Grasping Estimation using Trajectory Curvatures

Ongoing work

Diego Resende Faria

Institute of Systems and Robitcs

FCTUC

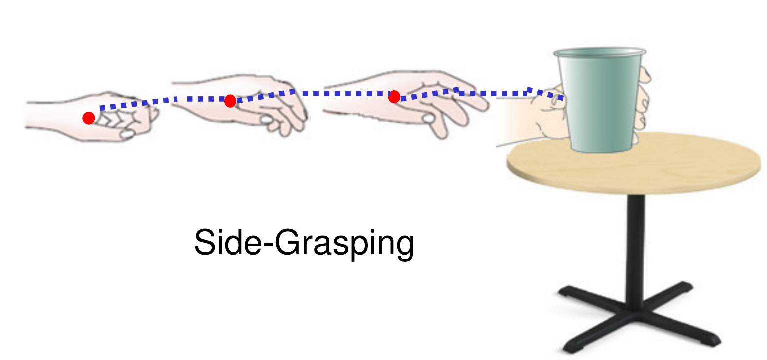
BACS

Mobile Robotics Laboratory
Institute of Systems and Robotics
ISR – Coimbra

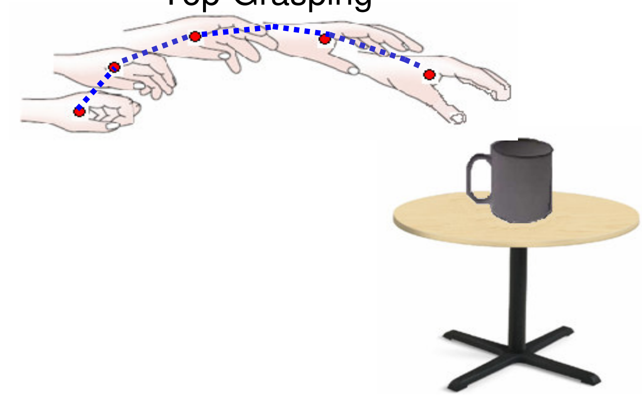
Goals

- Development of Visual tracking to analyze the way that a person grasps an object (in a reach to grasp task) using a probabilistic approach: The probabilistic framework uses low level features like skin color features and kinematics constraints for the human body; Times series is used for tracking prediction;

- Using High level Classifier of Trajectory Curvatures we can estimate (Bayesian Method) the way that a human grasps an object.



Top-Grasping

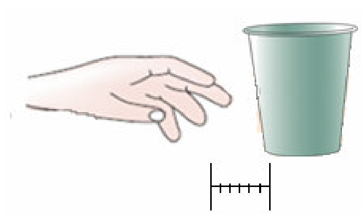


Goals

- Object Detection/Localization:

  Knowing when the object is grasped by the distance between hand and object; Local Features for object detection.

Hand close to object: distance >= Threshold



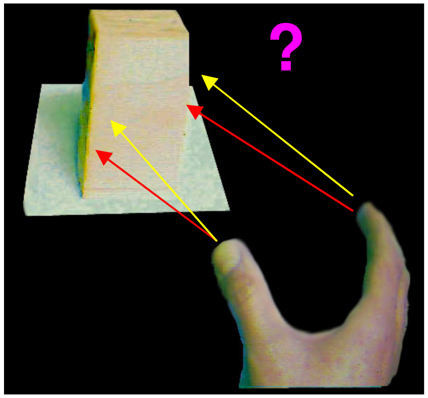
Introduction

Grasping movements have been the focus of interest of many researches, including neuroscientists, roboticists, etc. To understand the grasping behaviour some questions are done:

How to grasp an object?

Objects can be grasped in several different ways: the visual properties of an object influence the chosen grip.

Where should you place your fingers when grasping an object?



