UNIVERSITY OF EXTREMADURA

Polytechnic School

Telecommunications Technical Engineering (Image and Sound Speciality)

Escuela Politécnica

END OF DEGREE PROJECT

Image Features extraction for mobile robots navigation

Student: Antonio Redondo López

Tutor: Pedro M. Núñez Trujillo

Introduction

What is the project's scientific basis?

– Computer Vision

– Image Processing

– Robotics

– Visual Odometry

Robotics and Robolab

Robotics and Artificial Vision Laboratory
of the University of Extremadura

Among several Robolab research publications, related ones with the project are:

- **RobEx**: an open-hardware robotics platform(2010).

- **RoboComp**: a Tool-based Robotics Framework (2010).

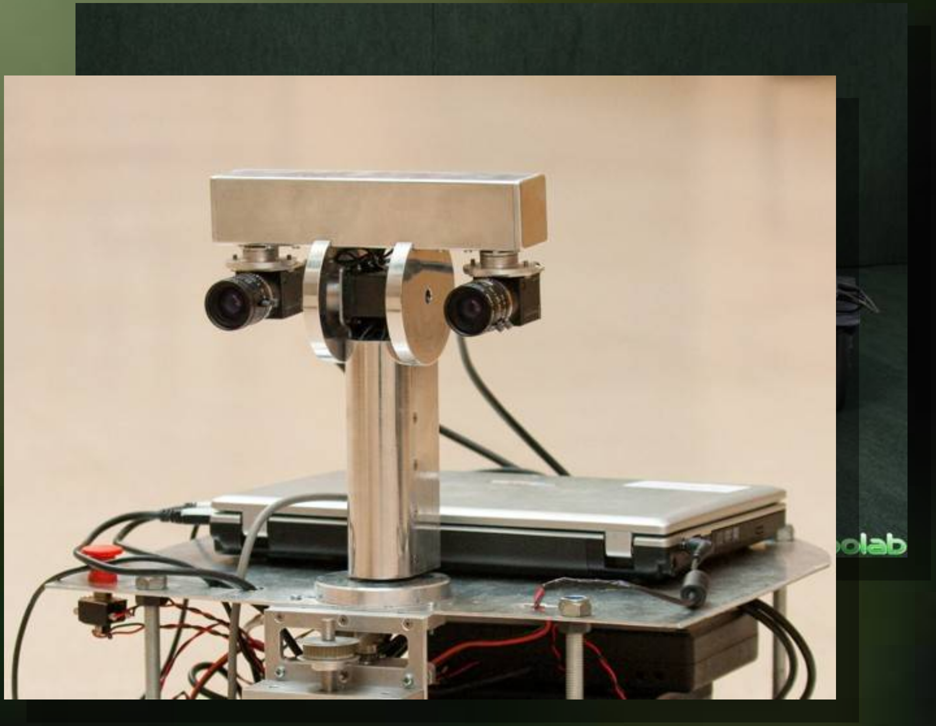


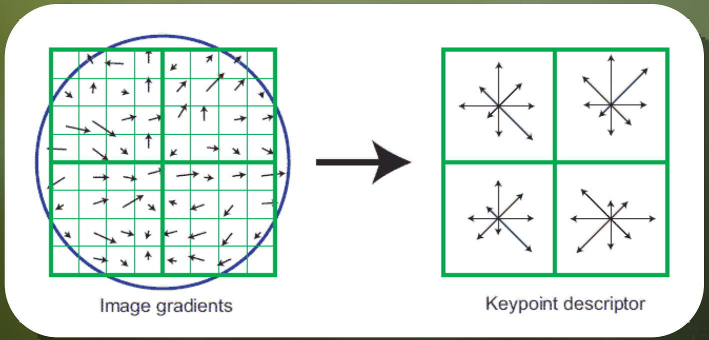
Image Features, Descriptors and Detectors

What Is a Feature

Image features are all those edges, points and shapes that can contain useful information to define an image.

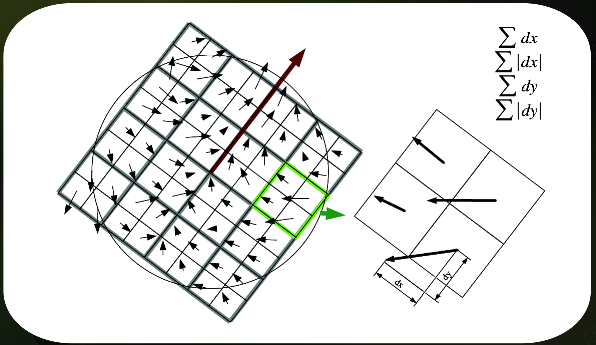
What Is a Descriptor

Some features detectors are ready to use the output information that generate to searching similar detected features in other images.



