VIMANCO: VIsion MAnipulation of Non-Cooperative Objects

Clean Space Industrial Days

ESTEC

25 October 2017

K. Kapellos

Konstantinos.Kapellos@trasysinternational.com

VIMANCO: *VIsion MAnipulation of Non Cooperative Object*

- Objectives
  - Development of a system allowing:
    - Specification
    - Validation via simulation
    - Implementation & demonstration

    of Vision Based Control for the manipulation of *non-cooperative objects* in space applications

- Results
  - Vision Processing Library
    - Object Recognition
    - Object Tracking
    - Visual Servoing
  - Real-Time Controller / Emulator
  - Vision Simulator
  - Ground Control Station
  - Vision System

- ESA TRP activity
  - TRASYS
  - IRISA/INRIA Rennes
  - K.U. Leuven
  - Galileo Avionica

Plan

- VIMANCO Approach
  - Object recognition
  - Visual Tracking
  - Visual Servoing

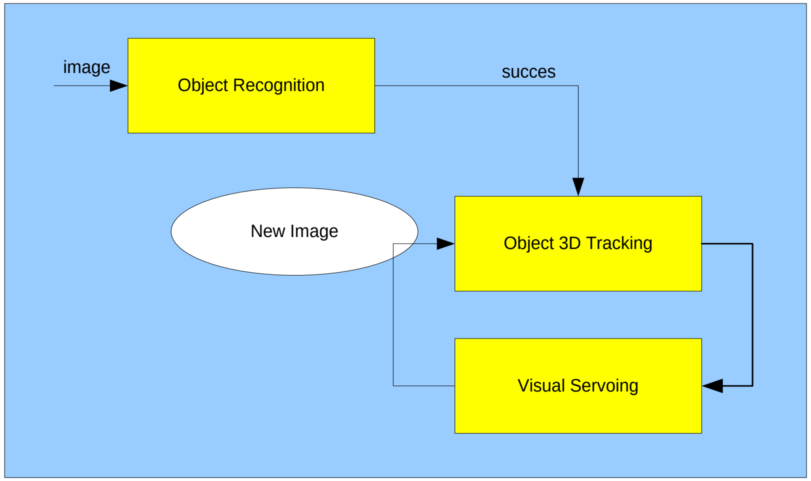
- VIMANCO Utilisation & Results
  - Eurobot Testbed & WET / SSBB / MSSTM - EGP

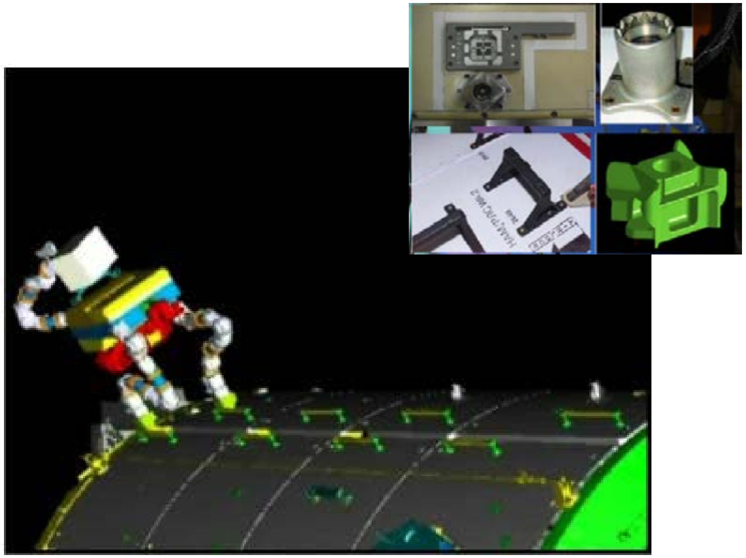
- Simulation environment
  - 3DROV

- Ground Control Station
  - DREAMS / 3DROCS

General Concept

- <u>Object Recognition</u>
  - Identify in an image a particular object and determine its position when seen in new situations
- Object Tracking
  - Determine the position of a particular object in every image acquired by one or more cameras
- <u>Visual Servoing</u>
  - Control the motion of a dynamic system using the data provided by one or more cameras



