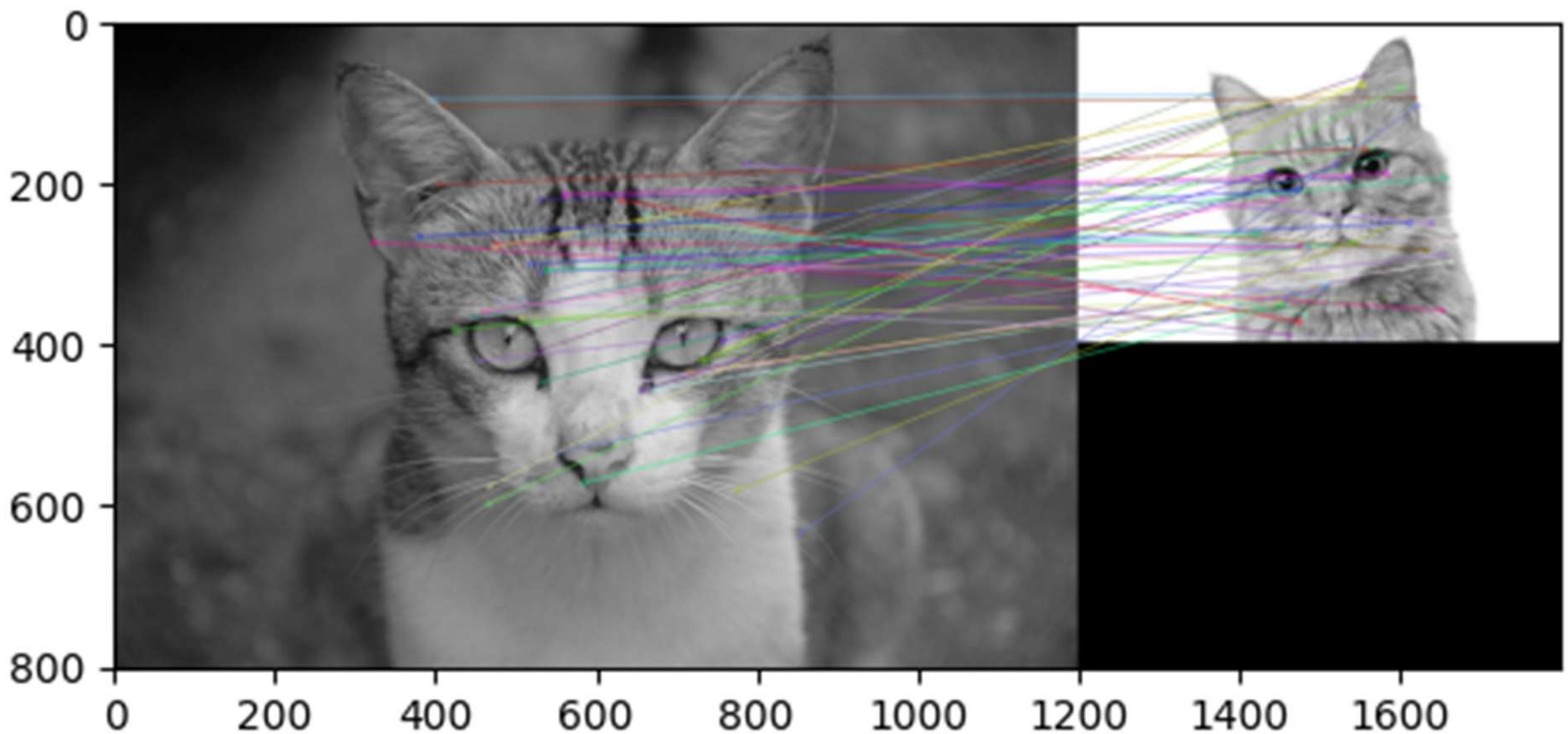
Perform Keypoint Comparison to Dog



1937 keypoints

1429 keypoints

Perform Keypoint Comparison to Cat



1937 keypoints

584 keypoints

Results

26.2%

73.8%

More keypoints (1429)
Fewer keypoint % matches

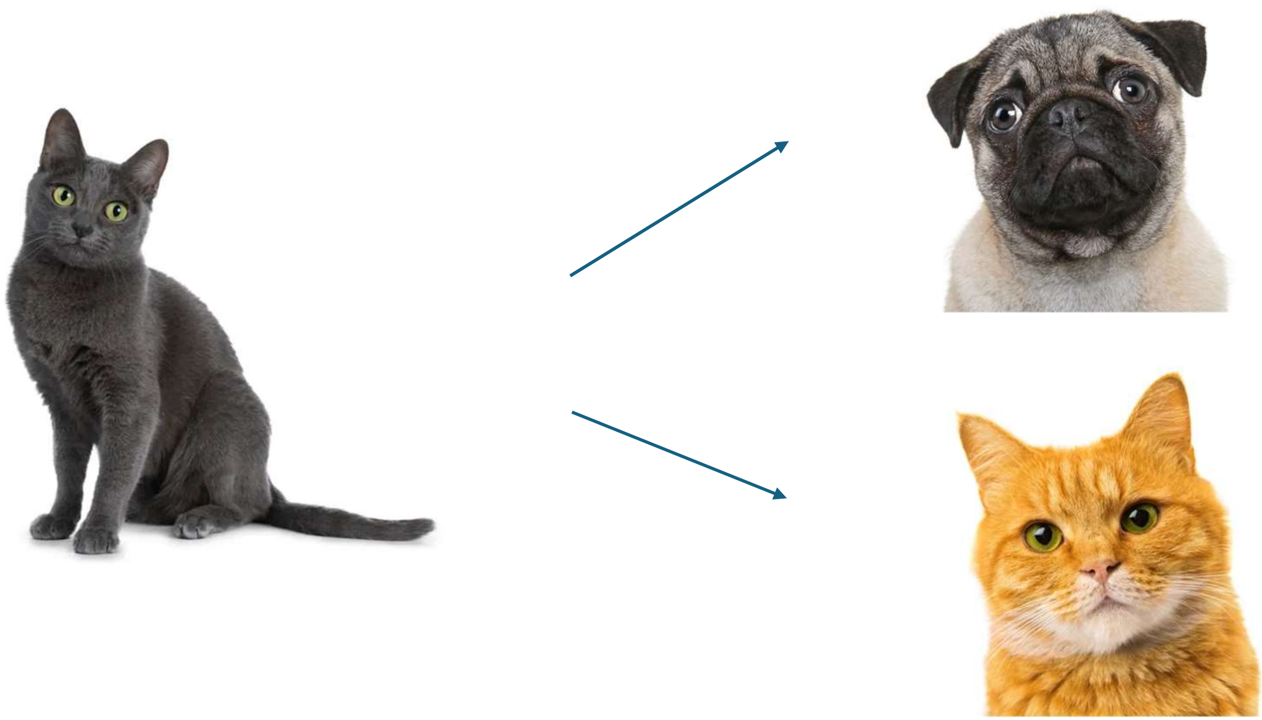
Keypoint: 0.308 match

Fewer keypoints (584)
Greater keypoint % matches

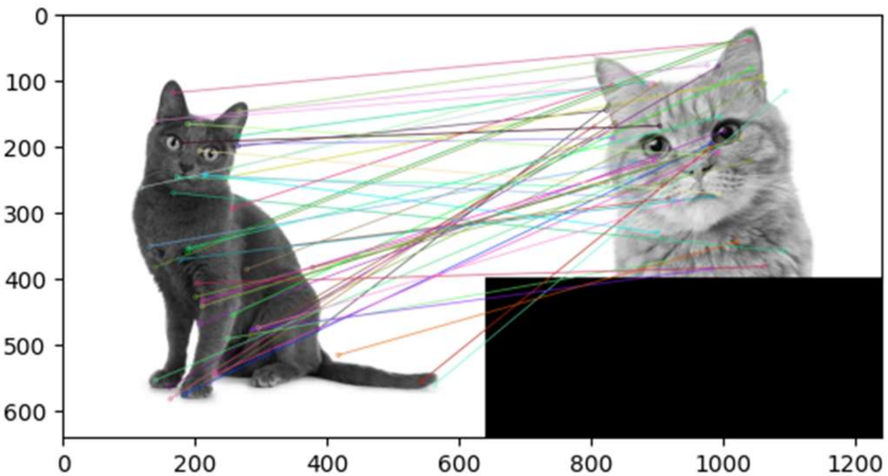
Keypoint: 0.869 match

