Visual servoing

- vision allows a robotic system to obtain geometrical and qualitative information on the surrounding environment
  - high level control → motion planning *(look-and-move – visual grasping)*
  - low level control → measures used in the control loop
- *visual servoing* control is based on feedback of visual measurements
  - *position-based visual servoing*
  - *image-based visual servoing*
  - *hybrid visual servoing*
- image processing is aimed at extracting numerical information referred to as *image feature parameters*
- *pose estimation methods* are based on the measurement of a certain number of points or *correspondences*
- numerical pose estimation methods are based on the integration of the linear mapping between the camera velocity in the operational space and the time derivative of the feature parameters in the *image plane*
- *camera calibration* is based on the measurement of a certain number of correspondences

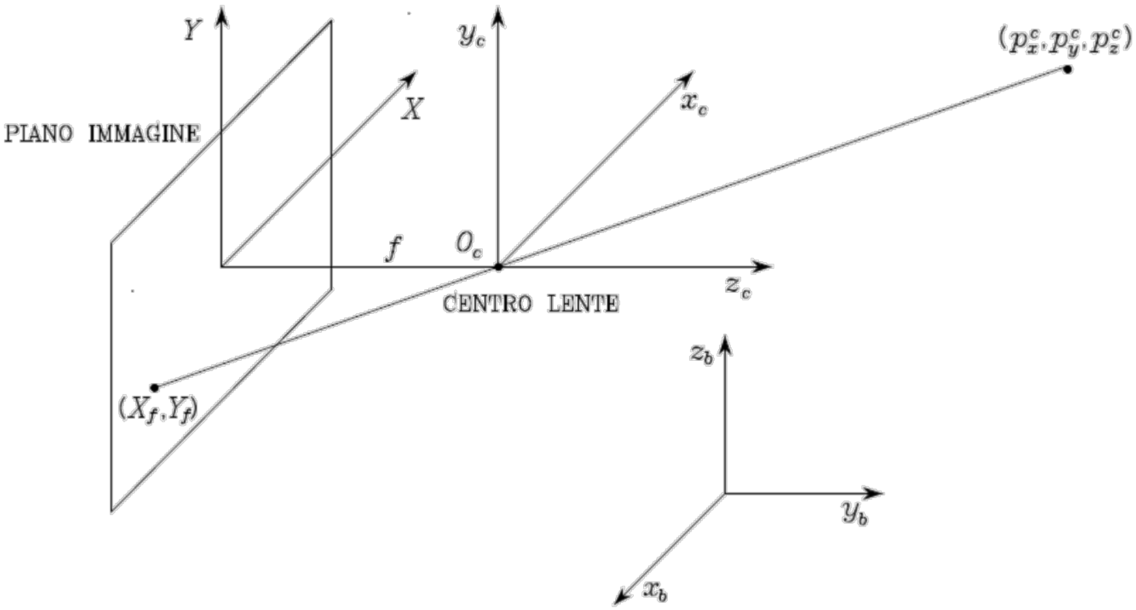
Configuration of the visual system

- multi-camera systems
  - information about its depth by evaluating its distance with respect to the visual system → *3D vision* or *stereo vision*
- mono-camera systems
  - two images of the same object from two different poses
  - if only a single image is available, the depth can be estimated on the basis of geometrical characteristics of the object known in advance. This
  - cheaper and easier to calibrate
  - lower accuracy
- *eye-to-hand* → fixed location
  - advantage is that the camera field of view does not change during the execution of the task, implying that the accuracy of such measurements is constant
  - the manipulator occludes, in part or in whole, the view of the objects
- *eye-in-hand* → mobile configuration
  - the camera is placed on the manipulator
  - high variability in the accuracy of measurements
  - the accuracy becomes almost constant and is usually higher than that achievable with eye-to-hand cameras
- *hybrid configuration* consisting of one or more cameras in eye-to-hand configuration, and one or more cameras in eye-in-hand configuration
  - ensures a good accuracy throughout the workspace, while avoiding the problems of occlusions

Image processing

- visual information is very rich and varied
  - complex and computational expensive transformations before it can be used for controlling a robotic system
    - extraction of numerical information from the image → *image feature parameters*
- two basic operations
  - *segmentation* → a representation suitable for the identification of measurable features of the image
  - *interpretation* → measurement of the feature parameters of the image
- the source information is contained in a two dimensional memory array representing the spatial sample of the image
  - *image function* ***I(X,Y)*** is a vector function whose components represent the values of one or more physical quantities related to the pixel in a sampled and quantized form
    - light intensity in the wavelengths of red, green and blue
    - or in shades of gray (number of gray levels depends on resolution – 256 gray levels)