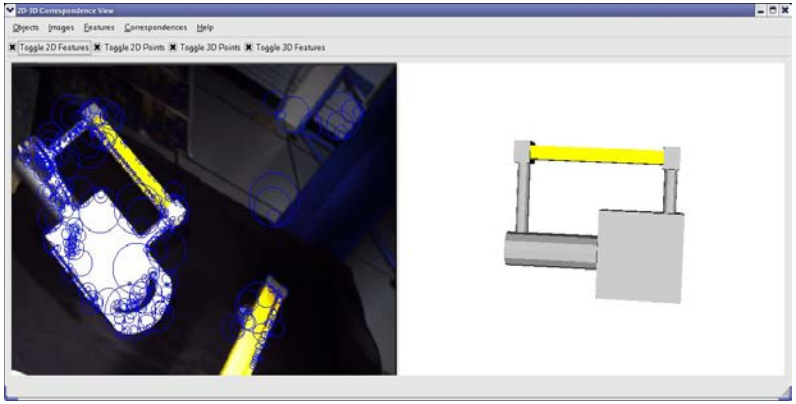


TRASYS
WE GET IT DONE

Object Recognition

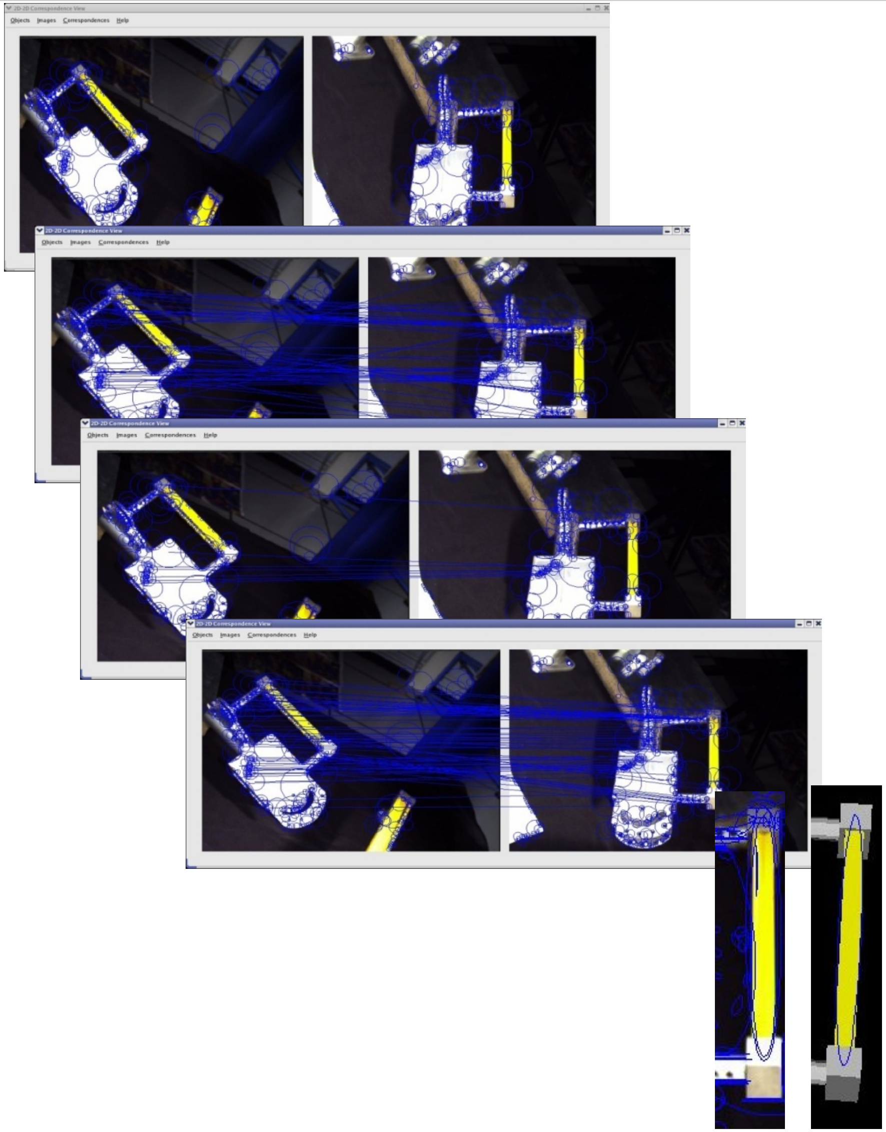
- Proposed Approach
  - Based on local invariant features
  - Combines advantages of model-based and appearance-based approaches
  - Feature consists of two parts:
    - Localization in the image: typically 2D coordinates of point, ellipse, …
    - Description of the feature: feature vector e.g. window of grayscale values around a point
  - Two stages process: off-line training (modeling) and on-line object recognition
  - Same algorithm for modeling and recognition

- Off-line Training
  - Acquisition of a set of ‘training’ images under different viewpoints and illumination conditions
  - For each image:
    - Compute features and descriptions
    - Indicate correspondences between 2D and 3D
    - Result: 3D coordinates for extracted features
  - The set of the features, their description and their 3D position constitutes the model of the object



Object Recognition

- On-line: for a target image
  - Compute features and descriptors using the same algorithm applied on the ‘training’ images
  - Match with the model image
  - Compute the consistent matches (P-RANSAC) and the corresponding camera pose
  - Using the computed pose, identify the new matches in the neighborhood of the projected 3D points (P-guided matching)
  - Verification step



Result: the position of the object in the image used as input to the Object Tracking / Visual Servoing loop

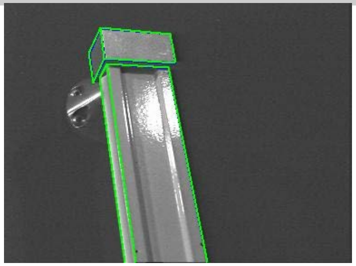
Object Tracking

- Allows the localisation, at video rate, of a given object, by using for each frame:
  - A CAD 3D model of the considered object
  - One (or more) current image(s) of this object acquired by one (or more) camera(s)
  - The previous localisation of the object in the image

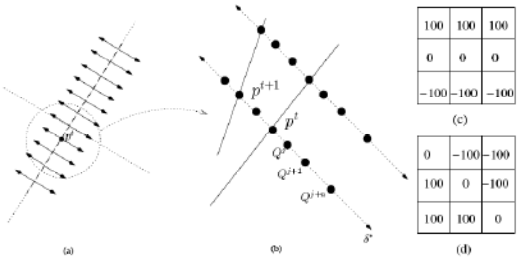
- The localisation is based on:
  - A robust Virtual Visual Servoing (VVS) technique
    - A virtual camera is moved from the previously determined pose to a pose where the projected contour of the object matches the set of extracted points
    - At convergence, the current pose of the virtual camera gives the pose of the object
  - A robust estimator directly included into the VVS control law in order to correctly reject potential outliers

- Robustness with respect to occlusions and light variation

Object Tracking: Algorithm

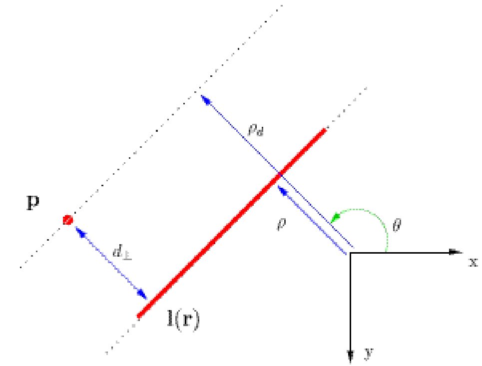


- Acquisition of a new image or set of images following the configuration
- Computation of the forward projection of the CAD model
- Subsample the forward projection of the CAD model to obtain a set of point p^t. The position of each point is computed in the current image using the ME algorithm providing a set of p^{t+1} points that should belong to the current object contour.