SIFT Descriptor

SURF Descriptor



What Is a Descriptor

Descriptors visual representation

Harris Detector

New generated imaged

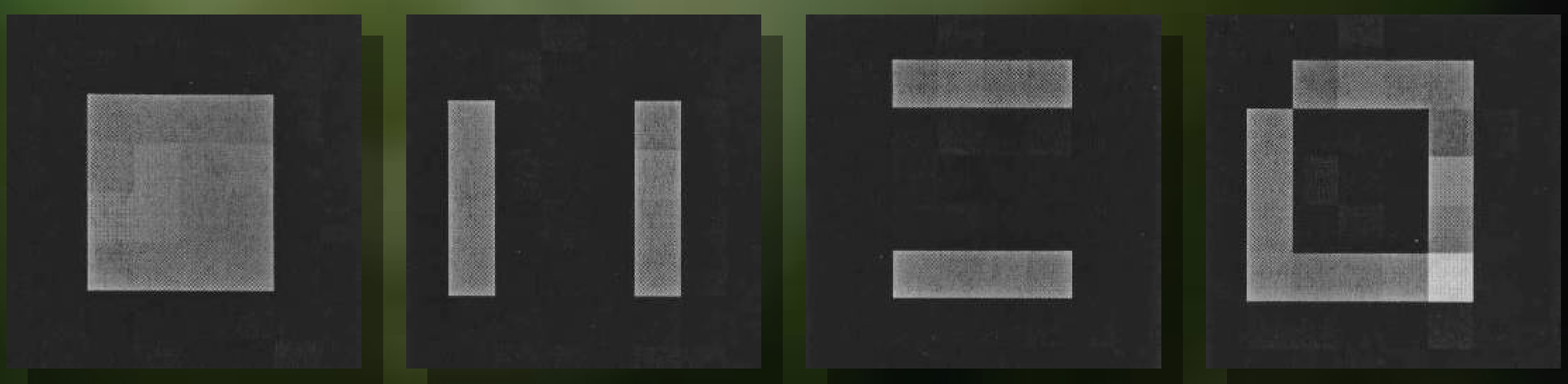
SURF Detector

Superimposed vectors

Main Detectors Overview

The feature detection principle is the differentiation.

$$G(x,y)=\frac{\partial P(x,y)}{\partial x}\cos\theta+\frac{\partial P(x,y)}{\partial y}\sin\theta$$



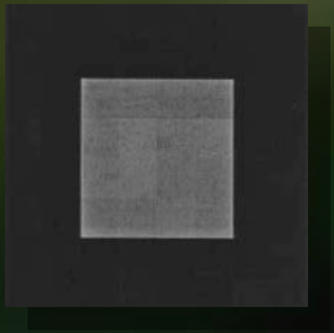
$P(x,y)$ $\qquad \frac{\partial P(x,y)}{\partial x}\cos\theta$ $\qquad \frac{\partial P(x,y)}{\partial y}\sin\theta$ $\qquad G(x,y)$

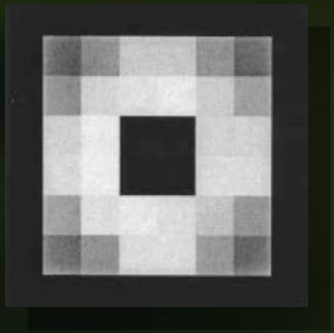
Sobel Operator

Sobel operator (Irwin Sowel, 1968) is thought of the differentiation through axis as a vector. In addition, it adds a low pass filter to smooth the image and be more strength to noise.

$$M = \sqrt{M_x(x, y)^2 + M_y(x, y)^2} \qquad \theta(x, y) = \arctan\left(\frac{M_y(x, y)}{M_x(x, y)}\right)$$

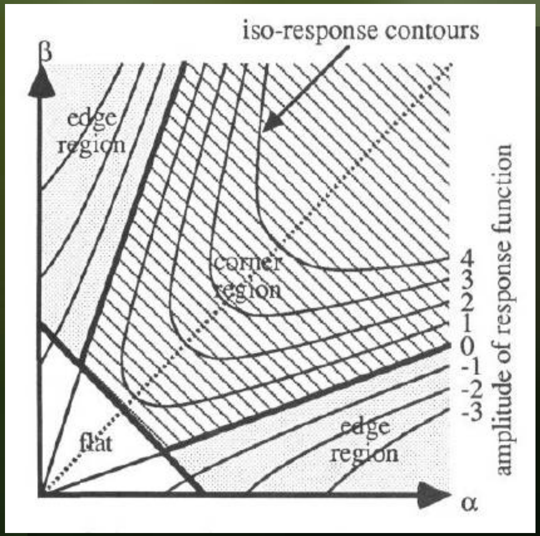
$$M_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \qquad M_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$