Perspective transformation



$$X_I = \alpha_x f \frac{p_x^c}{p_z^c} + X_0$$

$$Y_I = \alpha_y f \frac{p_y^c}{p_z^c} + Y_0$$

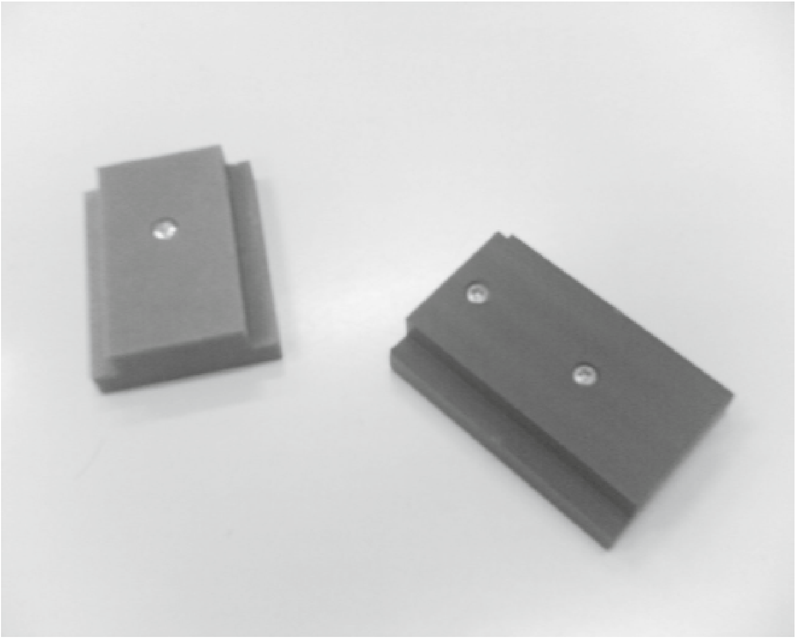
$$\lambda \begin{bmatrix} X_I \\ Y_I \\ 1 \end{bmatrix} = \boldsymbol{\Omega \Pi} \begin{bmatrix} p_x^c \\ p_y^c \\ p_z^c \\ 1 \end{bmatrix}$$

$$\boldsymbol{\Omega} = \begin{bmatrix} f\alpha_x & 0 & X_0 & 0 \\ 0 & f\alpha_y & Y_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \qquad \boldsymbol{\Pi} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$\lambda \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} = \boldsymbol{\Pi} \begin{bmatrix} p_x^c \\ p_y^c \\ p_z^c \\ 1 \end{bmatrix} \qquad \boldsymbol{\Xi} = \boldsymbol{\Omega \Pi T}_b^c$$

Gray-level histogram

- provides the frequency of occurrence of each gray level in the image
- the gray levels are quantized from 0 to 255
- the value $h(p)$ of the histogram at a particular gray level $p \in [0, 255]$ is the number of image pixels with gray level p
- if this value is divided by the total number of pixels, the histogram is termed *normalized histogram*



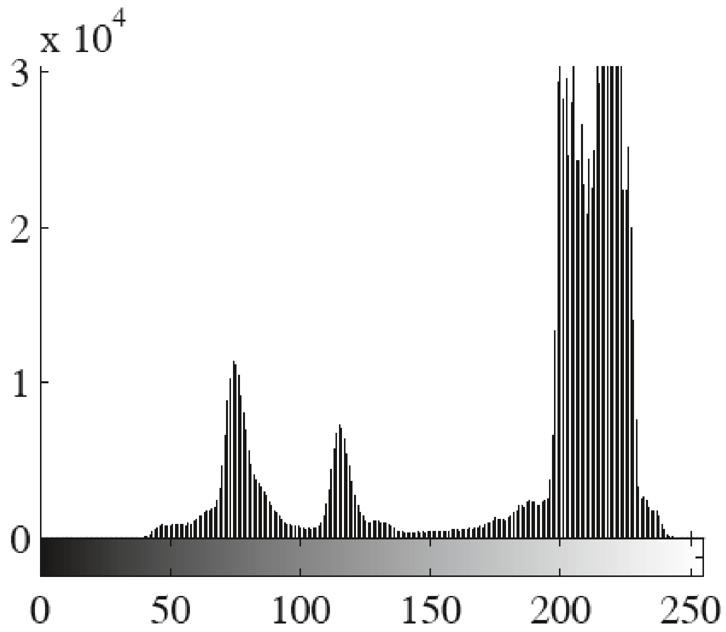
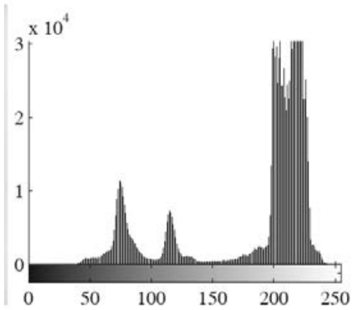


