



# Cloud-based Control and vSLAM through Cooperative Mapping and Localization

Berat Alper EROL

Autonomous Control Engineering Lab  
The University of Texas at San Antonio

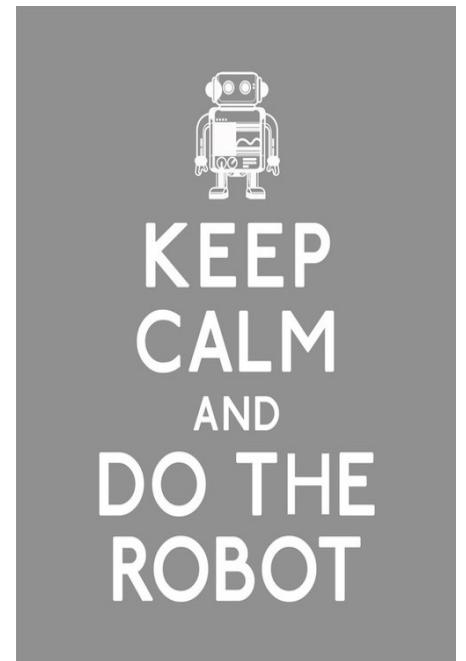
2016





# Outline

- Introduction
- Methodology
- Vision Capabilities
- Case Study
- Conclusion
- Future Works





# Introduction

*"I cannot define a robot, but I know one when I see one."*

Joseph Engelberger

- What is a Robot?
  - There are several definitions one can visualize of a robot in its mind.
- From a scientific perspective, Merriam-Webster's definition:
  - *"machine that looks like a human being and performs various complex acts, as walking or talking, of a human being."*
  - *"a device that automatically performs complicated often repetitive tasks or a mechanism guided by automatic controls."*
- From *robo*ta to robot: forced labor, work; and slavery.



# Introduction

- There would be multiple definitions for the word *robot* based on the environment.
  - An electro-mechanical machine that aims to achieve predefined tasks and future tasks from the past experiences, controlled by an onboard computer with a complex computer program, or remotely controlled from another agent with or without a wire connection.
- Therefore, we can classify the robots:
  - Industrial robots
  - Humanoid robots
  - Mobile robots
  - Service robots
  - Medical Robots

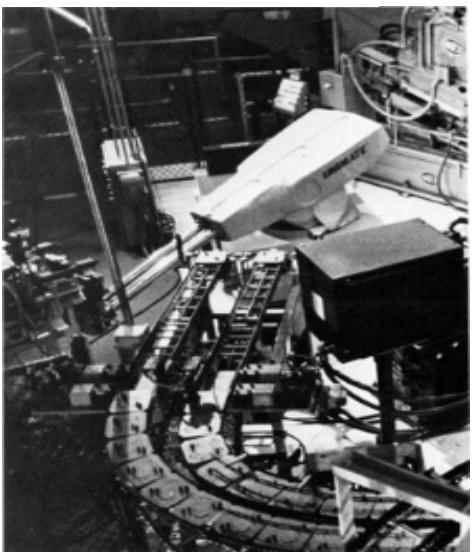


# Introduction

- Industrial robots- manipulators
  - Foundation of robotics.
- Service Robots
  - Domestic robots, assistants and office boys.
- Mobile Robots
  - An automatic electro-mechanical machine with locomotion.
- Humanoid Robots
  - A robot that built into human body shape to perform the tasks using the equipment that developed for humans, in the same environment.



# Introduction



Pictures are from different sources  
For courtesies look for references



# Methodology

- According to International Federation of Robotics (IFR), the number of sales for industrial robots, “*...more than 200,000 industrial robots will be installed in worldwide in 2014, 15% more than in 2013.*”
- Due to human interactions and improved vision capabilities.
- Robots bring a concern about safety.
  - Robots are as intelligent as engineers who designed, built and programmed them. They are as safe as the regulations and requirements that carried by users.