Harris Detector

Harris detector (1988) takes advantage of Moravec corner detector (1980). The detector tries to solve link problems between points into textures and surfaces where edge limits are not well define. On the other hand, **Shi and Tomasi** consider that is sometimes more reliable to use the smallest eigenvalue of M, that is, min(α, β) instead of M.



$$X = I * (-1,0,1) \approx \partial I / \partial x = I_{\partial x}$$

$$Y = I * (-1,0,1)^T \approx \partial I / \partial y = I_{\partial y}$$

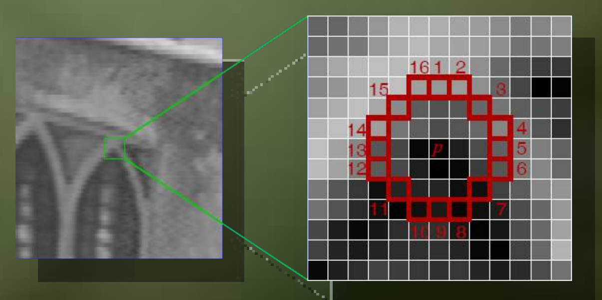
$$w_{u,v} = \exp -(u^2 + v^2) / 2\sigma^2$$

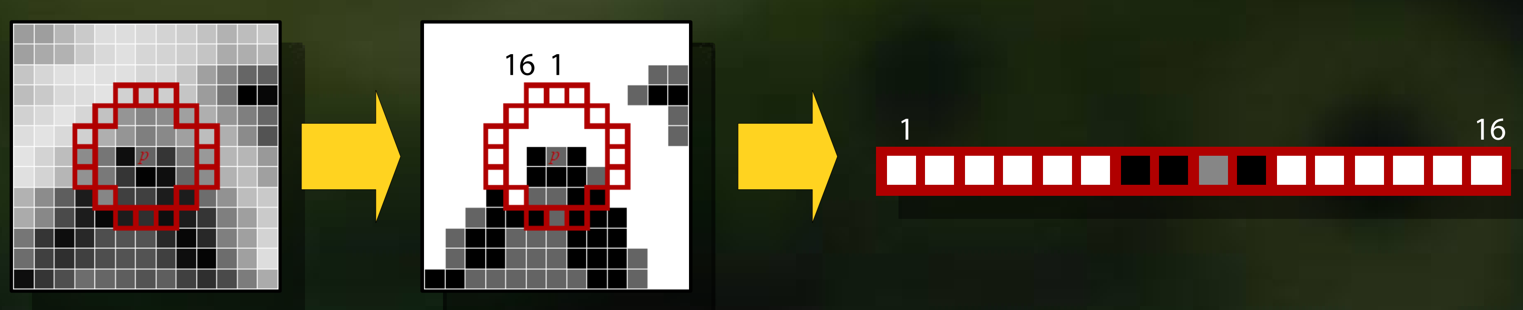
$$M = \begin{bmatrix} X^2 * w & (XY) * w \\ (XY) * w & Y^2 * w \end{bmatrix}$$

$$R = \alpha\beta - k(\alpha+\beta)^2 = Det(M) - k\ Tr^2(M)$$

FAST Detector

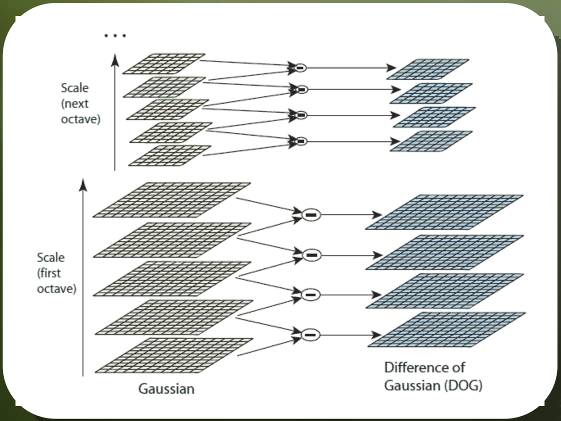
Developed by Edward Rosten and Tom Drummond in 2006, it detects corners by segmenting the image in circles and evaluating the circumference intensity whether the candidate point is a valid corner. To evaluate the pixel circumference pixels it uses a ID3 algorithm that convert it in a very fast detector.