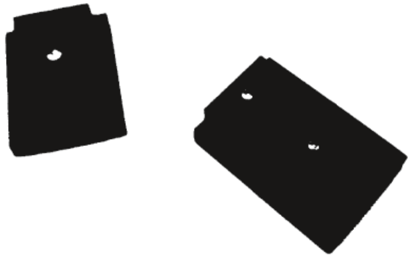
Image segmentation

- consists of a grouping process, by which the image is divided into a certain number of groups, referred to as *segments* (component of each group similar with respect to one or more characteristics)
  - distinct objects of the environment
  - or homogeneous object parts
- finding connected *regions* of the image
  - grouping sets of pixels sharing common features into two-dimensional connected areas
  - high memory usage
  - low computational load
- detection of *boundaries*
  - identifying the pixels corresponding to object contours and isolating them from the rest of the image
  - the boundary of an object, once extracted, can be used to define the position and shape of the object itself
  - complementary

Region-based segmentation

- obtaining connected regions by continuous merging of initially small groups of adjacent pixels into larger ones
  - if the pixels belonging to these regions satisfy a common property, termed *uniformity predicate* (verifying gray level)
  - *binary segmentation* or image *binarization* by comparing the gray level of each pixel with a threshold l
  - the peaks of the histogram are termed *modes (for the dark objects the closest minimum to the left)*
- in the presence of multiple objects, a further elaboration is required to separate the connected regions corresponding to the single objects
- the gray-scale histogram is noisy and the modes are difficult to identify
- various techniques have been developed to increase the robustness of binary segmentation
  - appropriate filtering of the image before binarization
  - algorithms for automatic selection of the threshold





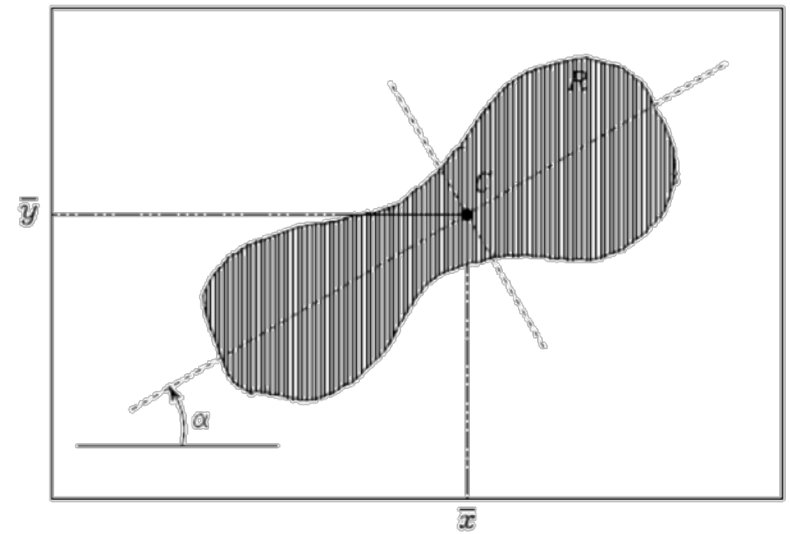
Boundary-based segmentation

- boundary-based segmentation techniques usually obtain a boundary by grouping many single local edges
  - corresponding to local discontinuities of image gray level
  - local edges are sets of pixels where the light intensity changes abruptly
- the algorithms for boundary detection
  - derive an intermediate image based on local edges from the original gray-scale image
    - construct short-curve segments by edge linking
      - obtain the boundaries by joining these curve segments through geometric primitives often known in advance
- edge detection is essentially a filtering process whereas boundary detection is a higher level task usually requiring more sophisticated software
- edge detection can be performed by grouping the pixels where the magnitude of the gradient is greater than a threshold
- in case of simple and well-defined shapes, boundary detection becomes straightforward and segmentation reduces to the sole edge detection
- several edge detection techniques exist, most of them require the calculation of the gradient or of the Laplacian of function $I(X_I, Y_I)$