- VVS: until convergence
  - Build the set of desired feature s
  - Computation of the interaction matrix Ls
  - Computation of the confidence in each data using the M-estimator and s. *The matrix D represents the confidence in each data.*
  - Computation of the new position of the virtual camera:
    - v = - λ (D Ls) $^+$ (s – s*)
    - if ||D * s|| < ε convergence is obtained

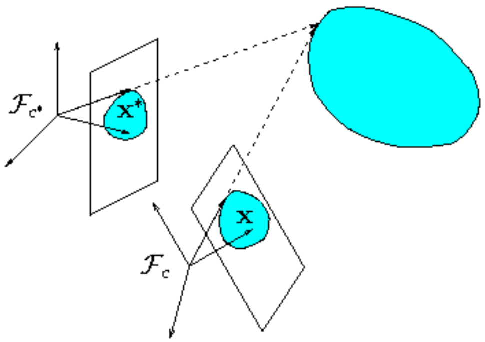
$$\mathbf{s} = \begin{bmatrix} d_l^1 \\ \cdot \\ \cdot \\ d_l^k \end{bmatrix} \qquad \mathbf{L_s} = \begin{bmatrix} \mathbf{L}_{d_l^1} \\ \cdot \\ \cdot \\ \mathbf{L}_{d_l^k} \end{bmatrix}$$

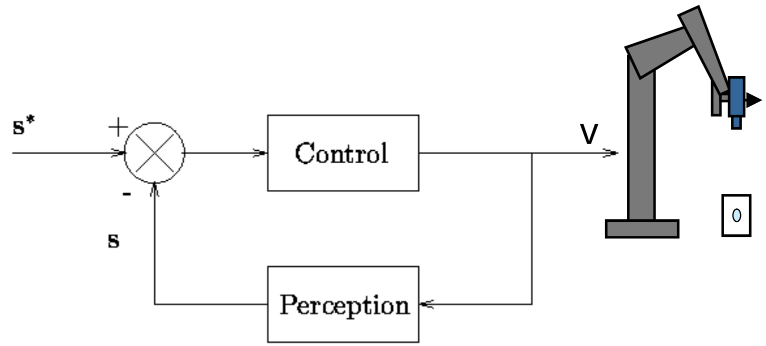


- At convergence the position $^cM_o^{t+1}$ is available to be used for the Visual Servoing

Visual Servoing

- Visual Servoing techniques consist of using the data provided by one or several cameras in order to control the motion of a robotic system





- Modeling
  - Design from the visual measurements ***x(t)*** a set of visual features ***s*** allowing the control of the desired degrees of freedom

$$\mathbf{s} = \mathbf{s}(\mathbf{x}, \mathbf{a}) \qquad\qquad \dot{\mathbf{s}} = \mathbf{L}(\mathbf{s}, \mathbf{a}, \mathbf{z})\ \mathbf{v}$$

- Control
  - Design a control law so that the features ***s*** reach a desired value ***s**** defining a correct realization of the objective

$$\mathrm{v} = \mathrm{f}\left(\widehat{\mathrm{L}^{+}}\ (\mathrm{s} - \mathrm{s}^{*})\right)$$

TRASYS

WE GET IT DONE

Visual Servoing: Algorithm

- Compute the $\boldsymbol{s}(t)$ of k visual information from the current camera position ${}^{c}\mathbf{M}_{o}(t)$
- Compute the $\boldsymbol{s}^{*}$ representing the desired value of $\boldsymbol{s}$ that guarantees the correct execution of the activity
- Compute the error $\boldsymbol{e}$
- Compute the interaction matrix $\mathbf{L}_{s}$ linked with $\boldsymbol{s}$
- Compute the robot control output

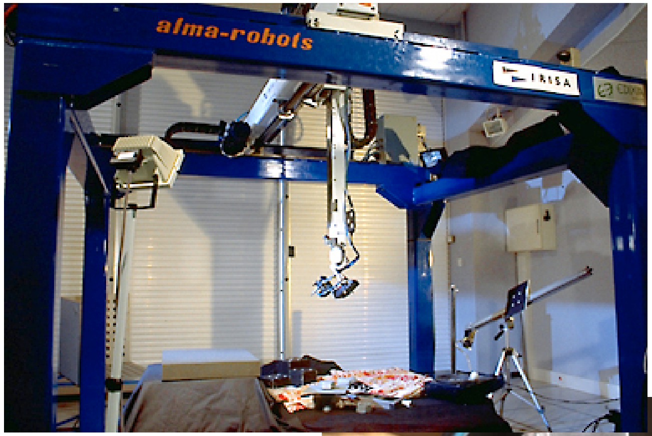
3D	2 ½ D
$\mathbf{s} = \begin{bmatrix} {}^{c^*}\mathbf{t}_c \\ \theta u \end{bmatrix}$	$\mathbf{s} = \begin{bmatrix} x \\ y \\ \log\left(Z/Z^*\right) \\ \theta u \end{bmatrix}$
$\mathbf{s}^* = [0]$	$\mathbf{s}^* = \begin{bmatrix} x^* \\ y^* \\ 0 \\ \mathbf{0} \end{bmatrix}$
$e = (\mathbf{s} - \mathbf{s}^*)$	

