*E. Rosten and T. Drummond, “Machine learning for high-speed corner detection,” ECCV, 2006.

Features from Accelerated Segment Test*

ISL Lab Seminar

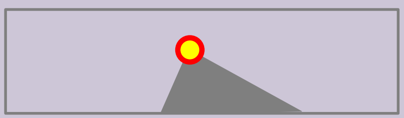
Jin-Hyung Kim

Contents

Features

A **feature** is a piece of information which is relevant for solving the computational task related to a certain application. Features may be specific structures in the image such as points, edges or objects. Features may also be the result of a general neighborhood operation or feature detection applied to the image.*

Corner



Intersection of two edges

- Harris
- Shi & Tomasi
- SUSAN
- FAST

Edge

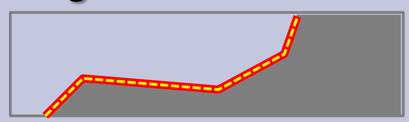
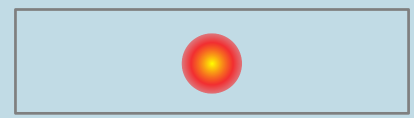


Image brightness changes sharply

- Canny
- Sobel
- Prewitt

Blob

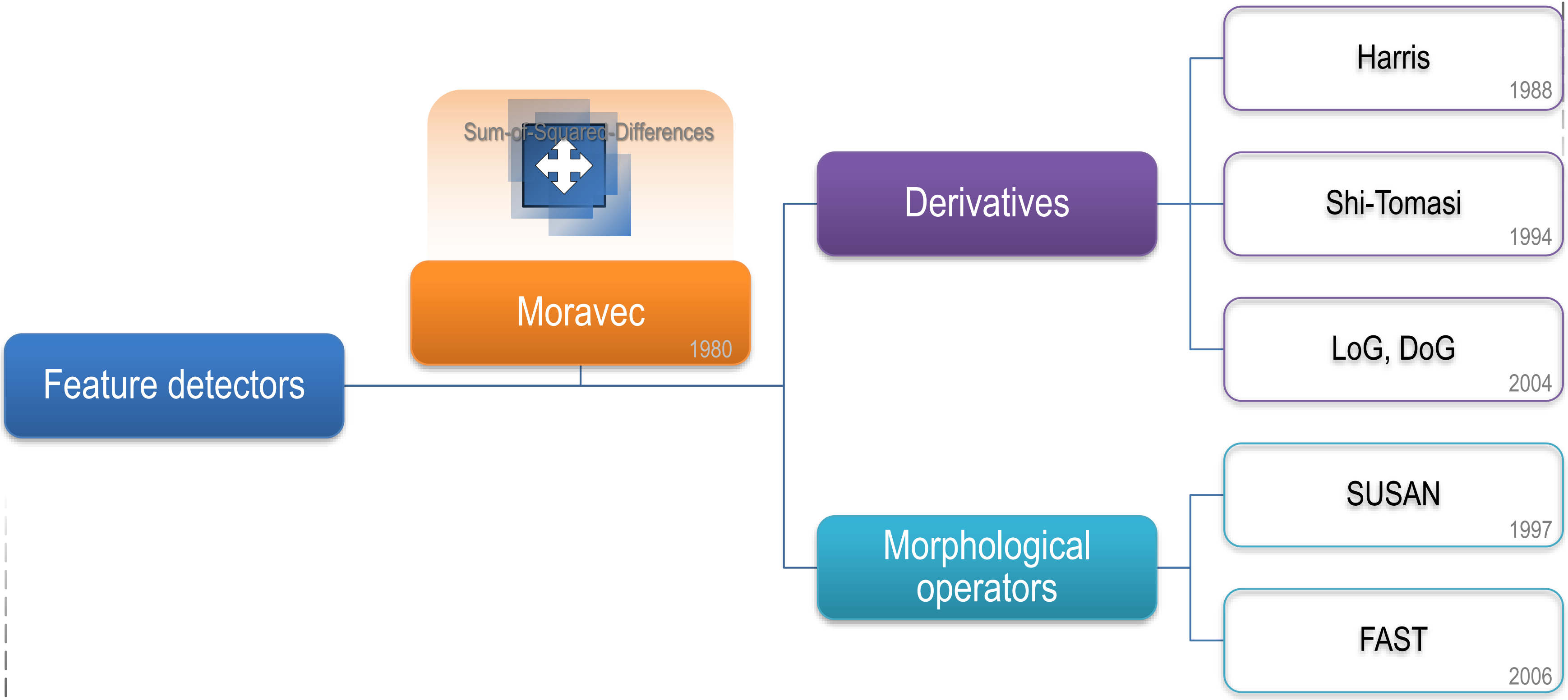


Regions that differ in properties compared to areas surrounding those regions

- Laplacian of Gaussian
- Difference of Gaussian

*http://en.wikipedia.org/wiki/Feature_(computer_vision)