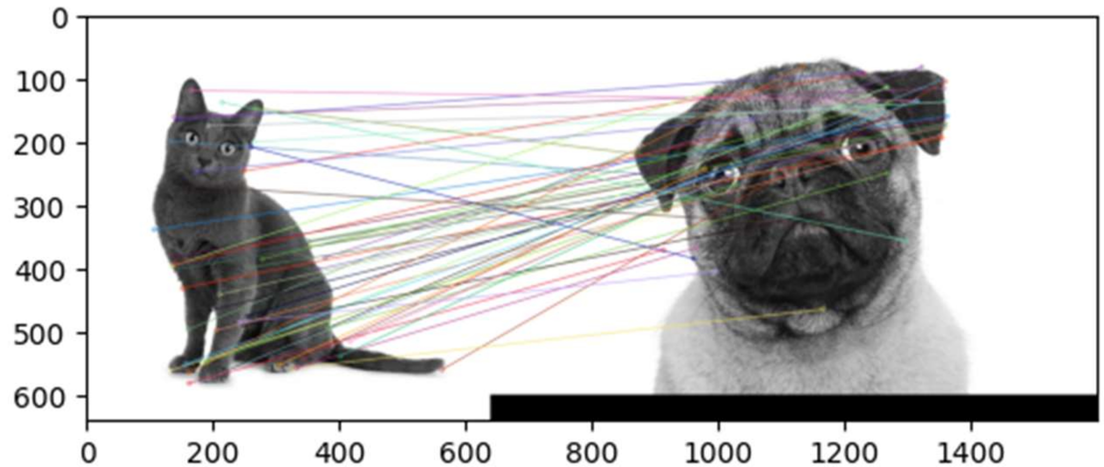
Probability: dog = 0.308 / (0.308 + 0.869) = 26.2%
Probability: cat = 0.869 / (0.308 + 0.869) = 73.8%

New Image Introduced



Perform Keypoint Comparison to Cat, Dog





358 keypoints 584 keypoints

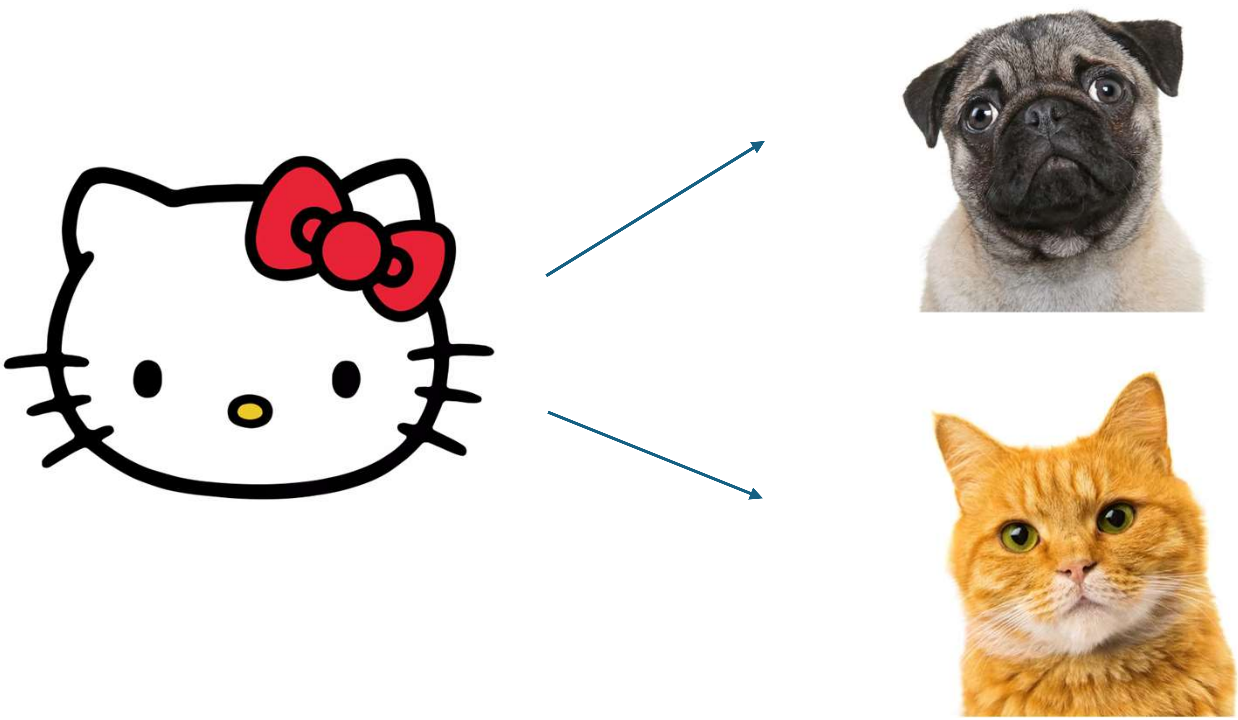
0.294 accuracy

358 keypoints 1429 keypoints

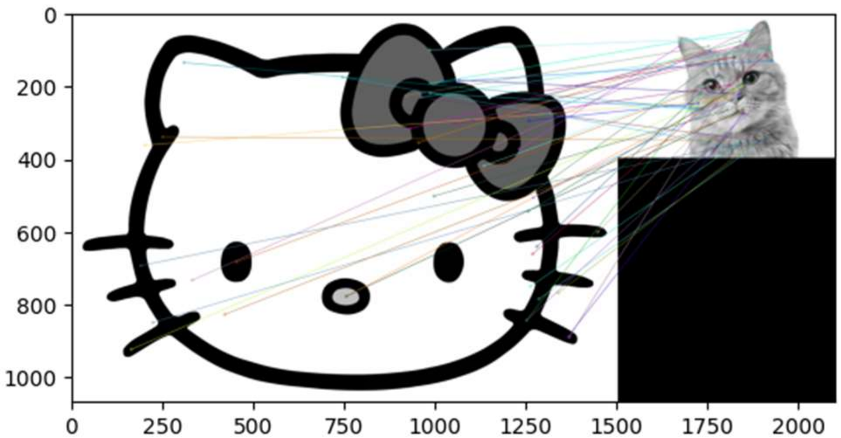
0.162 accuracy

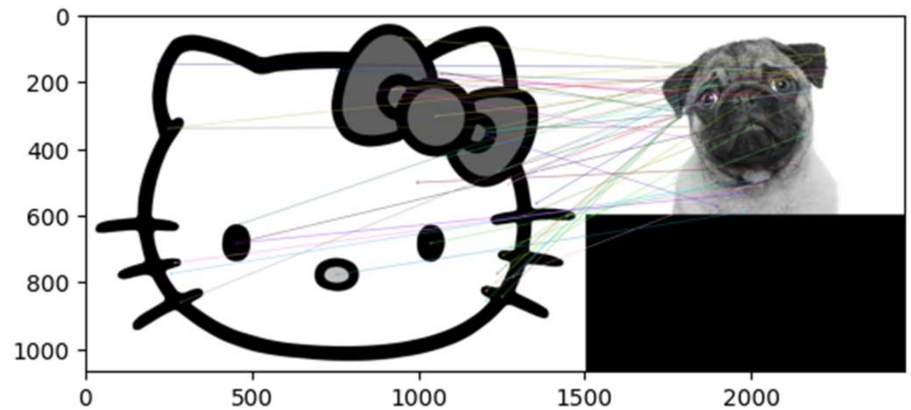
Decision: cat!