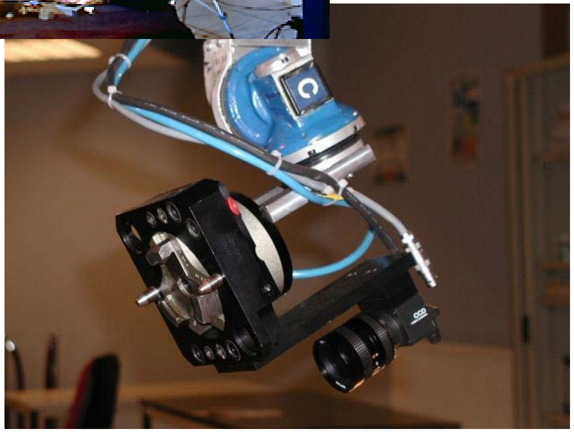
3D	2 ½ D
$\mathbf{L}_s = \begin{pmatrix} {}^{c^*}\mathbf{R}_c & \mathbf{0} \\ \mathbf{0} & \mathbf{L}_w \end{pmatrix}$	$\mathbf{L}_s = \begin{bmatrix} \mathbf{L}_p \\ \mathbf{L}_{\theta u} \end{bmatrix}$ $\mathbf{L}_p = \begin{bmatrix} -1/Z & 0 & x/Z & xy & -(1+x^2) & y \\ 0 & -1/Z & y/Z & (1+y^2) & -xy & -x \\ 0 & 0 & -1/Z & -y & x & 0 \end{bmatrix}$ $\mathbf{L}_{\theta u} = [\ 0\ \ \mathbf{L}_w]$
$L_w = I_3 - (\theta/2)\ [u]_x + (1 - \mathrm{sinc}\theta / \mathrm{sinc}^2(\theta/2))\ [u]_x^2$	

Joint Velocity	$dq = -\lambda * J_s^{+} * e$ $J_s = \mathbf{L}_s * {}^{c}V_n * {}^{n}J_n(q)$
Torque	$\Gamma = \hat{M}(q)\bar{q} + N(q, \dot{q}, t)$ $\Gamma = -\lambda \hat{M}\left(\frac{\hat{\partial e}}{\partial q}\right)^{-1} G\left(\mu De + \frac{\hat{\partial e}}{\partial q}\dot{q} + \frac{\hat{\partial e}}{\partial t}\right) + \hat{N} - \hat{M}\left(\frac{\hat{\partial e}}{\partial q}\right)^{-1} f$

VIMANCO Validation

- Use of internal facilities
- Eurobot Testbed
- Eurobot WET Model at TAS-I premises
- Eurobot Ground Prototype at ESTEC

