#### **Object Recognition in ROS**

Using Feature Extractors and Feature Matchers By Dolev Shapira

## Introduction and Goals

- Provide a reliable, modular system for object recognition.
- Provide guidance for the user how to optimize the use of the system.
- Creation of a "standalone" node that provides an of abstraction between object recognition and other processes.

## Introduction to ROS

#### "What is ROS?

The Robot Operating System (ROS) is a set of software libraries and tools that help you build robot applications. From drivers to state-of-theart algorithms, and with powerful developer tools, ROS has what you need for your next robotics project. And it's all open source."

ROS.org

## Introduction to ROS

ROS provides an interface that allows the user to create a modular independent pieces of code called "Nodes", communicating with each other using "Topics".

## Introduction to ROS

- Nodes "processes".
- Topics "message boards".
- Subscribers Nodes "watching" the topics and reading incoming messages.
- Publishers Nodes publishing ("posting") messages into topics.

# OpenCV

OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products.

## **Feature Detection**

We have seen in this course a biological mechanism used for feature detection.

- Simple cells Orientation.
- Complex cells Orientation and Direction.
- Hypercomplex cells "end-stop" property.
  The only question in hand is, what feature should we trace?

## **SIFT Feature Detector**

- Scale invariant.
- Rotation invariant.
- Functions well even when there is a change in illumination.
- Function relatively well even when there's noise.

David G. Lowe 2004

# How Does it Work?

- We load a set of "Templates" prepared beforehand.
- We sample images from the camera (scenes).
- We detect features both in the scene, and the templates.
- We match between the two using a Matcher.

## Templates

- Should not be noisy.
- Should not contain a transparent part of an object.
- Should not contain a reflective part of an object.
- Should contain "static" features.
- Should be of a part proportional to the object itself (with relation to the resolution).

#### Results





### Results