Depends on movement goal: lift, rotate, flip, push, slide

How human achieves grasping

Having a prior knowledge of the object

Seek the target

Identify the target among other objects

-Move the arm towards the object

-Preshape the hand to hold the object

Contact between hand and object

Refine the grasping and reposition the fingers

Successful grasp achieved

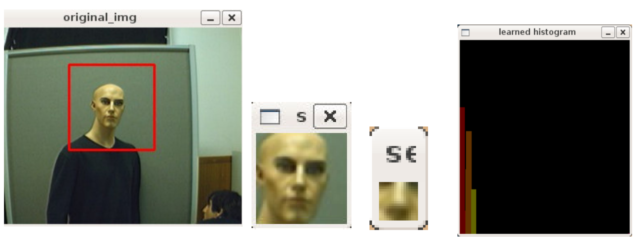
Visual Tracking Development

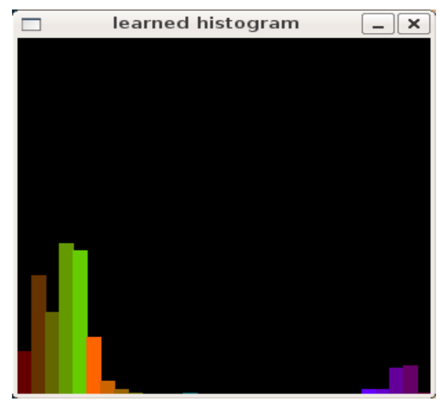
Learning Stage:

- Low level features: Skin color Histogram

  - The frequency distribution of skin color intensity is done analyzing the Hue channel from HSV image;

  - Samples of skin color were collected from faces images using haar-like features and then is built a histogram for each one;





-A mean histogram is calculated using 30 bins representing the hue range (0 – 180º).

Visual Tracking Development

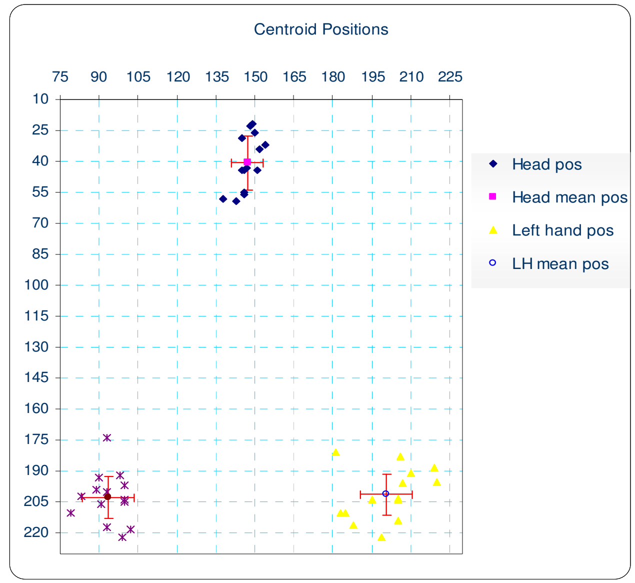
Learning Stage:

- Kinematics Constraints

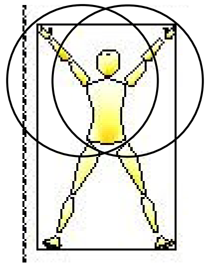
  - Samples of face and hands positions were acquired from people in front of the camera;

  - A mean value and STD of face and hand position and their sizes were calculated;

  - This process is useful for:
    - search window to initialize the tracking system;
    - avoid skin color objects detection bigger than face and hands.



Visual Tracking Development



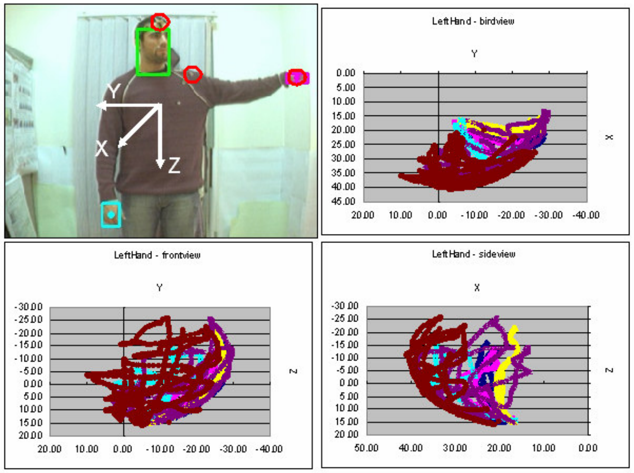
Learning Stage:

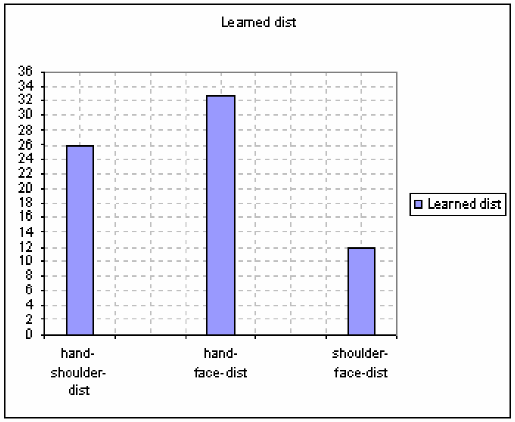
- Kinematics Constraints:

-Knowing the kinematics we can restrict the positions where the tracking can act, it is useful for the tracking does not detecting another skin color object out of its range, avoiding to lose the hands.