SIFT Detector

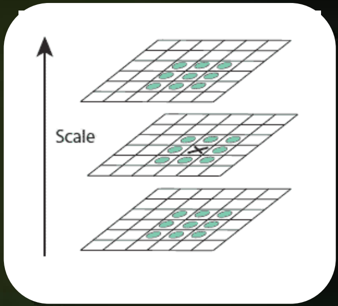
Scale-invariant feature transform (SIFT), published by David Lowe in 1999, uses scale-spaces to detect interesting regions on image. These detected features are stored in descriptors that can been searched in other images.



1. Scale-space extrema detection.
2. Keypoint localization.
3. Orientation assignment.
4. Keypoint descriptor.

It calculates the scale-space extrema in the difference-of-Gaussian function convolved with the image, *D(x, y, σ)*.

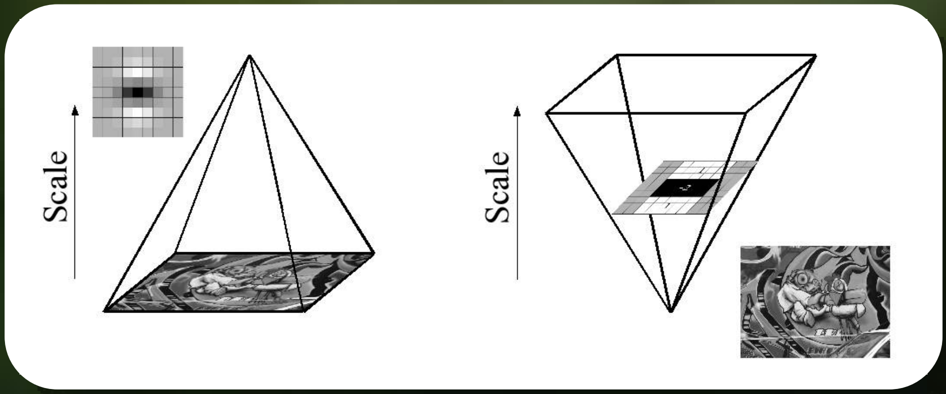
$$D(x,y,\sigma) = (G(x,y,k\sigma)-G(x,y,\sigma))*I(x,y)$$
$$= L(x,y,k\sigma)-L(x,y,\sigma)$$



SURF Detector

SURF (Speeded Up Robust Features) was presented by Herbert Bay et al. in 2003. It uses space-scales like SIFT descriptor but is quite faster than this thanks to (among others) SURF uses a Hessian matrix instead of a difference of Gaussians to detect candidate points.

$$H(x,\sigma) = \begin{bmatrix} L_{xx}(x,\sigma) & L_{xy}(x,\sigma) \\ L_{xy}(x,\sigma) & L_{yy}(x,\sigma) \end{bmatrix}$$



The scale space is analyzed by up-scaling the filter size rather than iteratively reducing the image size.

Comparing Image Descriptors: Proofs and Results

Harris vs. FAST

Timing Results

Timing results for feature detectors run on fields (768x288) of a PAL video sequence in ms, and as a percentage of the processing budget per frame. Approximately 500 features per field are detected.

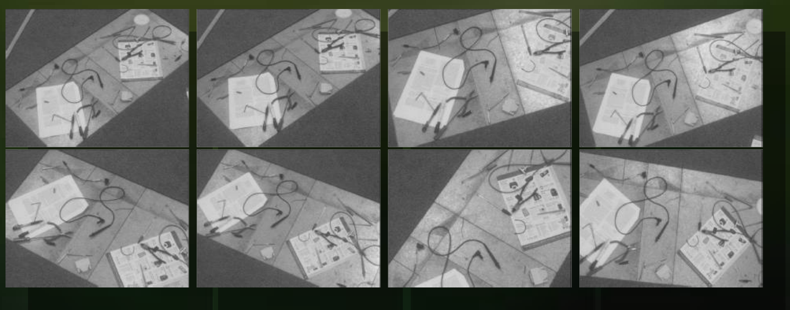
Detector	Opteron 2.6GHz		Pentium III 850MHz	
	Time (ms)	Time (%)	Time (ms)	Time (%)
FAST n = 9 (non-max suppression)	1.33	6.65	5.29	26.5
FAST n = 9 (raw)	1.08	5.40	4.34	21.7
FAST n = 12 (non-max suppression)	1.34	6.70	4.60	23.0
FAST n = 12 (raw)	1.17	5.85	4.31	21.5
Harris	24.0	120	166	830
DoG (SIFT)	60.1	301	345	1280
SUSAN	7.58	37.9	27.5	137.5

Repeatability

The detectors repeatability was tested with three image sets:

Photographs taken of a test rig with strong changes of perspective, scale and radial distortion.