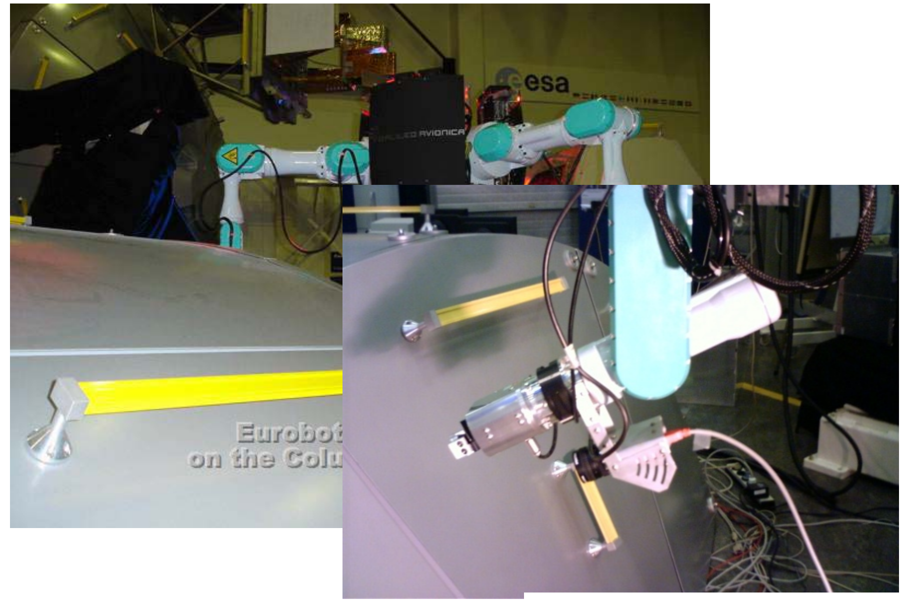




VIMANCO Validation

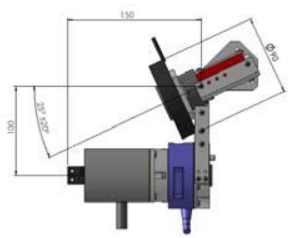
- Use of internal facilities

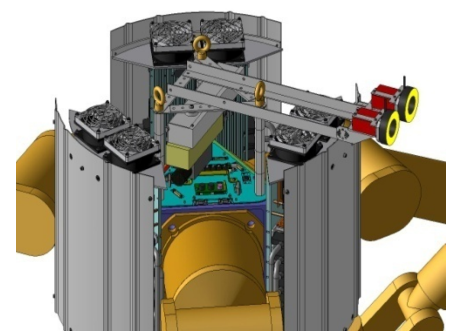
- Eurobot Testbed



- Eurobot WET Model at TAS-I premises

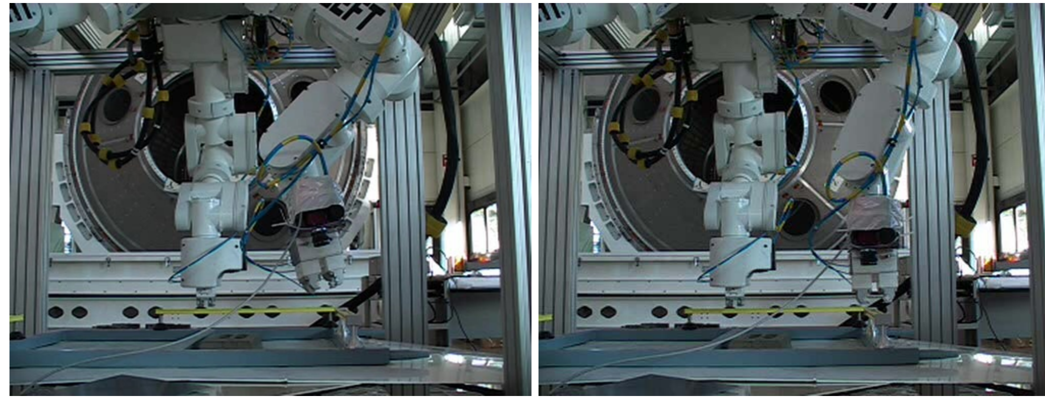
- Eurobot Ground Prototype at ESTEC

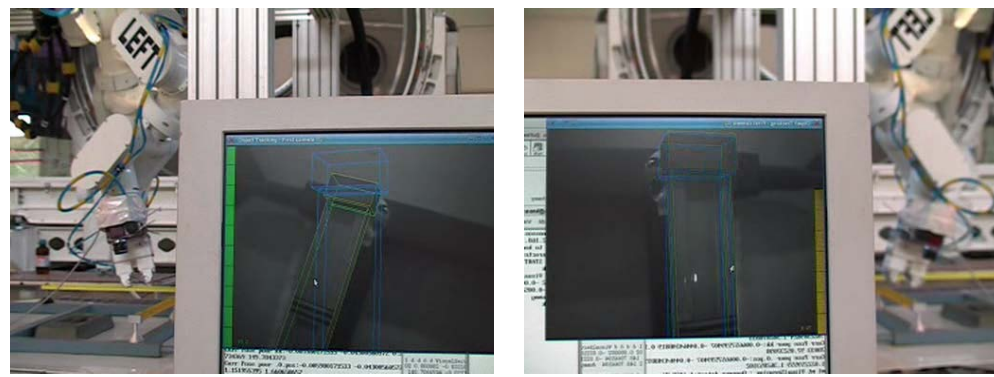




VIMANCO Validation

- Use of internal facilities
- Eurobot Testbed
- Eurobot WET Model at TAS-I premises
- Eurobot Ground Prototype at ESTEC





Simulation Environment

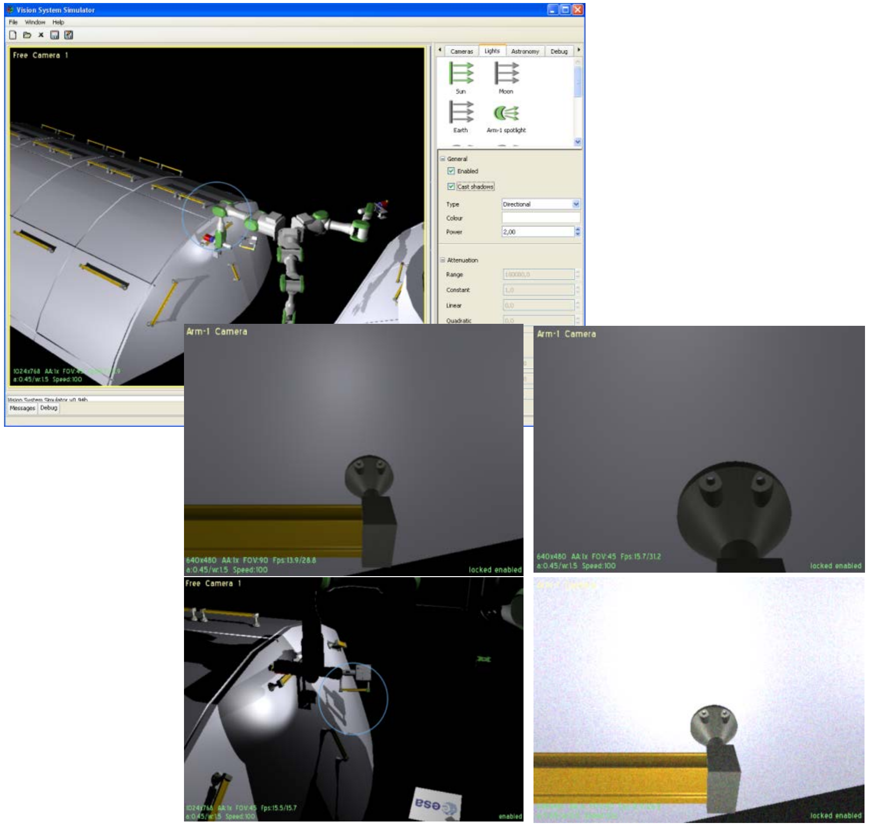
Design / Modeling

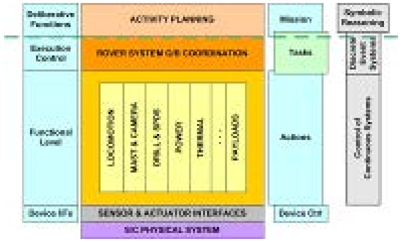
Robotic System
- Mechanical
- Thermal
- Power

Environment
- Atmosphere
- Orbiter & Timekeeping
- Terrain

Generic Controller
- Actions
- Tasks
- Real-time impl.

VIMANCO





TRASYS
WE GET IT DONE

3DROV Simulation Environment

Design / Modeling

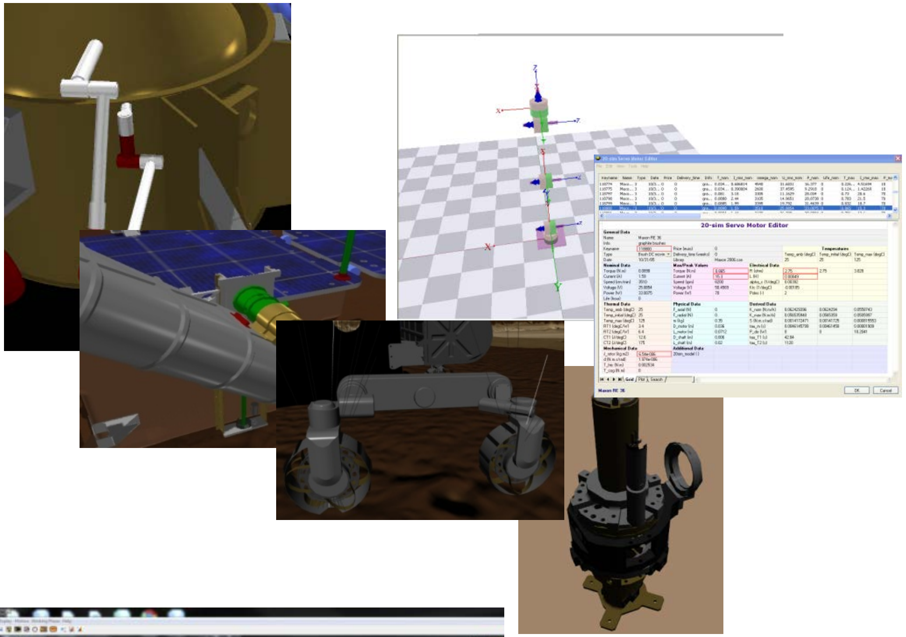
Robotic System
- Mechanical
- Thermal
- Power