What about this image?



Perform Keypoint Comparison to Cat, Dog





306 keypoints

584 keypoints

0.563 accuracy

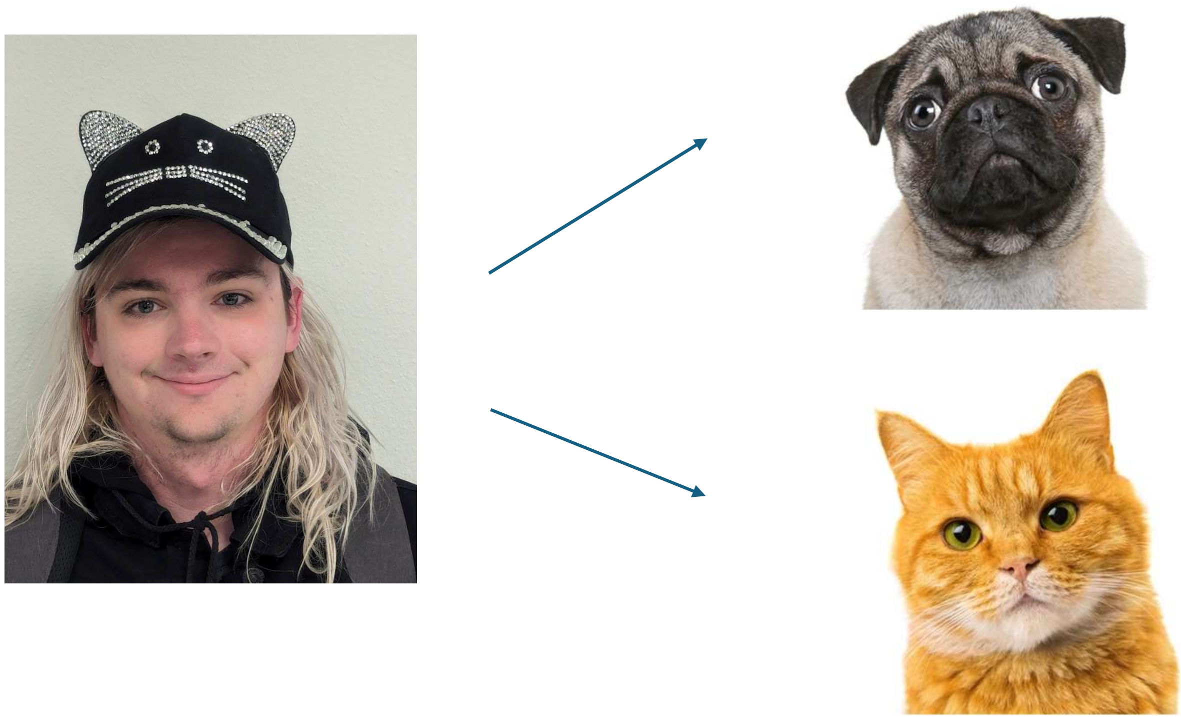
306 keypoints

1429 keypoints

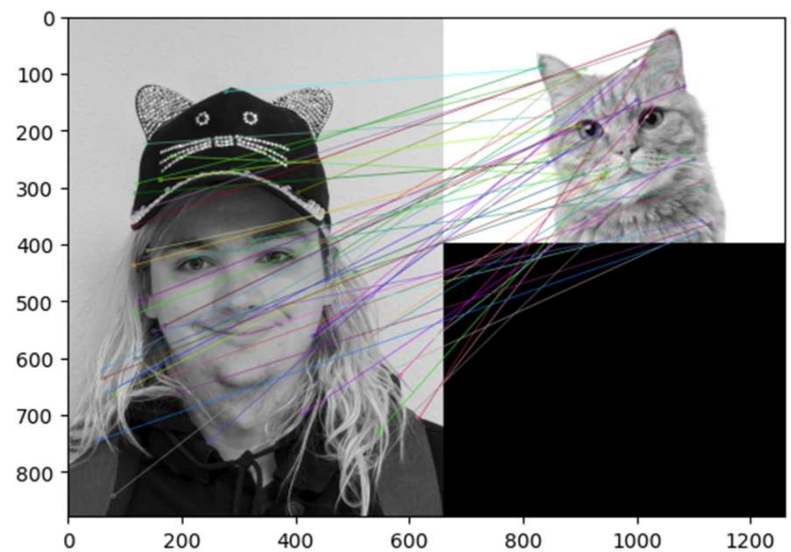
0.401 accuracy

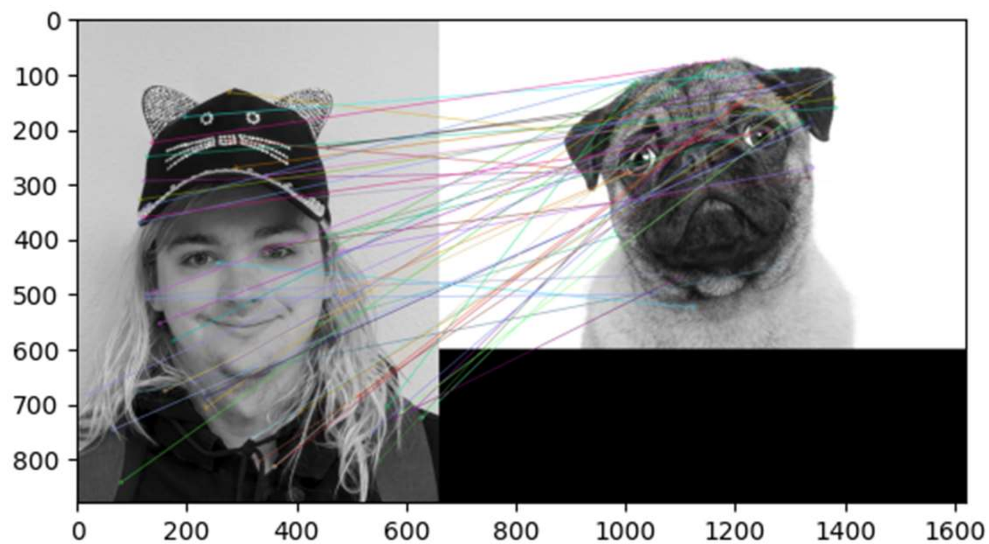
Decision: cat!

What about this image?