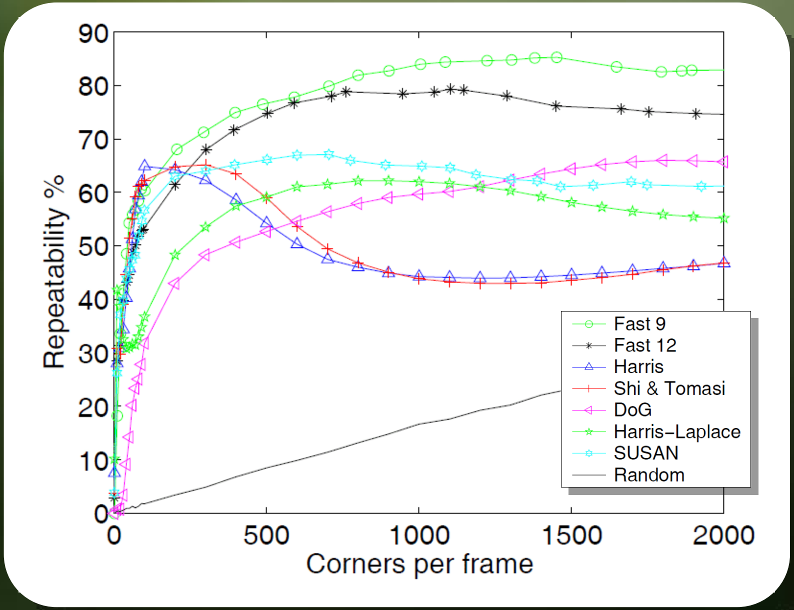
Textural features undergoing projective warps as well as geometric features. There are also big changes of scale.



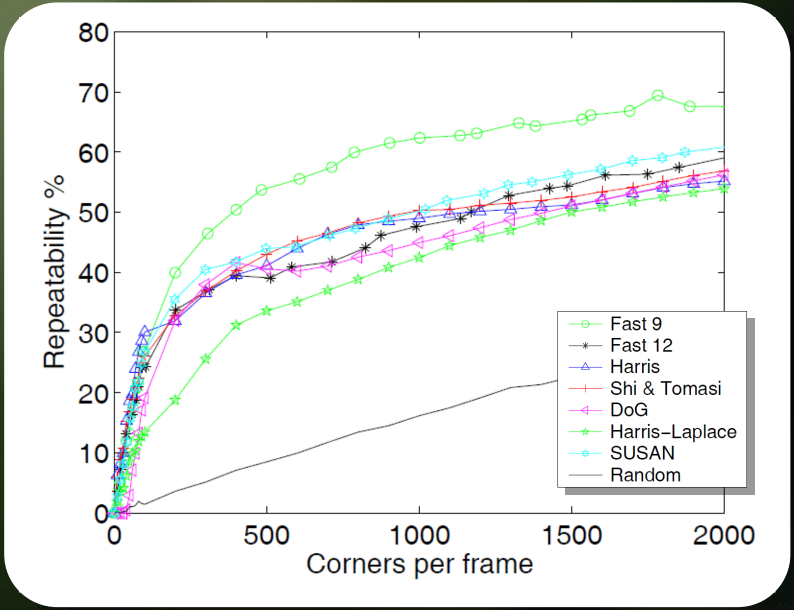
Flat plane with many objects with significant relief. This causes the appearance of features to change in a non affine way.

Repeatability

(a), (b) and (c): repeatability results for the three image sets as the number of features per frame.



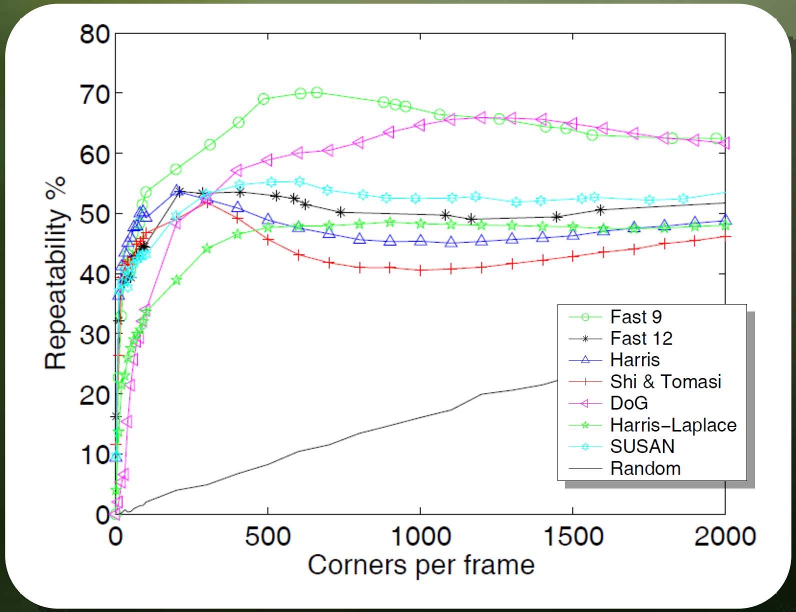
(a)