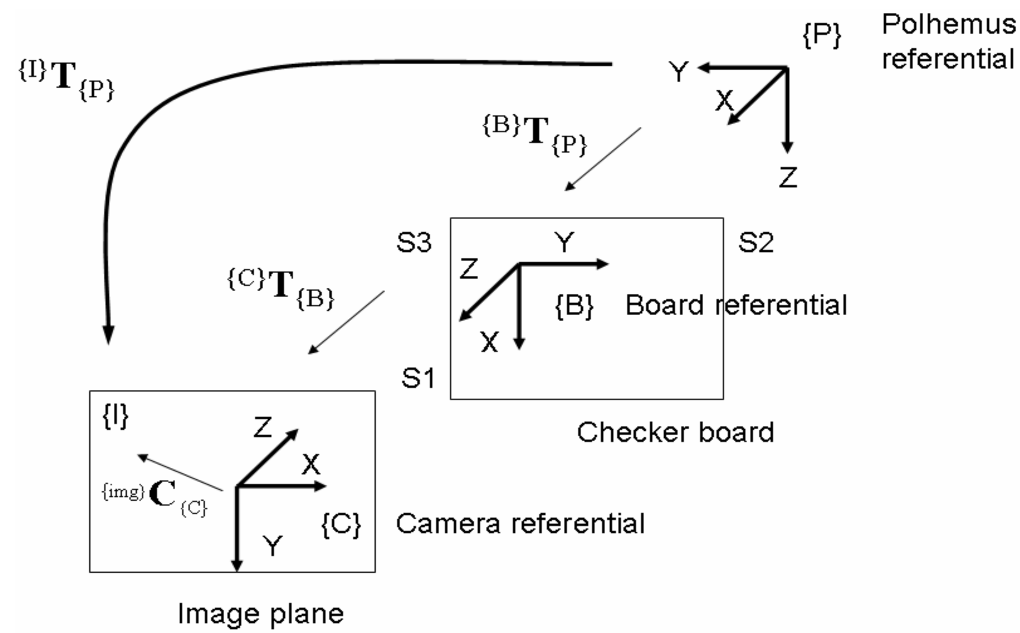
- For that was used a 3D tracker device, polhemus liberty system.
- Samples of people movement were collected (3D points of the movement)
- A mean value and standard deviation were calculated.

- A calibration step was developed: Cam-Pol (camera and tracker device) to use the 3D learned data In the 2D image plane.