Image interpretation

- image interpretation is the process of calculating the image feature parameters from the segments, represented in terms of boundaries or in terms of regions
- characterize the position, orientation and shape of the two-dimensional object
- Moment of a region

$$m_{i,j} = \sum_{X_I, Y_I \in \mathcal{R}} I(X_I, Y_I) X_I^i Y_I^j$$



- $m_{0,0}$ area of the region, total number of pixels
- coordinates of the centroid

$$\bar{x} = \frac{m_{1,0}}{m_{0,0}} \qquad \bar{y} = \frac{m_{0,1}}{m_{0,0}}$$

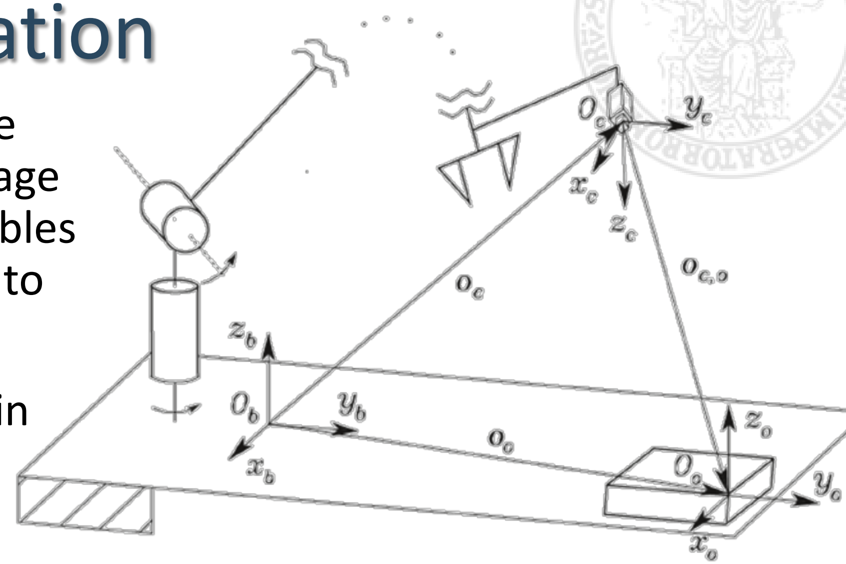
- central moments

$$\mu_{i,j} = \sum_{X_I, Y_I \in \mathcal{R}} (X_I - \bar{x})^i (Y_I - \bar{y})^j$$

- second order central moments → inertia moments $\mu_{2,0}$ $\mu_{0,2}$
- inertia tensor relative to the center of mass (eigenvalues define the principal moments of inertia, eigenvectors define the principal axes of inertia)

$$\mathcal{I} = \begin{bmatrix} \mu_{2,0} & \mu_{1,1} \\ \mu_{1,1} & \mu_{0,2} \end{bmatrix}$$

Pose estimation



- visual servoing is based on the mapping between the feature parameters of an object measured in the image plane of the camera and the operational space variables defining the relative pose of the object with respect to the camera
- often it is sufficient to derive a differential mapping in terms of velocity (easier to solve – linear, numerical integration algoritms)
- (k×1) vector s, termed feature vector (normalized pixel coordinates)

$$\boldsymbol{s} = \begin{bmatrix} X \\ Y \end{bmatrix} \quad \widetilde{\boldsymbol{s}} = \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix}$$

- the problem to solve is of computing the elements of $\boldsymbol{T}_o^c = \begin{bmatrix} \boldsymbol{R}_o^c & \boldsymbol{o}_{c,o}^c \\ \boldsymbol{0}^T & 1 \end{bmatrix}$

from the measurements of object feature parameters in the camera image plane

- consider n points on the object and the corresponding position vectors with respect to the object frame

$$\boldsymbol{r}_{o,i}^o = \boldsymbol{p}_i^o - \boldsymbol{o}_o^o, \; i = 1, \ldots, n,$$

- projections of points on the image plane defines the feature vector $\boldsymbol{s} = \begin{bmatrix} \boldsymbol{s}_1 \\ \vdots \\ \boldsymbol{s}_n \end{bmatrix}$ $\quad \boldsymbol{s}_i = \begin{bmatrix} X_i \\ Y_i \end{bmatrix}$

Pose estimation

- the homogeneous coordinates of the points of the object with respect to the camera frame can be expressed as