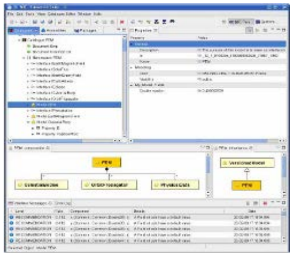
Environment
- Atmosphere
- Orbiter & Timekeeping
- Terrain

Generic Controller
- Actions
- Tasks
- Real-time impl.

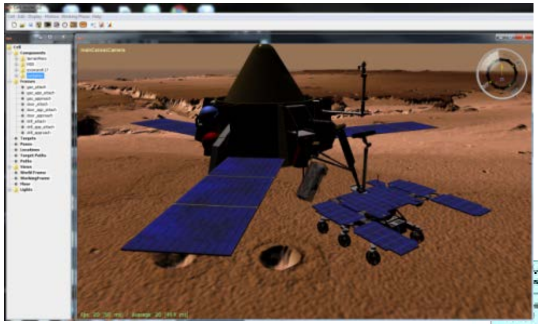
Models

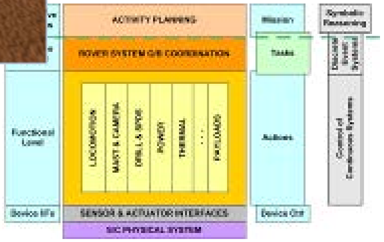
(SMP2 compliant)





Scheduler & Front-end: SIMSAT





TRASYS
WE GET IT DONE

3DROV Simulation Environment

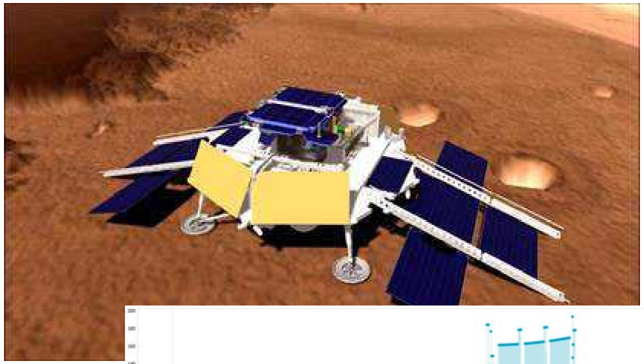
Design / Modeling

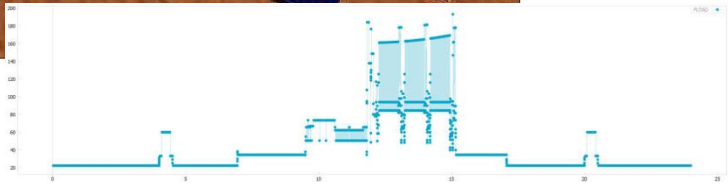
Robotic System
- Mechanical
- Thermal
- Power

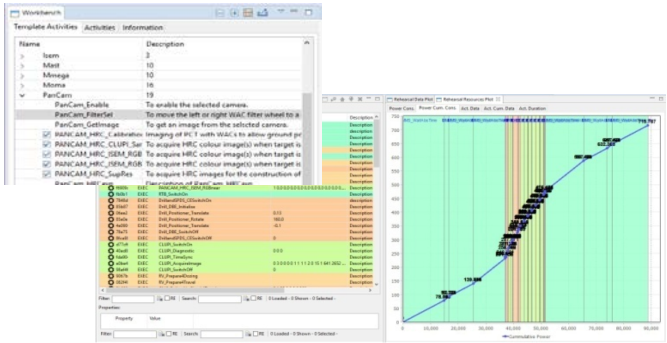
Environment
- Atmosphere
- Orbiter & Timekeeping
- Terrain

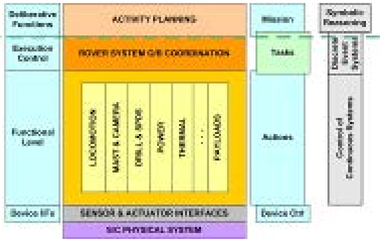
Generic Controller
- Actions
- Tasks
- Real-time impl.

ExoMars - ROSEX









TRASYS
WE GET IT DONE

3DROV Simulation Environment

Design / Modeling

Robotic System

- Mechanical
- Thermal
- Power

Environment

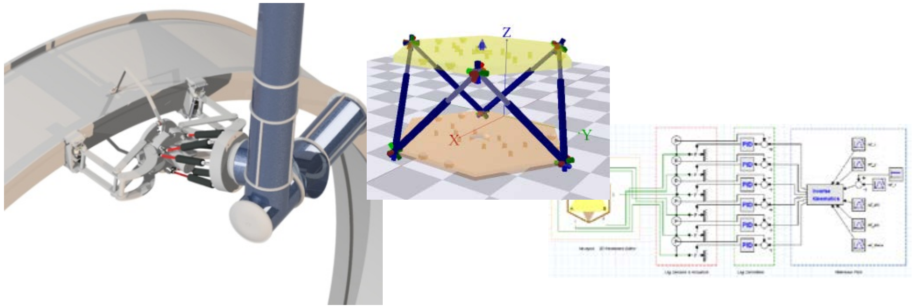
- Atmosphere
- Orbiter & Timekeeping
- Terrain

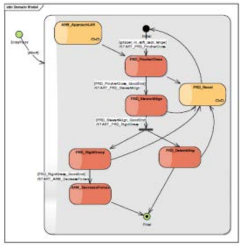
Generic Controller

- Actions
- Tasks
- Real-time impl.