Perform Keypoint Comparison to Cat, Dog





1770 keypoints 584 keypoints

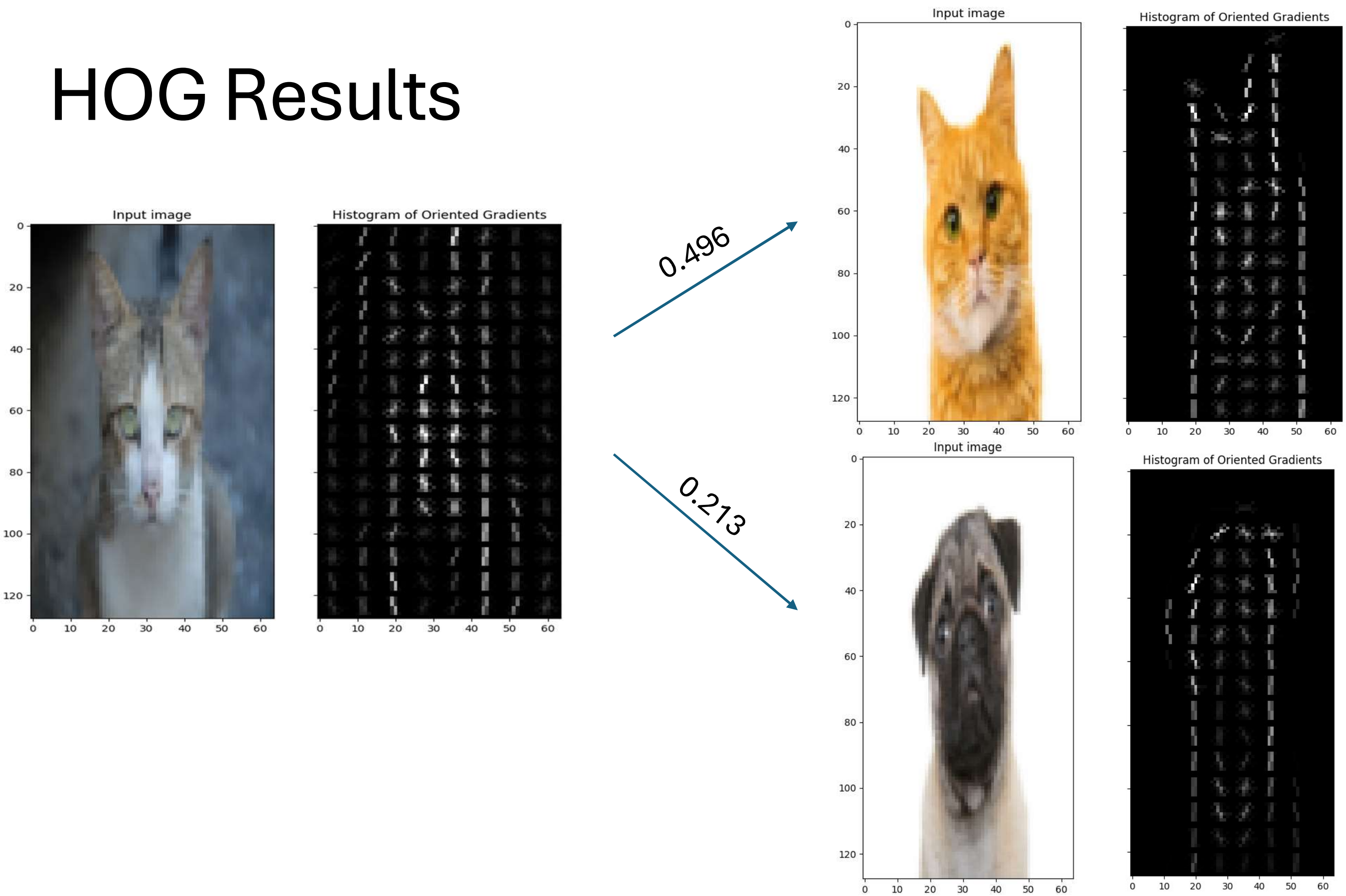
0.251 accuracy

1770 keypoints 1429 keypoints

0.203 accuracy

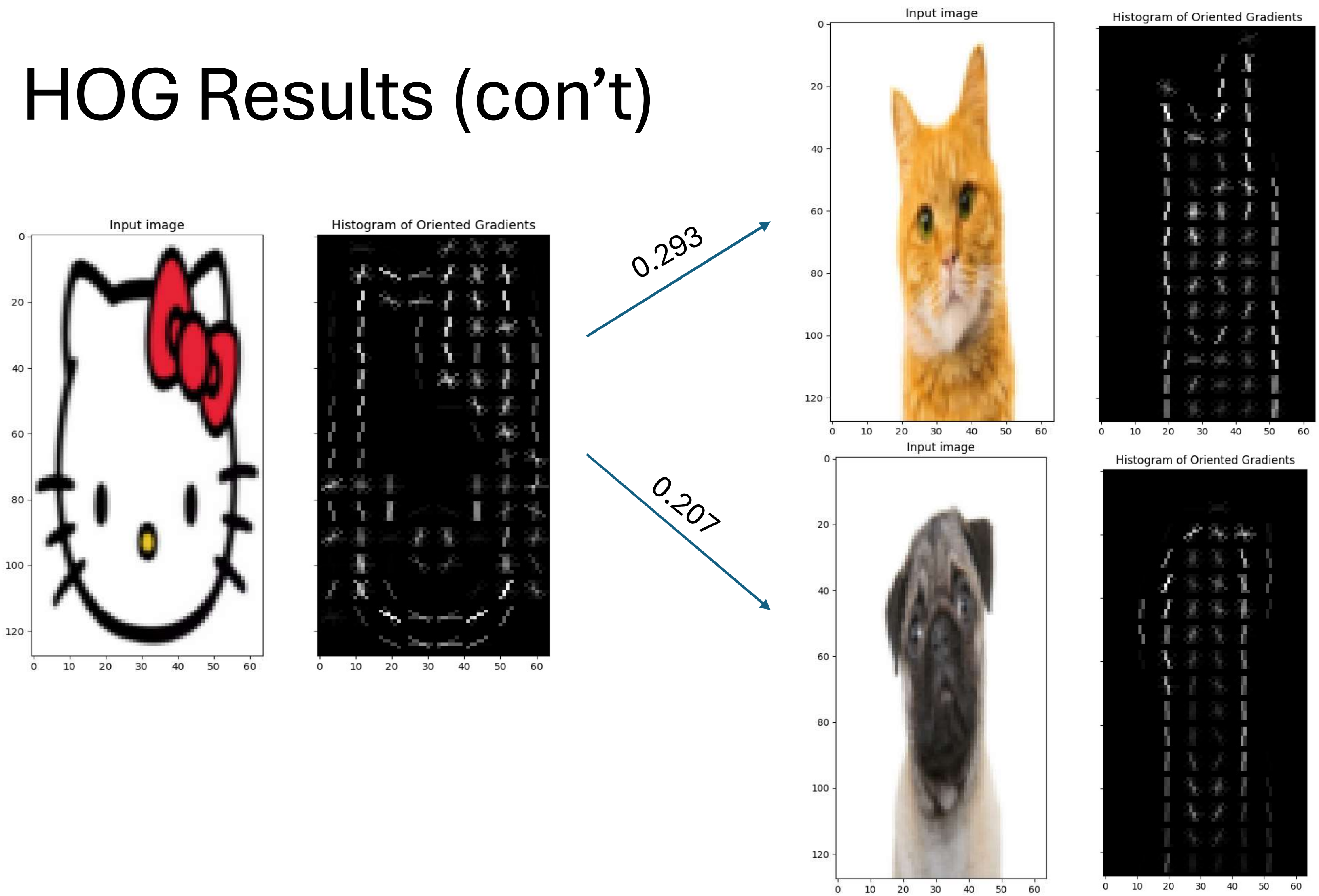
Decision: cat… or undetermined?