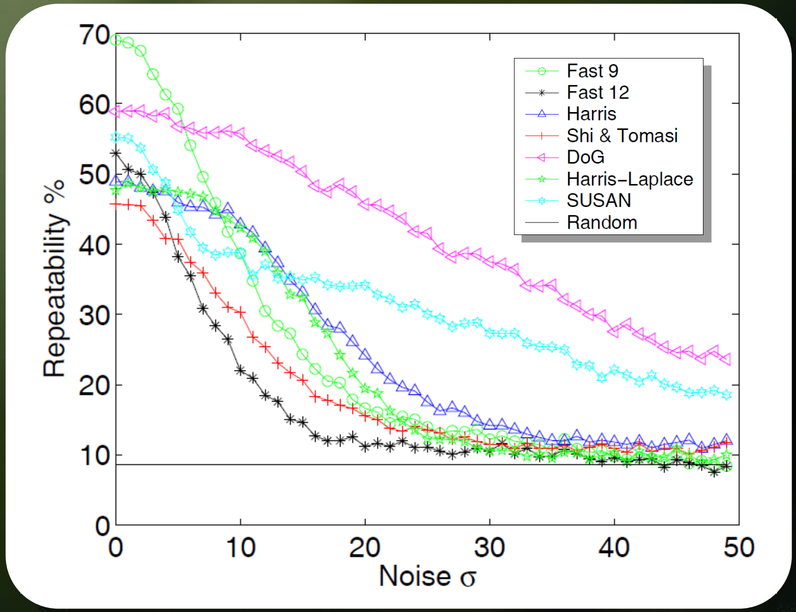
(b)

Repeatability

(d): repeatability results for the bas-relief data set (500 features per frame) as the amount of added Gaussian noise.



(c)



(d)

SIFT vs. SURF

Timing Results

The detector is compared to the Difference of Gaussians (DoG) detector by Lowe (from the SIFT detector), and the Harris Laplace detectors proposed by Mikolajczyk. The thresholds were adapted according to the number of points found with the DoG detector.

Detector	Threshold	Points	Time (ms)
FH-15 (SURF)	60.000	1813	160
FH-9 (SURF)	50.000	1411	70
Hessian-Laplace (Hessian-Affine)	1000	1979	700
Harris-Laplace (Harris-Affine)	2500	1664	2100
DoG (SIFT)	default	1520	400

Repeatability

The detectors repeatability was tested with four image sets:

Viewpoint change

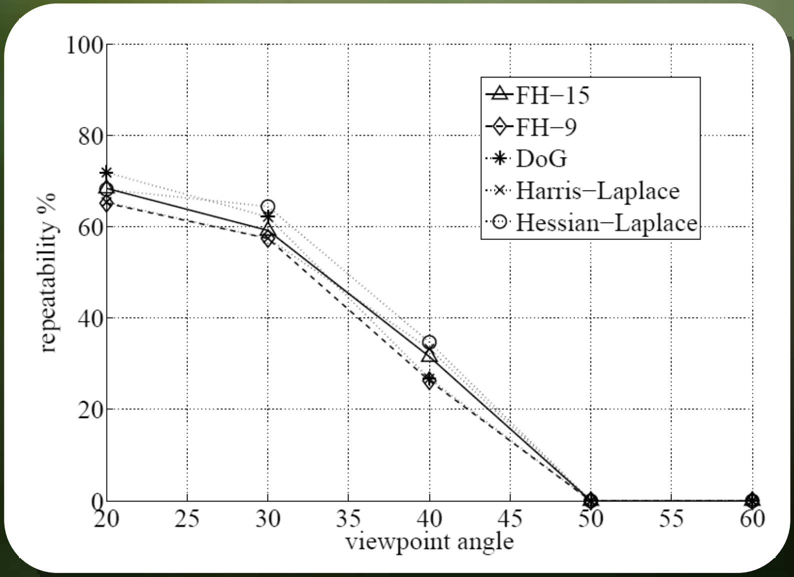
Viewpoint change

Scale change

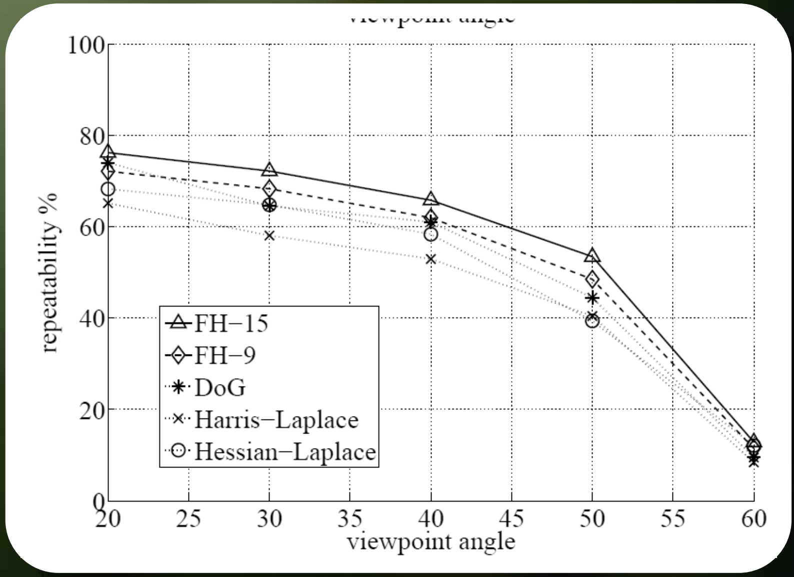
Blur

Repeatability

(a) and (b): Repeatability results for Graffiti and Wall image sets (viewpoint change).



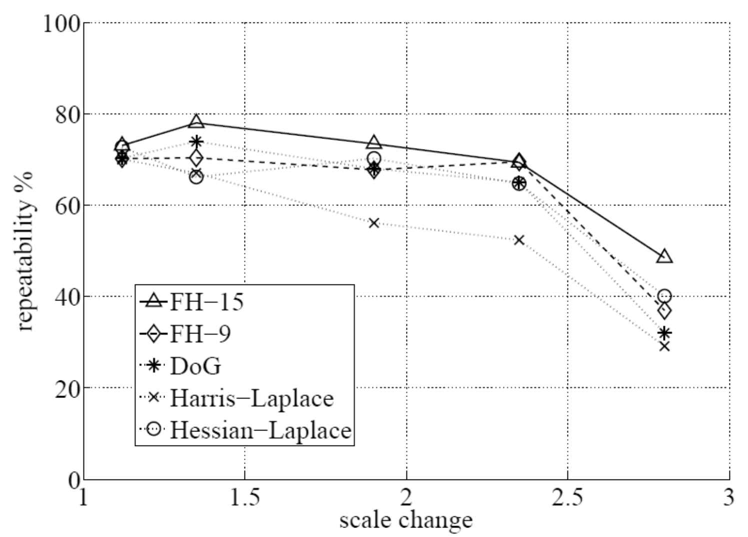
(a)



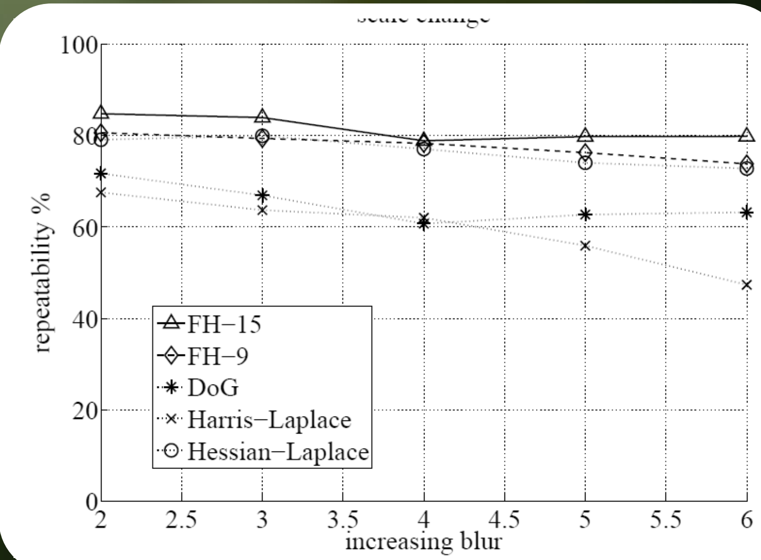
(b)

Repeatability

(c) and (d): repeatability score for Boat image set (scale change) and Bikes image set (image blur).