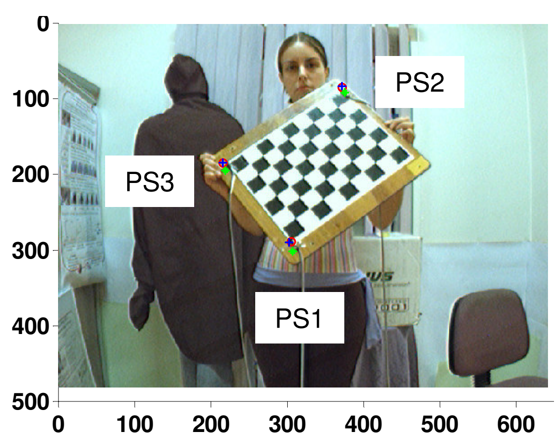
Visual Tracking Development

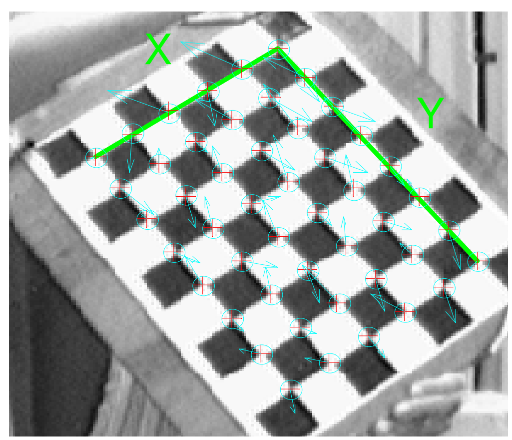


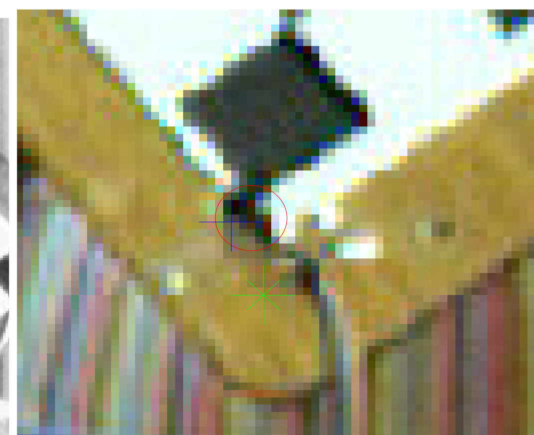
Cam-Pol Calibration:

-3 sensors were used in the chessboard,

- Results: From some trials we achieve a Mean and STD Transformations matrix for the transformation of 3-D Polhemus data (in {P}) to 2-D image data (in {I}).







Mobile Robotics Laboratory
Institute of Systems and Robotics
ISR – Coimbra

Visual Tracking Development

Skin color Map:

The probability distribution image : probability of each pixel belonging to the skin color for object detection. This is done by histogram back-projection algorithm.

$$\left\{ \hat{P}_u = \min\left(\frac{255}{\max\left(\hat{q}\right)} \hat{q}_u \,,\, 255 \right) \right\}_{u=1\ldots m}$$

Where q represents the learned skin color histogram

CAMShift algorithm:

- Mean Shift embedded (it finds the densest region using Statistical Moments);
- Adaptive region-sizing step;
- The kernel is a simple step function applied to a skin-probability map;
- After to choose a initial position (from learning stage) is calculated the zero and first moment to find the centroid. The search window is centred at the mean location found. The process is then repeated for continuous tracking.

$$M_{00} = \sum_x \sum_y I(x, y) \quad M_{10} = \sum_x \sum_y xI(x, y) \quad M_{01} = \sum_x \sum_y yI(x, y) \quad x_c = \frac{M_{10}}{M_{00}}; \quad y_c = \frac{M_{01}}{M_{00}};$$

Visual Tracking Development

Hand position and velocity prediction step: Time Series

A common technique in business and economic forecasting was used for tracking prediction. Double Exponential Smoothing ***relies on the idea that user motion can be adequately modeled by a simple linear trend equation with slope and y-intercept parameters that vary slowly over time***.

The Holt's model is given by: $\hat{Y}_t = \alpha \hat{Y}_t + \beta \hat{Y}_{t-1} + \varepsilon_t$

The serie mean level and the slope are estimated by L_t and T_t as follows:

$$L_t = \alpha Y_t + (1 - \alpha)(L_{t-1} + T_{t-1}) \qquad 0 < \alpha < 1$$

$$T_t = \beta (L_t - L_{t-1}) + (1 - \beta) T_{t-1} \qquad 0 < \beta < 1$$

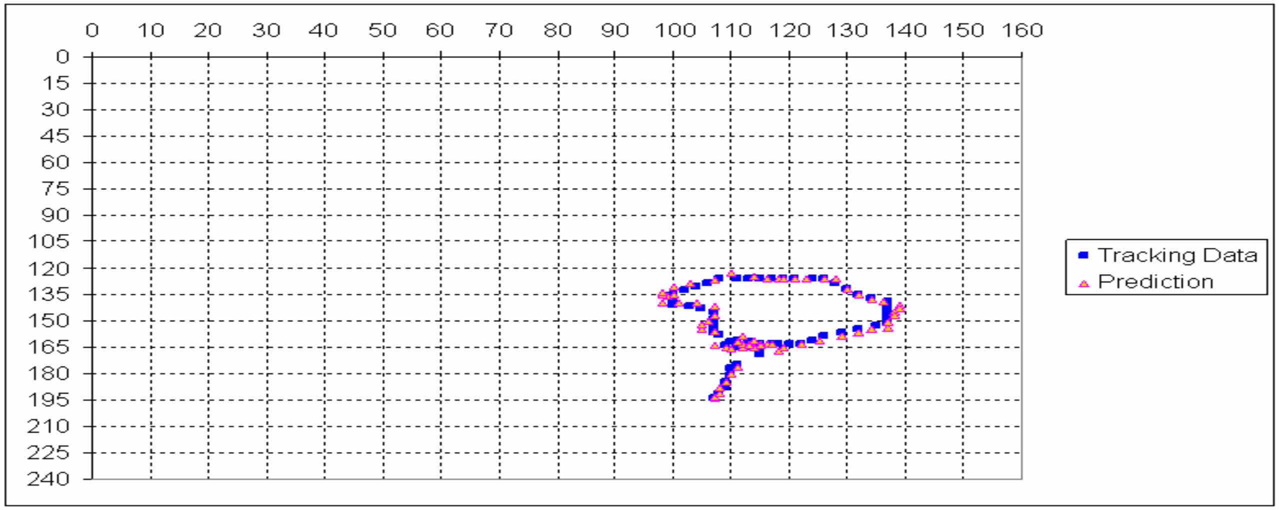
The prediction is given by: $\hat{Y}_t(h) = L_t + h.T_t \qquad \forall h > 0$