HOG Results

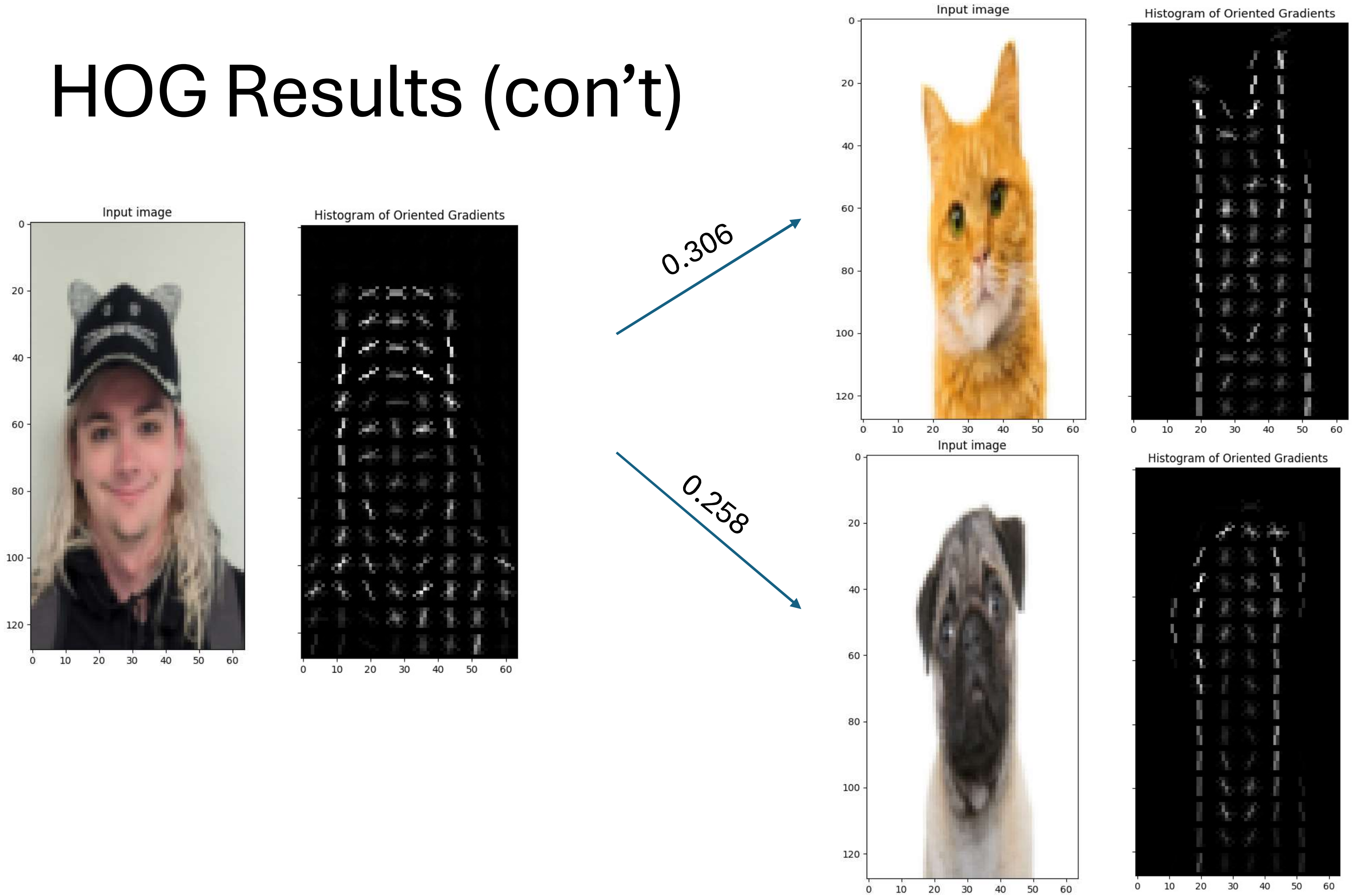


Input image
Histogram of Oriented Gradients
0.496
0.213
Input image
Histogram of Oriented Gradients
Input image
Histogram of Oriented Gradients

HOG Results (con't)



HOG Results (con't)



Conclusion: HOG

- HOG is based off feature descriptors, which extract the useful information and discard the unnecessary parts.

- HOG calculates the horizontal and vertical component of the gradient's magnitude and direction of each individual pixel. It then organizes the information into a 9-bin histogram to determine shifts in the data.

- Block normalization can be further utilized to make the model more optimal and less biased.

- HOGs can be used everywhere from the field of autonomous vehicles to the field of AR and VR (mostly anything involving image detection).

Conclusion: SIFT

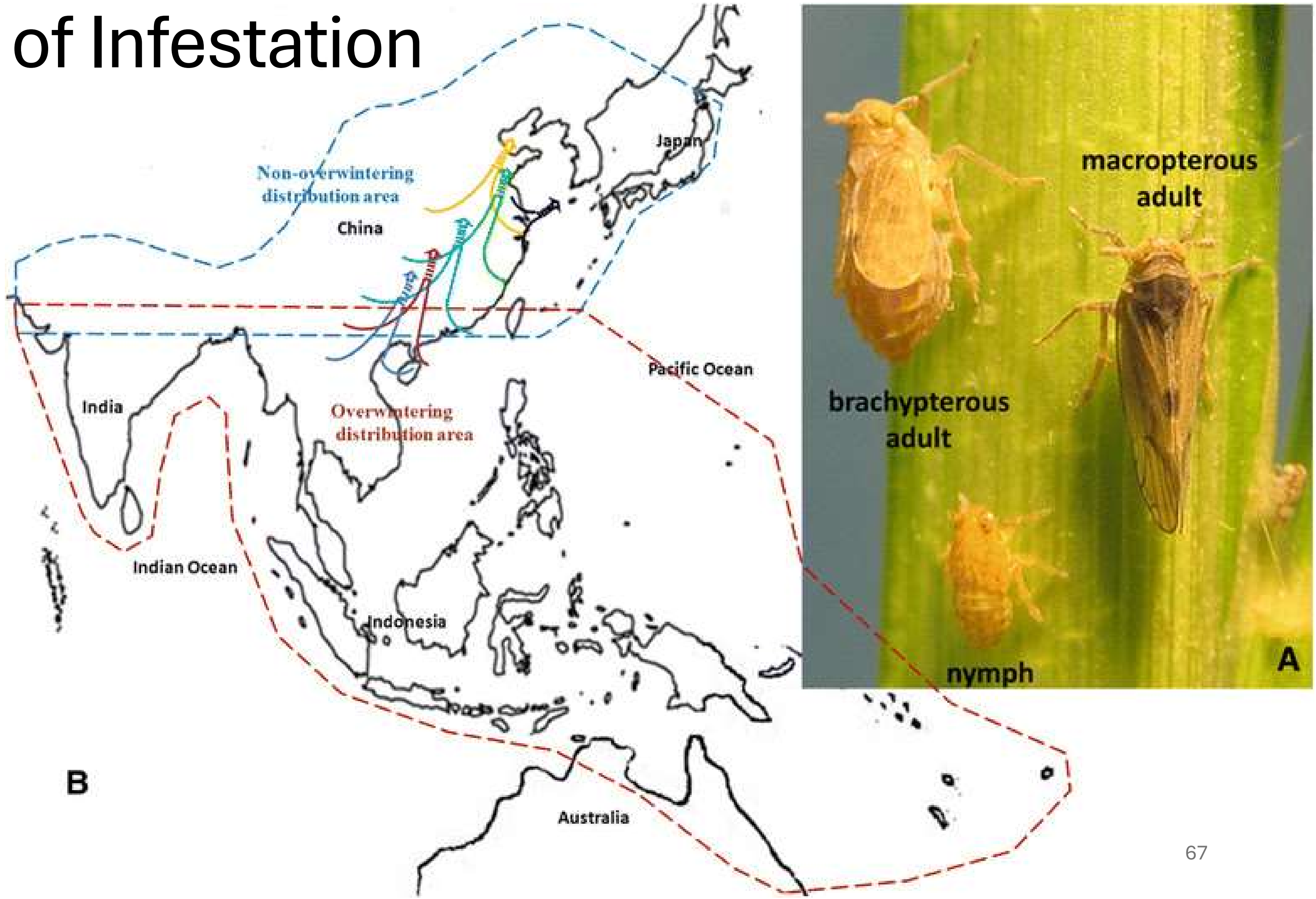
- Built on strong foundations
  - First principles (LoG and DoG)
  - Biological vision (Descriptor)
  - Empirical results
- Many heuristic optimizations
  - Rejection of bad points
  - Sub-pixel level fitting
  - Thresholds carefully chosen