PREDATOR

([104]) PREDATOR: ENVISAT Capturing Strategy using a Stewart platform





ARM_ApproachLAR	The arm moves the Gripper …
PRD_PincherClose	At reception of an external …
PRD_AlignementErrors	With the pinchers in contact …
PRD_StewartAlign	With the pincher softly capturing …
PRD_RigidGrasp	At reception of the event indicating …
PRD_Detumbling	With the gripper rigidly grasping …
ARM_DecreaseForces	While de-tumbling the selected …

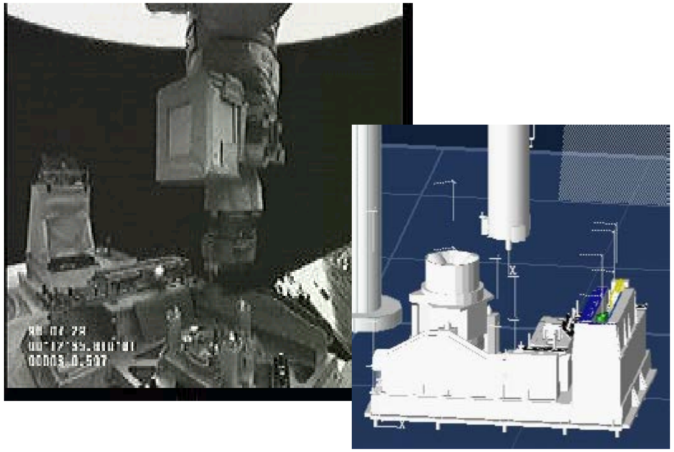
TRASYS
WE GET IT DONE

Ground Control Station

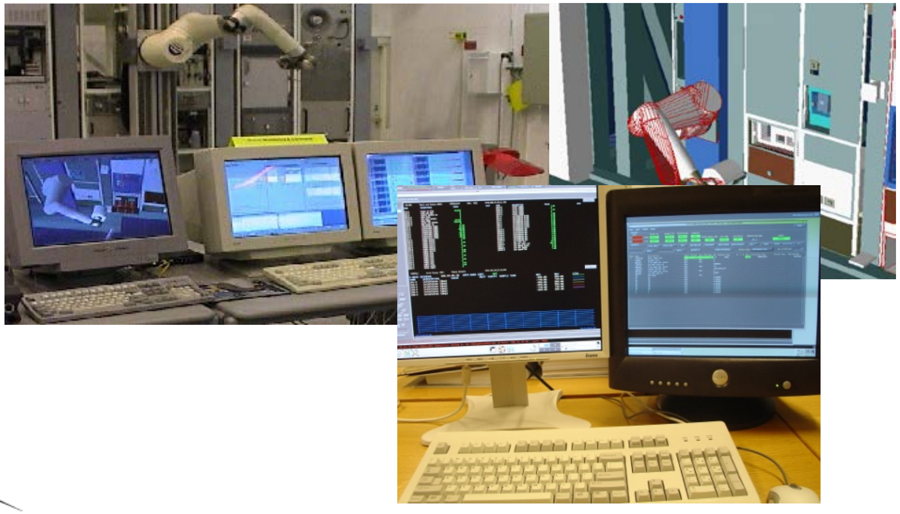
- DREAMS / 3DROCS
  - Prepare and Validate Activities
  - Operate in:
    - Telemanipulation
    - Interactive Autonomy, and
    - Autonomy

    operational modes
  - Situational Awareness
  - Operations assessment

VIABLE



JET / Europa

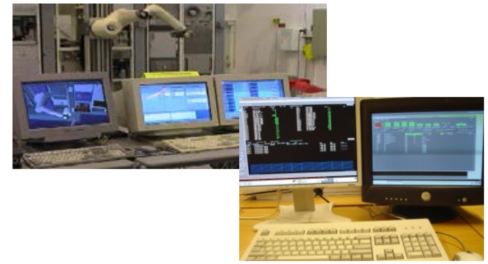


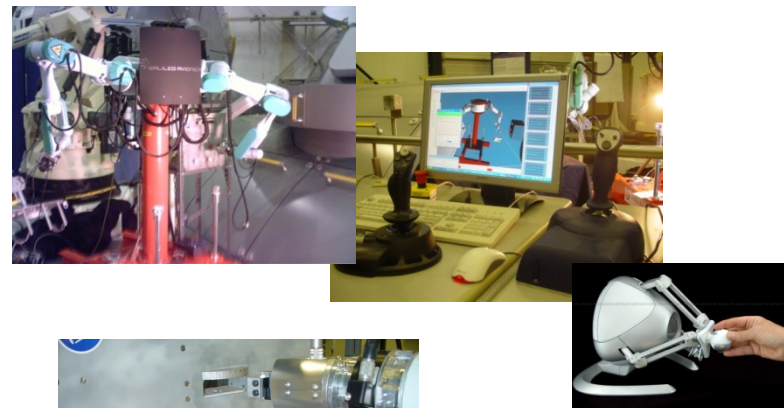
Ground Control Station

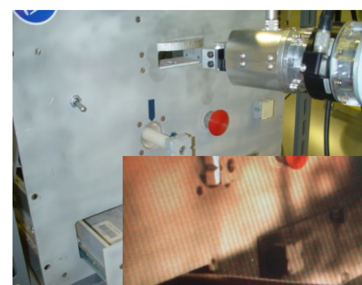
- DREAMS / 3DROCS
  - Prepare and Validate Activities
  - Operate in:
    - Telemanipulation
    - Interactive Autonomy, and
    - Autonomy