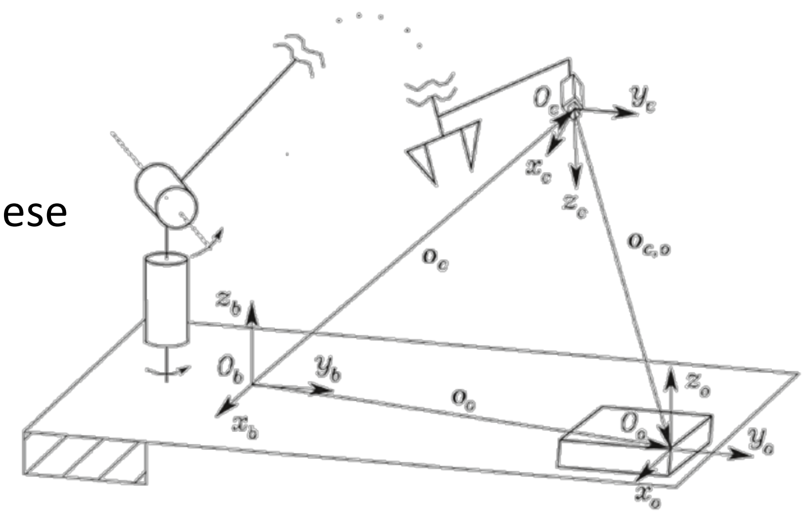
$$\widetilde{\boldsymbol{r}}^{c}_{o,i} = \boldsymbol{T}^{c}_{o}\widetilde{\boldsymbol{r}}^{o}_{o,i}$$

- the homogeneous coordinates of the projections of these points on the image plane are given by

$$\lambda_i \widetilde{\boldsymbol{s}}_i = \boldsymbol{\Pi}\boldsymbol{T}^{c}_{o}\widetilde{\boldsymbol{r}}^{o}_{o,i} \qquad \lambda_i > 0$$

- n correspondences are available for n points of the object, whose coordinates are known both in the object frame and in the image plane

- these correspondences define a system of equations to be solved for the unknown elements of matrix $\boldsymbol{T}^{c}_{o}$



The visual servoing problem

- visual measurements are used to compute the end-effector pose with respect to an object observed by the camera
- on the basis of visual measurements elaborated in real time the goal is to reach and keep a (constant or time-varying) desired pose with respect to the observed object
- the direct measurements provided by the visual system are concerned with feature parameters in the image plane
- the robotic task is defined in the operational space
  - *position-based visual servoing* termed *visual servoing in the operational space*
  - *image-based visual servoing* termed *visual servoing in the image space*

Position-based visual servoing

- conceptually similar to the operational space control
- but feedback is based on the real-time estimation of the pose of the observed object with respect to the camera using visual measurements
  - the estimation can be performed analytically
  - iterative numerical algorithms
- advantage → possibility of acting directly on operational space variables
- drawback → due to the absence of direct control of the image features, the object may exit from the camera field of view during the transient or as a consequence of planning errors
- the feedback loop is open due to lack of visual measurements and instability may occur