Feature Extraction Method	Advantages	Disadvantages
Local appearance-based methods		
1. **BRIEF** (Binary Robust Independent Elementary Features)	• Very fast and efficient to compute	• Less accurate than other local appearance-based methods
2. **HOG** (Histogram of Oriented Gradients)	• Effective for capturing shape and texture information • Robust to changes in illumination and occlusion	• Computationally expensive
3. **LBP** (Local Binary Pattern)	• Simple and efficient to compute • Robust to changes in illumination and occlusion	• Not as effective for capturing shape information as HOG
Interest point-based methods		
1. **SIFT** (Scale-Invariant Feature Transform)	• Very accurate and robust to geometric transformations	• Computationally expensive
2. **SURF** (Speeded-Up Robust Features)	• Faster than SIFT while maintaining good accuracy	• Not as accurate as SIFT under some conditions
3. **ORB** (Oriented FAST and Rotated BRIEF)	• Very fast and efficient to compute	• Less accurate than other interest point-based methods

Brown Planthoppers (BPH)

- The brown planthopper (BPH), *Nilaparvata lugens (Stål),* is an insect pest of rice.
- Suck the sap from the rice phloem causing direct damage to rice plants and transmitting viral diseases.
- Responsible for significant damage to the rice yield as well as substantial economic losses for the rice-growing region that extends west to Pakistan, east to Japan, and south to Australia.
- Responsible for approximately 30% of rice crop yield losses potentially ruining the lives of many small farmers.

BPH Areas of Infestation



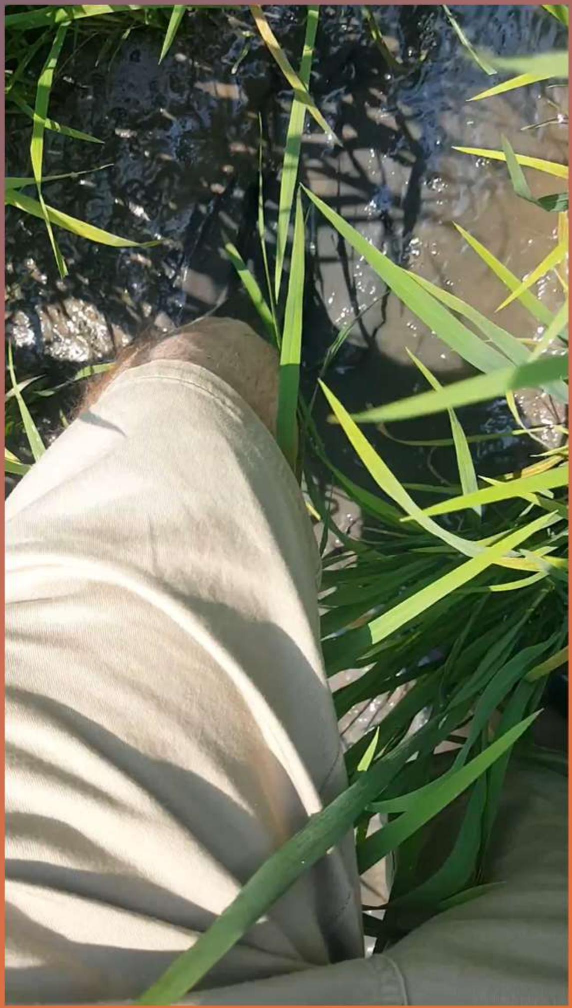
Three BPH forms

Nymph

Brachypterous

Macropterous


It is not all fun and games…





Why not just apply pesticides?

- have adverse environmental effects on the rice paddies and the soils downstream.
- kill the natural predators and parasitoids of BPH, disrupting the ecological balance.
- non-judicious use of insecticides has caused resistance and resurgence of BPH.
- use of pesticides is often cost-prohibitive.

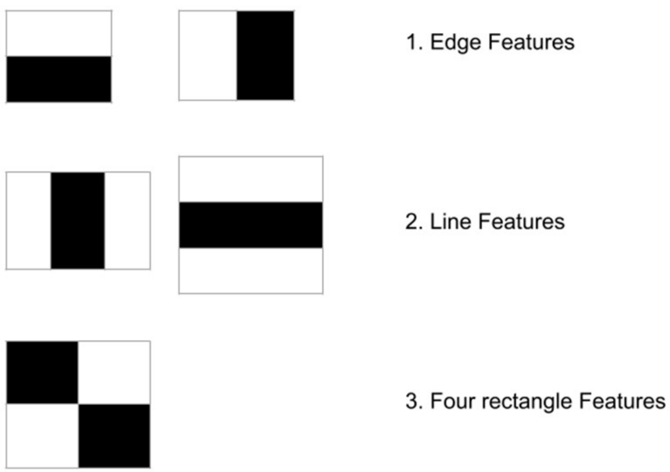
Beginning with BPH Images

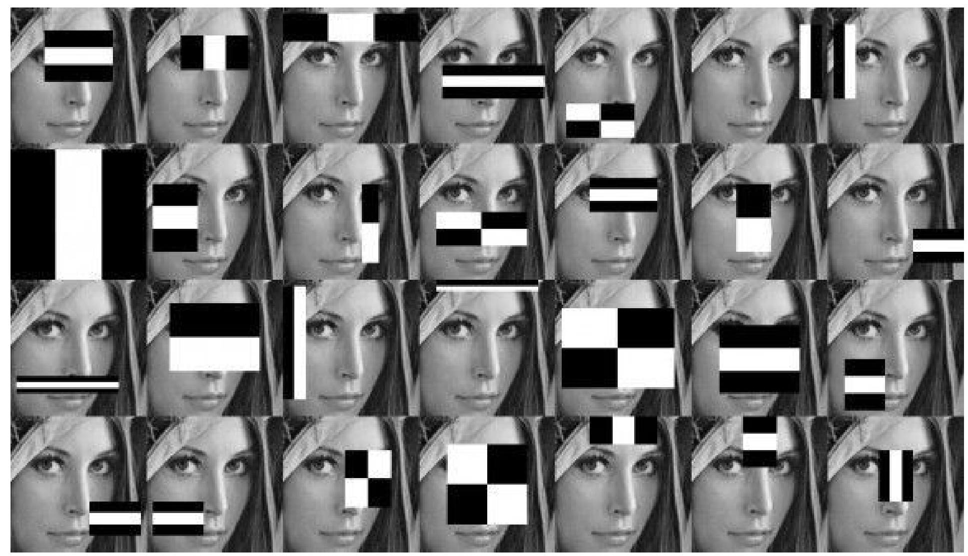
- a collection of 96 unique 1080x1920 pixel images of BPH on rice plants in a greenhouse
- images were taken under standard lighting conditions with a smartphone at a distance of 30-60 cm.
- contained various forms of BPH, averaging 24 BPH per plant.
- camera was rotated around the plant to capture BPH in situ.
- manually annotated by an entomologist to develop our ground truth. Lab setting, but actual BPH on rice plants

Applying AdaBoost

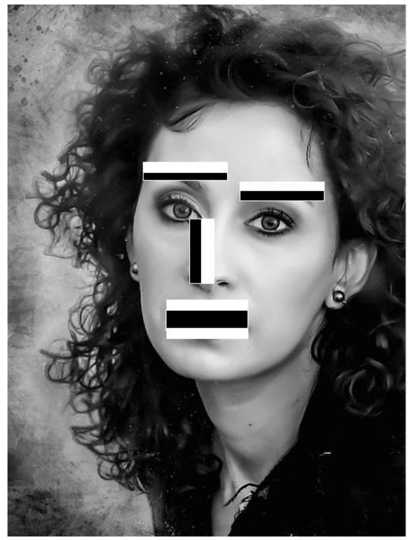
- AdaBoost: an algorithm for constructing a strong classifier from a linear combination of weak classifiers.
- Viola and Jones built a face detection framework using an integral image, AdaBoost, and a sequence of classifiers to provide accurate face detection.
- We have adapted this method to detect BPH using a series collection of positive planthopper examples and negative non-planthopper examples as training inputs.