Feature detectors



Harris, Shi-Tomasi

Harris corner detector

- Harris build an approximation to the second derivative of the SSD w.r.t the shift.

$$\mathbf{H} = \begin{bmatrix} \langle I_x^2 \rangle & \langle I_x I_y \rangle \\ \langle I_x I_y \rangle & \langle I_y^2 \rangle \end{bmatrix}$$

  - Define the corner response : $C = |\mathbf{H}| - k(trace\,\mathbf{H})^2$

Shi-Tomasi : Good Feature To Track

- Based on the assumption of affine image deformation, a mathematical analysis led Shi & Tomasi conclude that it is better to use the smallest eigen value of $\mathbf{H}$ as the corner strength function

$$C = \min(\lambda_1, \lambda_2)$$

SUSAN

- **Smallest Univalue Segment Assimilating Nucleus**
  - Assumes that a corner resembles a blurred wedge, and finds the characteristics of the wedge(amplitude, angle, blur) by fitting it to the local image.
  - Calculating the corner strength
    - ➢ Computes self similarity by looking at the proportion of pixels inside a disc whose intensity is within some threshold of the center(nucleus) value.
      1. Place a circular mask around the pixel(the nucleus)
      2. Calculate the number of pixels within the circular mask which have similar brightness to the nucleus(USAN)

$$n(M) = \sum_{\vec{m} \in M} c(\vec{m}) \quad c(\vec{m}) = e^{-\left(\frac{I(\vec{m}) - I(\vec{m}_0)}{t}\right)^6}$$

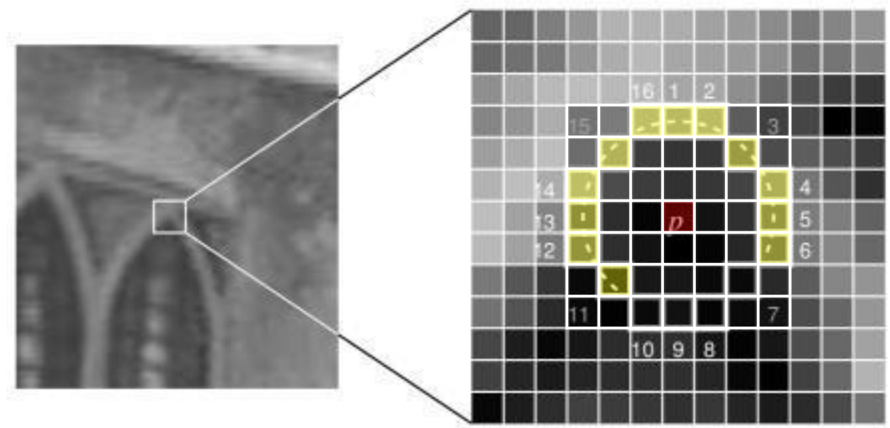
      3. Subtract the USAN size from the geometric threshold to produce a corner strength image

$$R(M) = \begin{cases} g - n(M) & if \quad n(M) < g \\ 0 & otherwise, \end{cases}$$

      4. Test for false positives by finding the USAN's centroid and its contiguity

FAST

- **The Segment-Test algorithm**
  - If $\geq N$ contiguous pixels in a Bresenham circle of radius r around a center pixel p are all brighter than p by some threshold or all darker than p by some threshold, then there is a feature at p