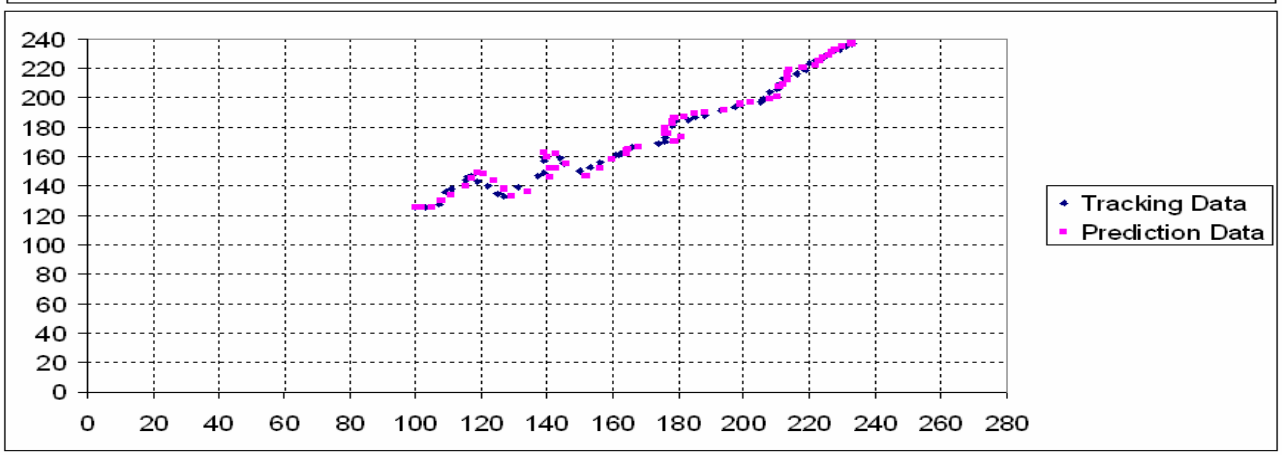
The alpha and beta can be found in different ways, in our case the values are found using **MSE** after analysing some observation

Visual Tracking Development

Prediction step: Time Series - Double Exponential Smoothing

Simulated data





Trajectory Curvature Detection

Hands Trajectories Curvature Detection by Second order Derivative

Given 3 points we can verify the trajectory curvature:

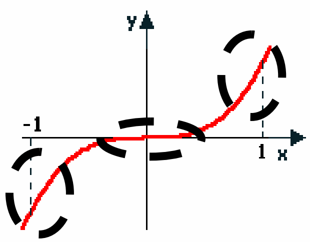
$$D1 = \begin{cases} (y2 - y1) / (x2 - x1), & (x2 - x1) \neq 0 \text{ and } (y2 - y1) \neq 0; \\ 0, & (y2 - y1) == 0 \text{ or } ((y2 - y1) == 0 \text{ and } (x2 - x1) == 0); \\ (y2 - y1), & (x2 - x1) == 0 \end{cases}$$

$$D2 = \begin{cases} (y3 - y2) / (x3 - x2), & (x3 - x2) \neq 0 \text{ and } (y3 - y2) \neq 0; \\ 0, & (y3 - y2) == 0 \text{ or } ((y3 - y2) == 0 \text{ and } (x3 - x2) == 0); \\ (y3 - y2), & (x3 - x2) == 0 \end{cases}$$

Curvature = D2 – D1;