    operational modes
  - Situational Awareness
  - Operations assessment

EUROPA







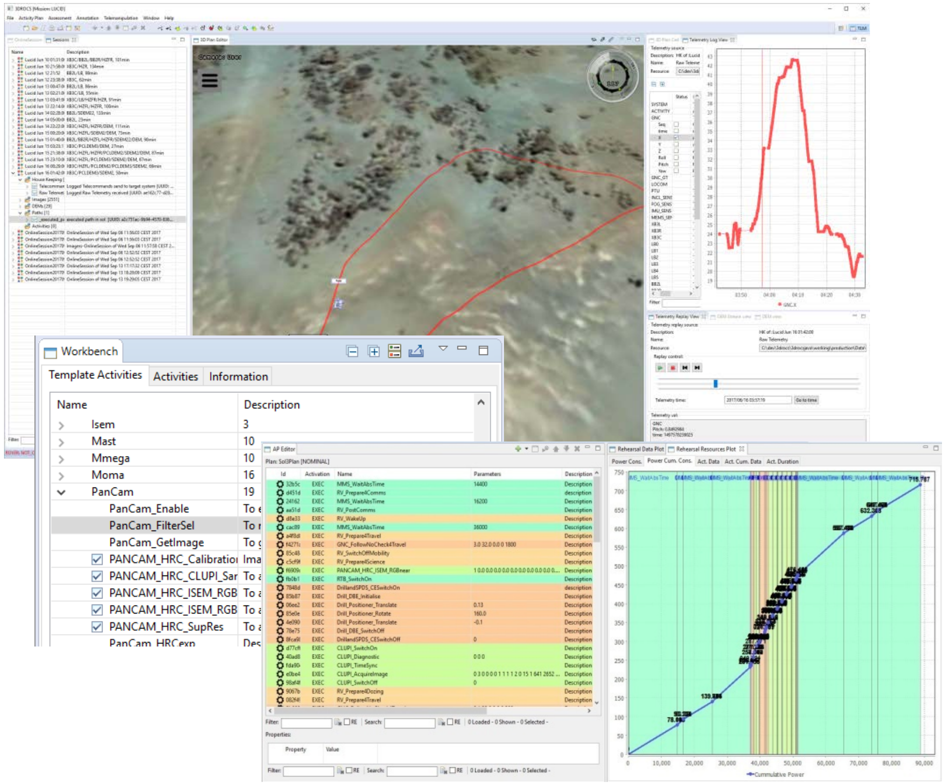
TRASYS
WE GET IT DONE

Ground Control Station

- DREAMS / 3DROCS
  - Prepare and Validate Activities
  - Operate in:
    - Telemanipulation
    - Interactive Autonomy, and
    - Autonomy

    operational modes
  - Situational Awareness
  - Operations assessment

EXOMARS - ROCS



ROCS Milestones:

- CDR: completed

- V2: May 2018

- V3: Dec. 2019

- Launch: 2020

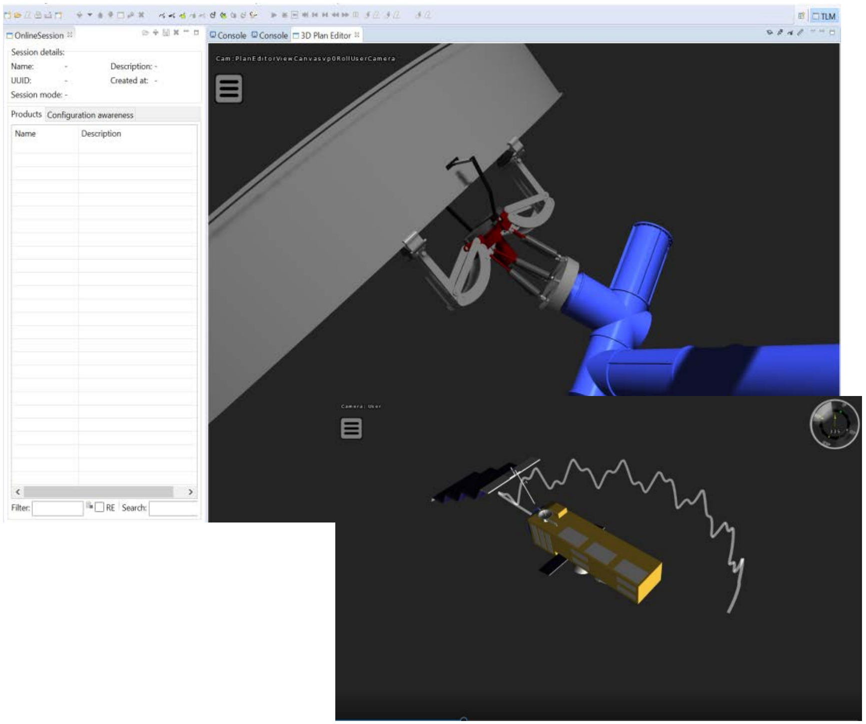
TRASYS
WE GET IT DONE

Ground Control Station

- DREAMS / 3DROCS
  - Prepare and Validate Activities
  - Operate in:
    - Telemanipulation
    - Interactive Autonomy, and
    - Autonomy

    operational modes
  - Situational Awareness
  - Operations assessment

<u>PREDATOR</u>



TRASYS
WE GET IT DONE

Conclusion

- VIMANCO: *VIsion MAnipulation of Non Cooperative Object* is a system allowing
  - Specification
  - Validation via simulation
  - Implementation

  of Vision Based Control for the manipulation of *non-cooperative objects* in space applications

- VIMANCO provides
  - Vision Processing Library
    - Object Recognition
    - Object Tracking
    - Visual Servoing
  - Real-Time Controller / Emulator
  - Vision Simulator
  - Ground Control Station
  - Vision System

- The PREDATOR Activity (HTR/TRASYS/NTUA):
  - Foresees a simulator for the ENVISAT LAR capturing with possibility to include Vision based control
  - Foresees a testbed where Vision based control could be tested