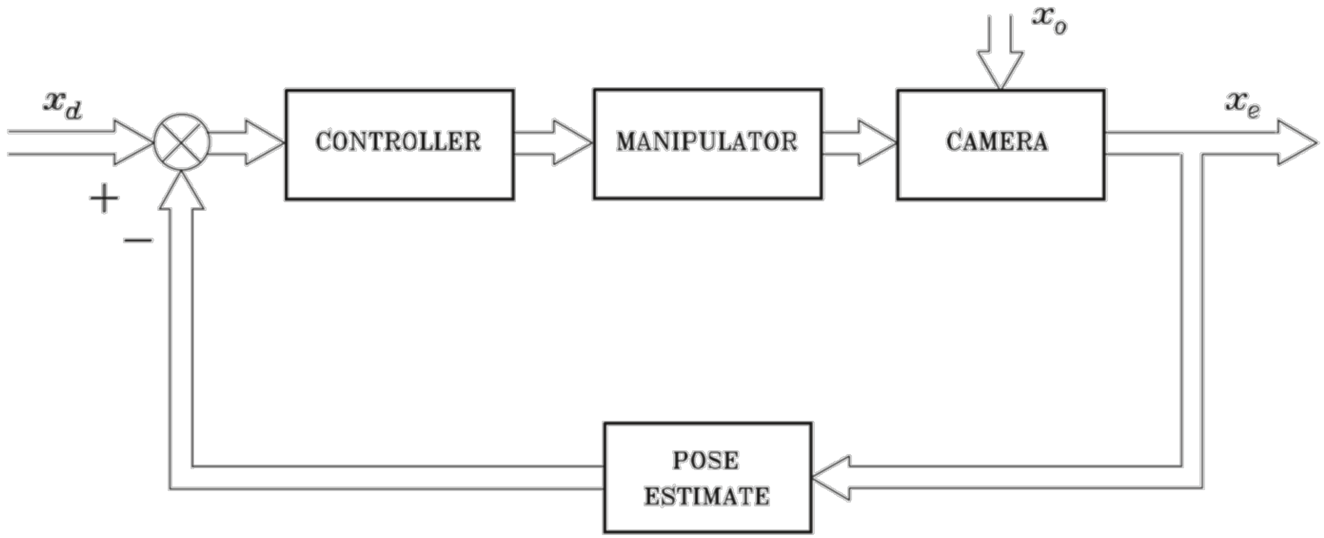
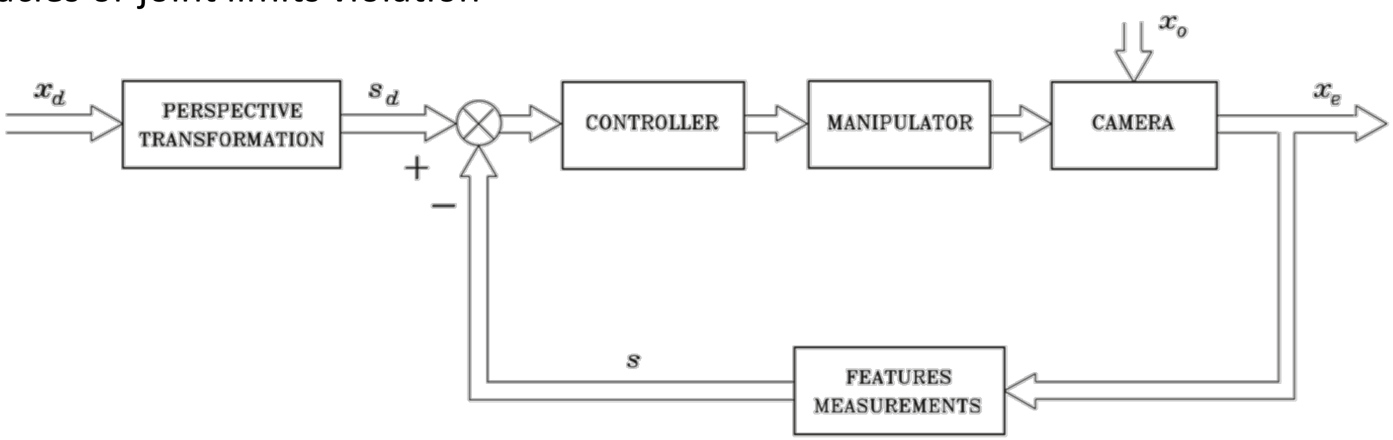


Image-space visual servoing

- the error is defined as the difference between the value of the image feature parameters in the desired configuration and the value of the parameters measured with the camera in the current pose
- advantage
  - real-time estimate of the pose of the object with respect to the camera is not required
  - it is possible to keep the object within the camera field of view during the motion
- disadvantage
  - due to the nonlinearity of the mapping between the image feature parameters and the operational space variables, singular configurations may occur, which cause instability or saturation of the control action
  - the end-effector trajectories cannot be easily predicted in advance and may produce collisions with obstacles or joint limits violation



Comparison

- camera calibration
  - position-based visual servoing is more sensitive to camera calibration errors compared to image-based visual servoing
    - for the first approach, the presence of uncertainties on calibration parameters, both intrinsic and extrinsic, produces errors on the estimate of operational space variables that may be seen as an external disturbance
  - in the image-based visual servoing approach, the quantities used for the computation of the control action are directly defined in the image plane and measured in pixel units
    - the desired value of the feature parameters is measured using the camera
    - the uncertainty affecting calibration parameters can be seen as a disturbance acting on the forward path of the control loop, where disturbance rejection capability is high
- geometric model of the object
  - for position-based visual servoing, the object geometry must be known if only one camera is used, because it is necessary for pose estimation, while it may be unknown when a stereo camera system is used
  - image-based visual servoing does not require, in principle, knowledge of the object geometry, even for mono-camera systems
- for both approaches, the problem of regulation to a constant set-point is presented
- the object is assumed to be fixed with respect to the base frame
- single calibrated camera, mounted on the manipulator's end-effector
- the end-effector frame is chosen so as to coincide with the camera frame