r=3, N=12

FAST

- **Features from Accelerated Segment Test**

Test pixels 1 & 9

If($p_0 \doteqdot p_1$ && $p_0 \doteqdot p_9$), than p_0 is not a feature

Test pixels 5 &13

If($p_0 \doteqdot p_5$ && $p_0 \doteqdot p_{13}$), than p_0 is not a feature

Segment-Test

If $<N(12)$ contiguous pixels are differ from p_0, than p_0 is not a feature

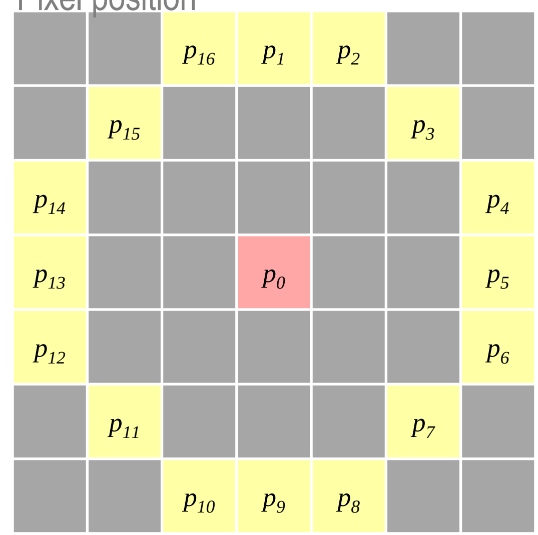
p_0 is a feature

Computationally efficient

- For $N=12$, at least 12 pixels to be tested to tell if p is a feature, but only 2 tests may be required to tell that p is not a feature.

- Problems
  ① $N=12$ is not the best choice
  ② The ordering of questions is not optimal (Machine learning)
  ③ Multiple features are detected adjacent to one another (Non-maximum suppression)

Pixel position



Machine learning a corner detector

1. Select a set of images for training
2. Run FAST algorithm in every images to find feature points
3. For every feature point, store the 16 pixels around it as a vector. Do it for all the images to get feature vector $\mathbf{P}$
   - $\mathbf{P}$: the set of all pixels in all training images
4. Depending on the states, the feature vector $\mathbf{P}$ is subdivided into 3 subsets, $\mathbf{P}_d$, $\mathbf{P}_s$, $\mathbf{P}_b$

$$S_{p\to x} = \begin{cases} d, & I_{p\to x} \le I_p - t & \text{(darker)} \\ s, & I_p - t < I_{p\to x} < I_p + t & \text{(similar)} \\ b, & I_p + t \le I_{p\to x} & \text{(brighter)} \end{cases} \qquad \begin{array}{l} p \in \mathbf{P} \\ x \in \{1..16\} \end{array}$$

1. Define a new boolean variable, K_p, which is true if $\mathbf{P}$ is a corner and false otherwise
2. Use the **ID3** algorithm(decision tree classifier) to query each subset using the variable K_p for the knowledge about the true class. It selects the x which yields the most information about whether the candidate pixel is a corner, measured by the entropy of K_p
   - The entropy of K for the set $\mathbf{P}$ is: $H(\mathbf{P}) = (c+\bar{c})\log_2(c+\bar{c}) - c\log_2 c - \bar{c}\log_2 \bar{c}$

$$\begin{array}{lll} \text{where} & c = \left|\{p \mid K_p \text{ is true}\}\right| & \text{(number of corners)} \\ \text{and} & \bar{c} = \left|\{p \mid K_p \text{ is false}\}\right| & \text{(number of non-corners)} \end{array}$$

3. This is recursively applied to all the subsets until its entropy is zero
4. The decision tree so created is used for fast detection in other images

Ex) a total of **4235 lines** of code are generated for $N=9$

For $N=9$ and $r=3$, only **2.26 questions** are required on average to classify a pixel

Non-maximal suppression

- **Score function V**
  - Since the segment test does not compute a corner response function, a score function is required.
  - Several intuitive definitions for V:

    ① ~~The maximum value of n for which p is still a corner~~

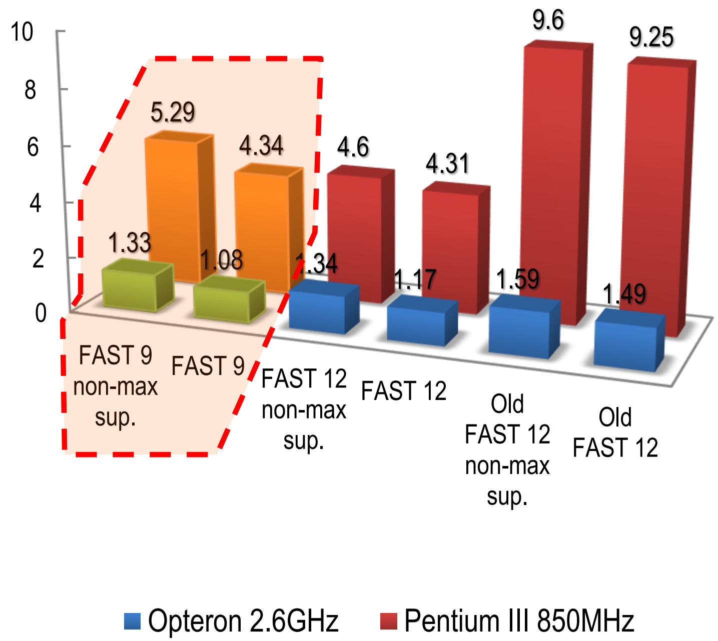
    ② ~~The maximum value of t for which p is still a corner~~