Curvature Types Mapping

$$\text{Curvature} = \begin{cases} 1, & \text{curvature} > 0 \rightarrow \text{Up} \\ 0, & \text{curvature} == 0 \rightarrow \text{Line (no curvature)} \\ -1, & \text{curvature} < 0 \rightarrow \text{Down} \end{cases}$$



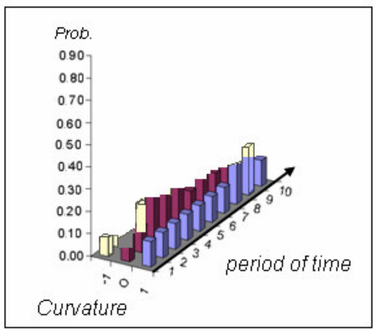
A trajectory can be composed by these 3 types of curvature

Mobile Robotics Laboratory
Institute of Systems and Robotics
ISR – Coimbra

Curvature Trajectory Learning

Grasping Learning Table Composed by Trajecory Curvatures

From Low Level features we have the hand position displacement $\Delta h_{(x,y)}$ known as trajectory, and calculating the 2º order Derivative ***f''*** we achieve the curvatures of a trajectory.



Given a Set of observations (obtained from low level features) to represent a type of Grasping ***G***, in each period of time ***P*** that is composed of 7 image frames, we have the probability of each type of curvature ***C***.

$$P(C \mid G\ P)$$

After some people performing the same type of grasping *G*, a mean histogram is built counting the number of different curvature values that appear for a grasping type along a period of time. This is done for each type of grasping, so that a Curvature Learning Table is built for each one. In each period of time *P*, for each 3 points given by hand position displacement, the curvature *c* is calculated. The training data is a set of *C* for each *G*.

Grasping Estimation

The General Grasping Classification Model

Definitions

1. ***g*** is a known grasping from all possible *G* (Grasping types);
2. ***c*** is a certain value of feature *C* (Cuvature types);
3. ***i*** is a given index from all possible period of time *P* (each period of time is composed by *7 Images Frames*). It represents a new feature in each period of time that indicates online classification.

Learning:

The probabilty $\mathbf{P}(c \mid g, i)$ that feature C has certain value c can be defined by learning the probability distribution $\mathbf{P}(C \mid G, P)$.

Classification:

Knowing $\mathbf{P}(c \mid G, i)$ and the prior $\mathbf{P}(G)$ we can able to apply Bayes rule and compute the probability distribution for G given the period i and the feature c. Initially, grasping types *G* is a uniform distribution and during the classification its values is updated applying Bayes rule. In each period of time i, in each 3 points is calculated the curvature given by 2ª order derivative f'', this is done until the final of period i, and then a mean value of curvature *c* is calculated, and the curvature result is analyzed in the learning table in the same period i, and then its probability is used to update the grasping types *G.*

$$\mathbf{P}(G_{t+1} \mid c_{t+1}, i_{t+1}) = \mathbf{P}(c_{t+1} \mid G, i) \; \mathbf{P}(G_t)$$

Mobile Robotics Laboratory
Institute of Systems and Robotics
ISR – Coimbra

Grasping Estimation

The Bayes rule is aplied in each period of time P until the end (P_n=10 -> 70 frames).

After all updating $\forall g \in G$ we use as rule for classification the maximum a posteriori method (MAP).

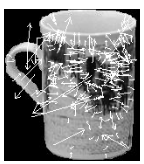
A normalization to get the percentage of each type of graspng is done, so that the sum of the percentage be 100%, then:

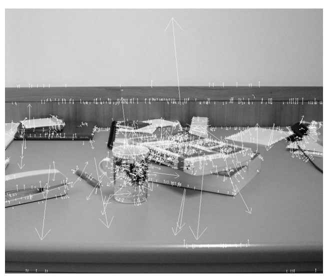
$$Normalization\ of\ \ g = \frac{P(g)}{\sum_{i=1}^{n} g_i}$$

Object Localization

• The intention of localizing the object that will be grasped is due to detect when the hand is close to object (when the object is grasped).

• A solution to acquire the object localization is recognize it by SIFT descriptor (Scale-invariant feature transform). Some interesting properties of SIFT is *invariant to image scale and rotation, robust to changes in illumination, noise, occlusion and minor changes in viewpoint*.





Conclusion

• This application is useful to know how a human grasp an object. We can use this information for imitation learning task to endow a robot to imitate the human.

•Some key points that influences the way that a human grasps an object are:

- the task goal;
- the object shape / size / weight / size of the contact surface, etc.