Applying Haar-like Features to AdaBoost



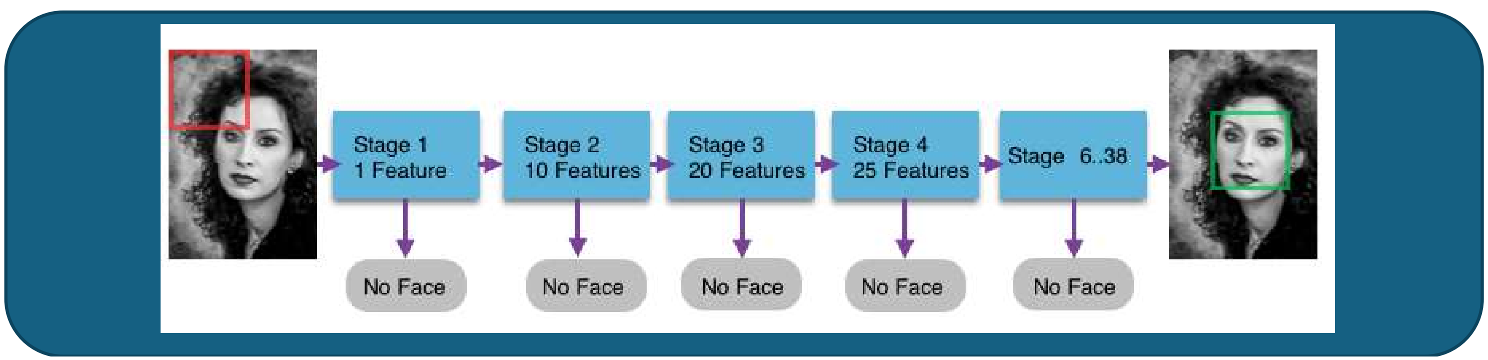


Applying Haar-like Features to AdaBoost



0	0	1	1
0	0	1	1
0	0	1	1
0	0	1	1

0.1	0.2	0.6	0.8
0.3	0.2	0.6	0.8
0.2	0.1	0.8	0.6
0.2	0.1	0.8	0.9

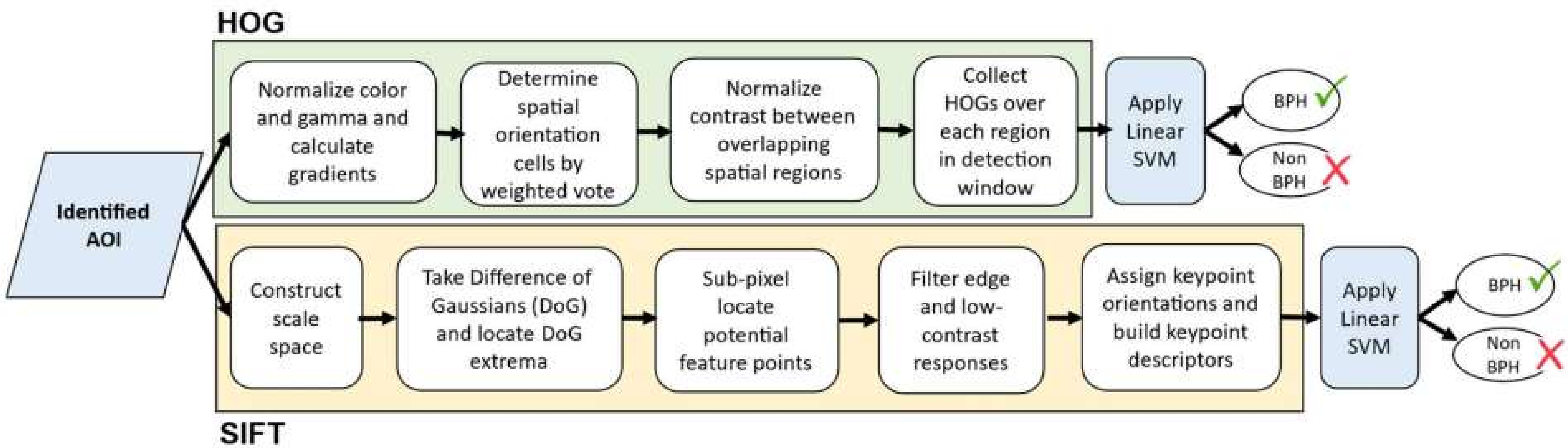


Identifying AOI using Adaboost and Haar Features

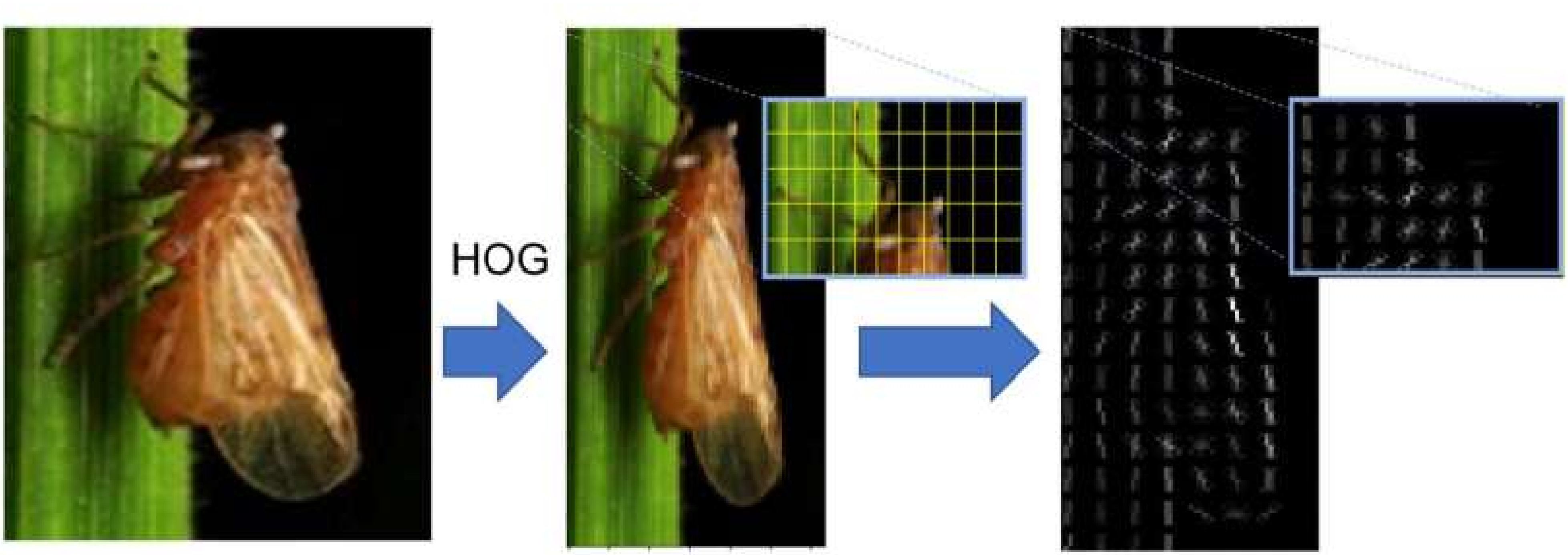
Finding Areas of Interest

Images of insects on a rice plant taken in the field (left) and the annotation (right). Red dots indicate BPH, blue dots indicate other species