(c)



(d)

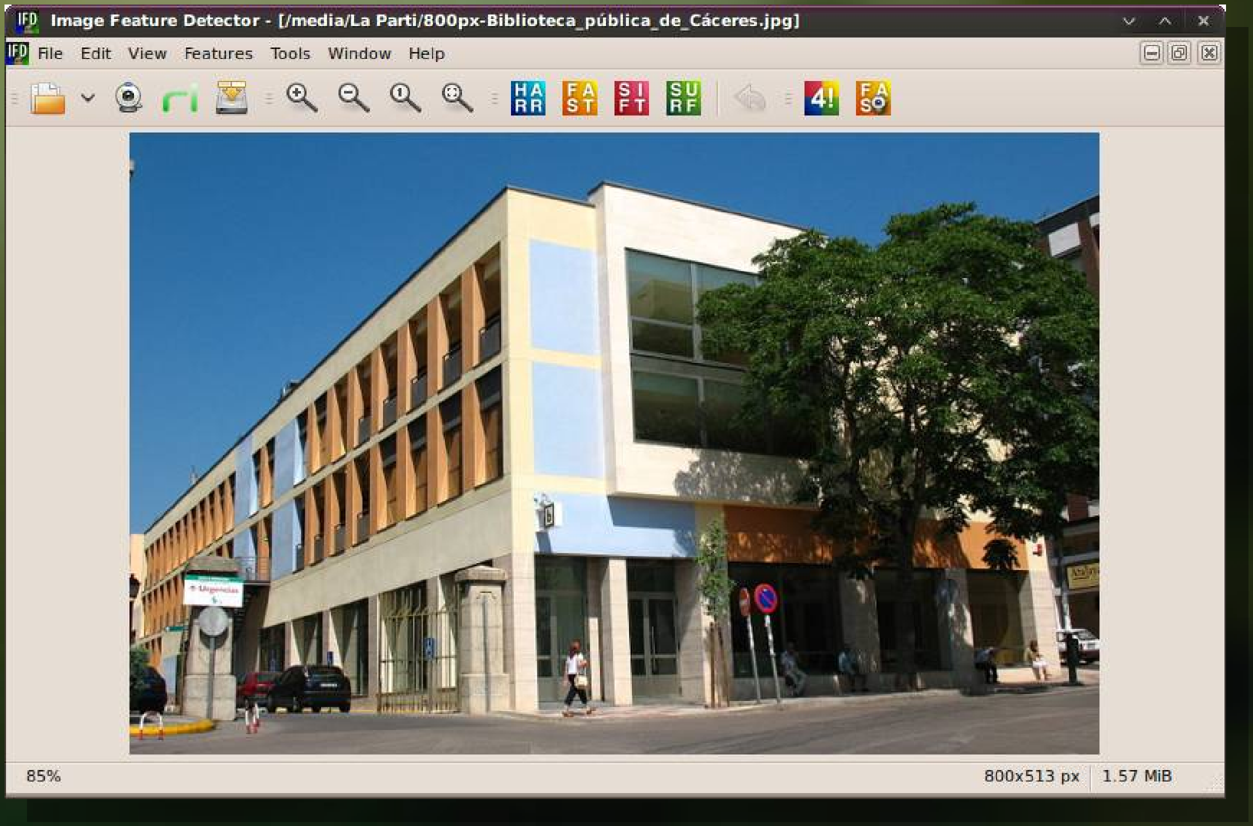
Image Feature Detector: the Computer Program

Overview

Image Feature Detector uses the Qt application framework and the OpenCV library to carry out the implementation of the feature detectors.

Image Feature Detector

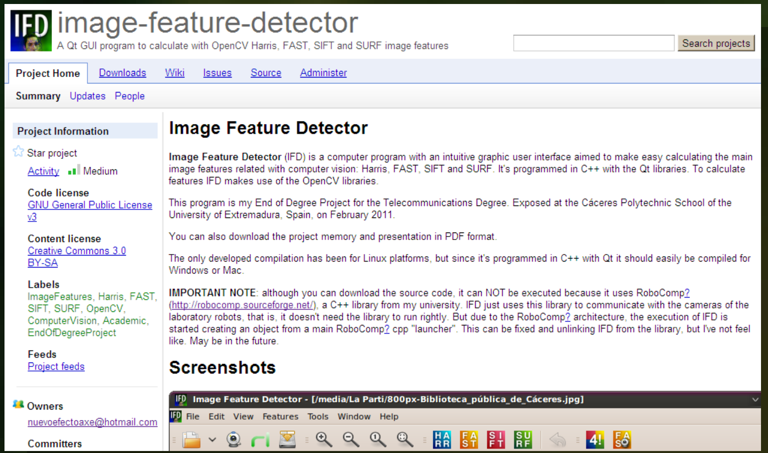
Image Feature Detector GUI has a typical *Main Window* with a *Multiple Document Interface*.



Free as in Freedom

Thanks to their licenses Image Features Extraction for Mobile Robots Navigation project has a website at Google Code to let other students or developers take advantage of the work.

http://code.google.com/p/image-feature-detector



And now let's see the program in action!