    ③ The SAD btw. the pixels in the contiguous arc and the center pixel

$$V = \max\left(\sum_{x \in S_{bright}} \left| I_{p \to x} - I_p \right| - t,\ \sum_{x \in S_{dark}} \left| I_{p \to x} - I_p \right| - t \right)$$

$$S_{bright} = \{ x \mid I_{p \to x} \geq I_p + t \}$$
$$S_{dark} = \{ x \mid I_{p \to x} < I_p - t \}$$

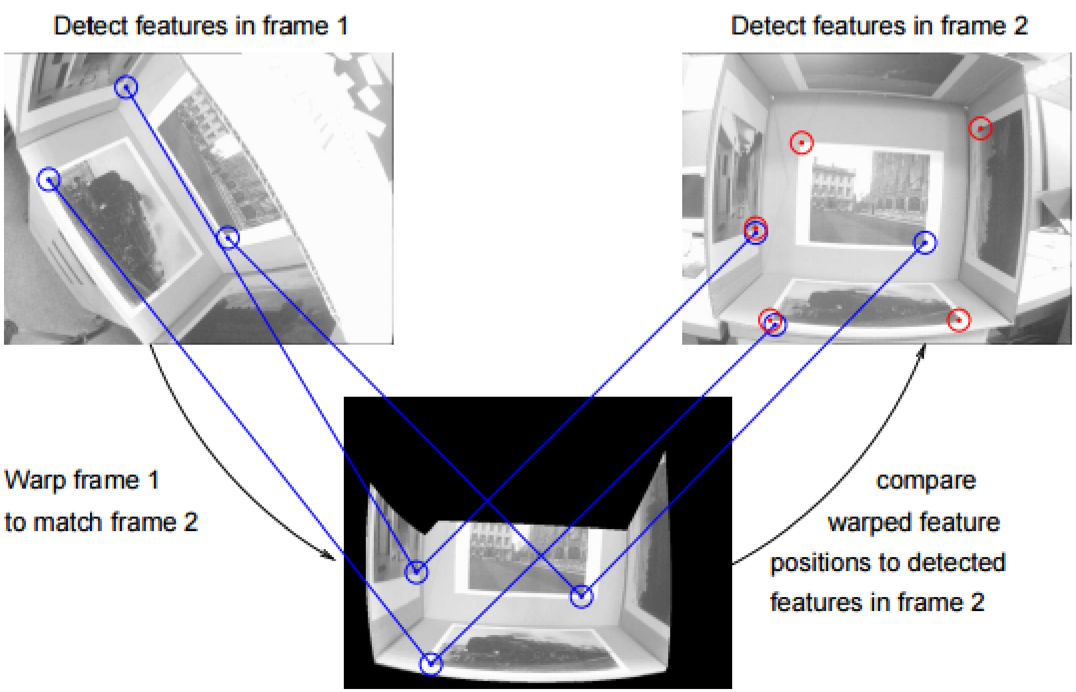
Timing results



768 × 288 PAL image set (ms)

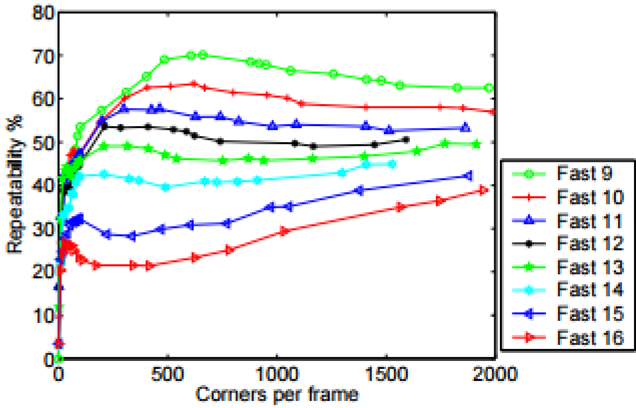
Detector	Opteron 2.6GHz	Pentium III 850MHz
FAST 9 non-max sup.	1.33	5.29
FAST 9	1.08	4.34
FAST 12 non-max sup.	1.34	4.60
FAST 12	1.17	4.31
Old FAST 12 non-max sup.	1.59	9.60
Old FAST 12	1.49	9.25
Harris	24.0	166
DoG	60.1	345
SUSAN	7.58	27.5