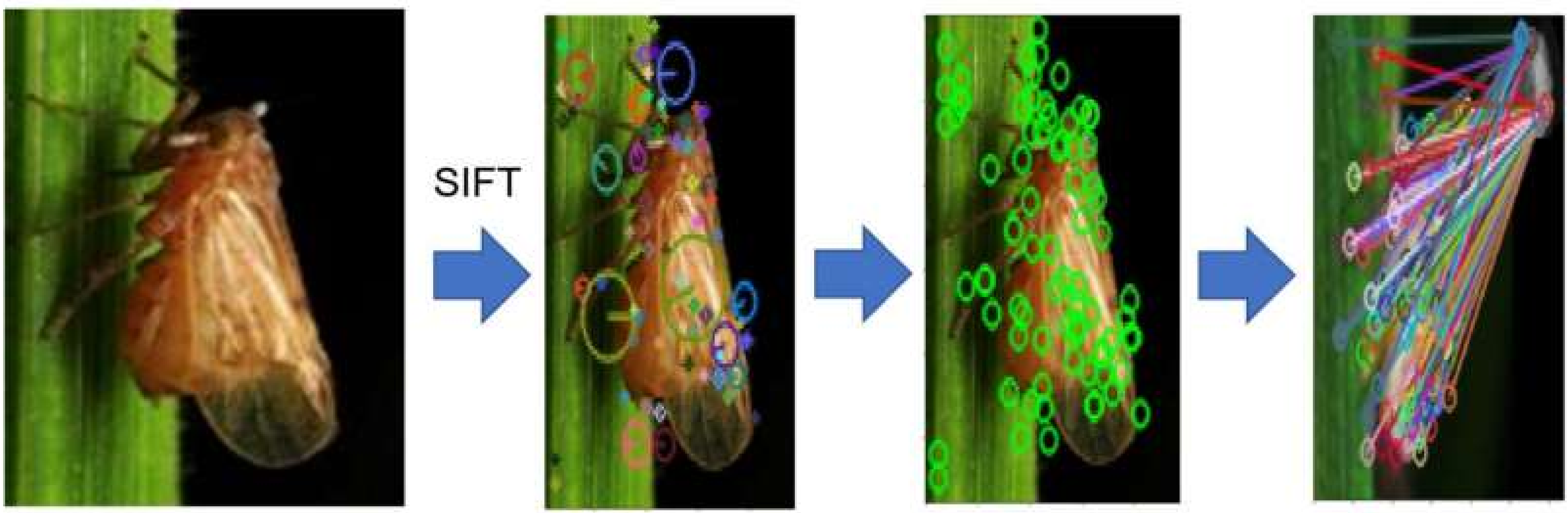
Process overview



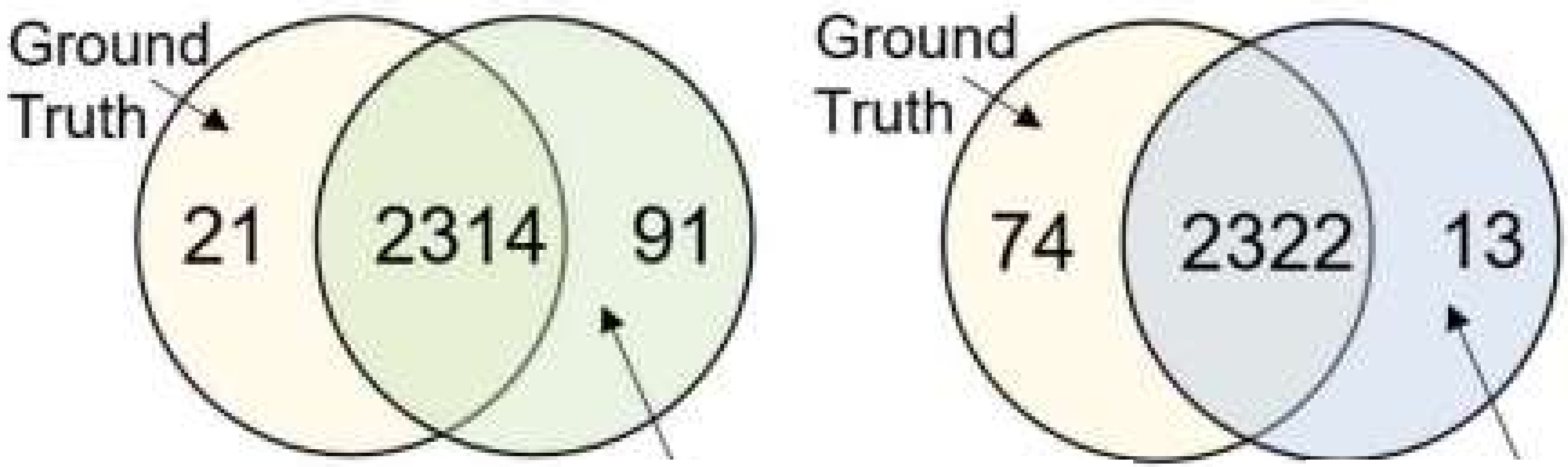
HOG (Histogram of Oriented Gradients)
Process overview



SIFT (Scale-Invariant Feature Transform) Process overview



Results – Accuracy by Technique



HOG-SVM
(more FN)

SIFT-SVM
(more FP)

Results by BPH Densities in Images

TABLE 1. IDENTIFICATION RESULTS OF BPH AT DIFFERENT DENSITY GRADES BY DETECTION LAYER

Detection Method	Detection rate (DR) and False Detection Rate (FDR) at different densities, (based on number of BPH per image)							
	<20		*21-40*		*>40*		*Unweighted Average*	
	DR	FDR	DR	FDR	DR	FDR	DR	FDR
Detection Layer 1 (Detecting HOG features)	89.6	10.4	91.2	8.8	92.8	7.2	91.2	8.8
Detection Layer 2 (SVM based using block size reduction on HOG)	90.3	9.7	92.9	7.1	95.3	4.7	93.3	6.7
Detection Layer 3 (Threshold judgment)	93.3	6.7	95.0	5.0	97.8	2.2	96.2	3.8
Detection Layer 1 (Detecting SIFT features)	90.5	9.5	92.4	7.6	93.2	6.8	92.0	8.0
Detection Layer 2 (SVM based using block size reduction on SIFT)	93.5	6.5	94.0	6.0	95.6	4.4	94.4	5.6
Detection Layer 3 (Threshold judgment)	95.3	4.7	96.8	3.2	98.5	1.5	96.9	3.1

Comparison with other BPH Identification Studies

TABLE 2. COMPARISON OF OUR RESULTS WITH OTHER BPH IDENTIFICATION STUDIES

Study	*Accuracy*	*Recall*	*Precision*
Qing et al. [3]	75.60	85.20	90.40
Nazri et al. [6]	95.00	-	-
Watcharabutsarakham et. al [7]	83.00	86.04	83.24
Wang et. al [8]	70.00	-	-
Harris and Trisyono [11]	82.65	76.80	73.85
He et al [12]	94.50	88.00	-
Our HOG-SVM model	95.38	99.10	96.22
Our SIFT-SVM model	96.38	99.44	96.91

Summary

- proposed a study that evaluates and compares two feature descriptors
- determined the Areas of Interest (AOI) using a modified version of Adaboost with Haar-like features.
- Fed the outputs of the two feature descriptors independently and pass them through a Support Vector Machine (SVM) classifier.
- evaluate the effectiveness of each approach on a set of images

this approach will help improve efforts to forecast infestations, resulting in better use of pesticides and fewer rice crop yields lost to BPH.