Repeatability evaluation

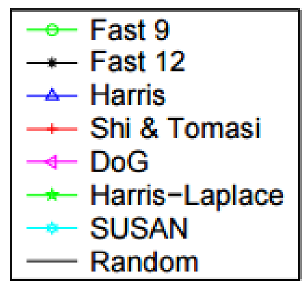


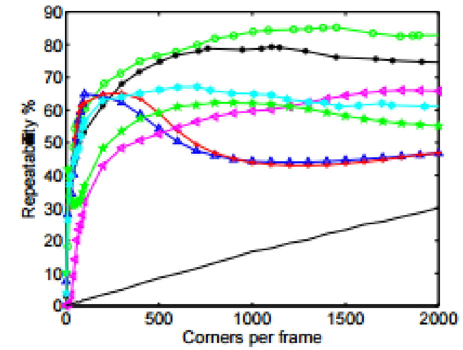
Repeatability is measured as the percentage of features detected from view 1 which are also detected in view 2

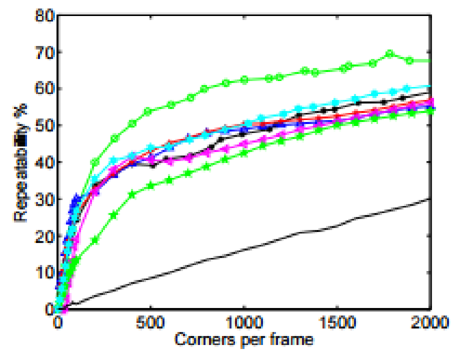
Repeatability evaluation

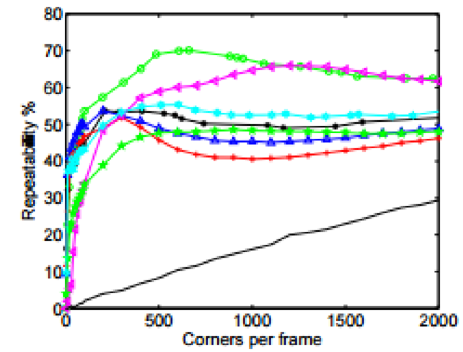


FAST 9 is the FAST detector

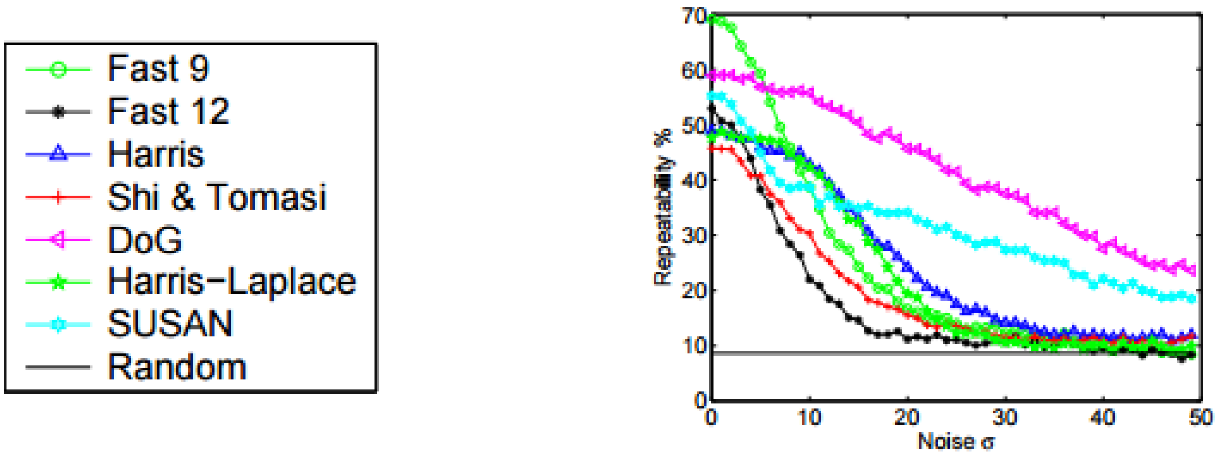








Noise performance evaluation



Conclusion

The author has used machine learning to derive a very fast, high quality corner detector.

Advantages

- Faster
- High levels of repeatability under large aspect changes and for different kinds of feature

Disadvantages

- Not robust to high levels noise
- Can respond to 1px wide lines at certain angles
- Depends on a threshold