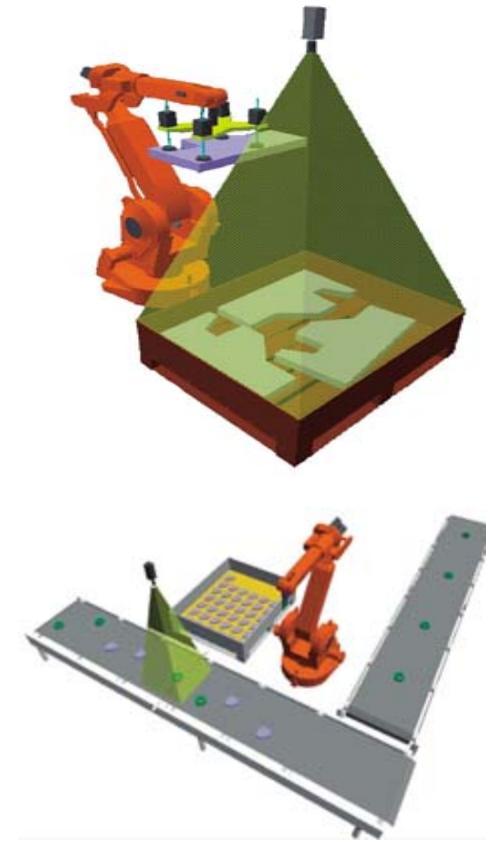
# Methodology

- The most important source of capability: they can sense their environment, and by this ability they can understand and map their surroundings.
- How can they do this?
  - It is simply by using sensors, and cameras.
  - Therefore, it is crucial to keep robots seeing, learning, and especially for mobile robots mapping the workspace.
  - Then, they can easily response to the dynamics, and adapt themselves.



# Methodology

- Watching the work environment and giving rapid response to the dynamic changes.
  - Off-side and integrated cameras for surveillance. Mostly in industrial production and assembly lines, pick and place solutions.
- Integrate a camera; then, processing the data for mapping the environment.



Courtesy of Machine Vision News



Courtesy of ACE Lab



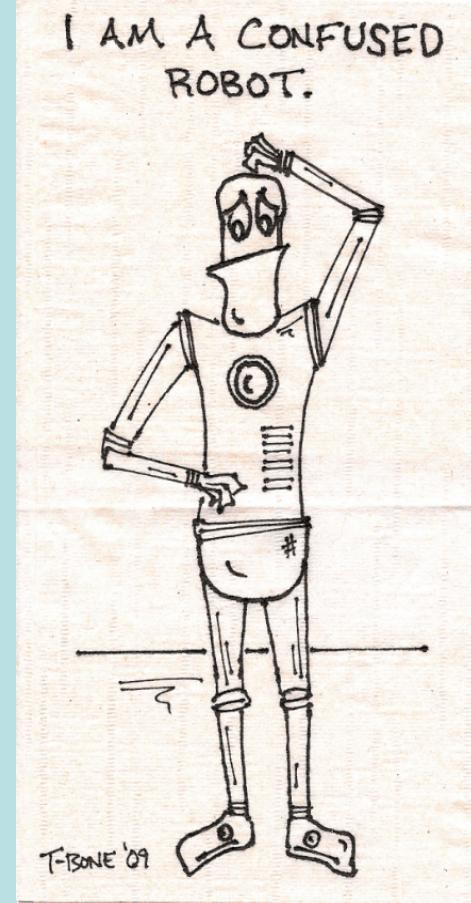
# Methodology

- Localization is a problem for any robotic system performing any autonomous operation.
  - Requires a system to understand the work environment, obstacles around the system, and memory to map the environment.
- This problem is defined in the literature as the chicken and egg problem because calculating the location requires a map, and to map the area the system should know its current location and surroundings.
  - Therefore, a system needs to initialize itself before and during the mapping process as well as localize itself simultaneously.



# Methodology

- SLAM
  - Computational problem of constructing or updating a map of an unknown environment while simultaneously keeping track of an robot's location within it.



- The problem
  - Mapping
  - Localization
- The question:
  - In order to build a map, we must know our position.
  - To determine our position, we need a map.



# Methodology

- Simultaneous Localization and Mapping - SLAM
  - SLAM proposed in early 2000's, and is one of the most popular and applied methods designed for more accurate localization and navigation.
  - The complexity of the problem depends on the type of system and the operation to be performed. Any SLAM problem will have the computational complexity that involves a high power consumption due to the mapping process.
  - The process uses tremendous amount of data gathered by the systems sensors in addition to the power consumed by the system during this process. This limits the operation to very short time iterations, and requires a strong on-board processing power.



# Methodology

There are many ways to solve each of the smaller parts.

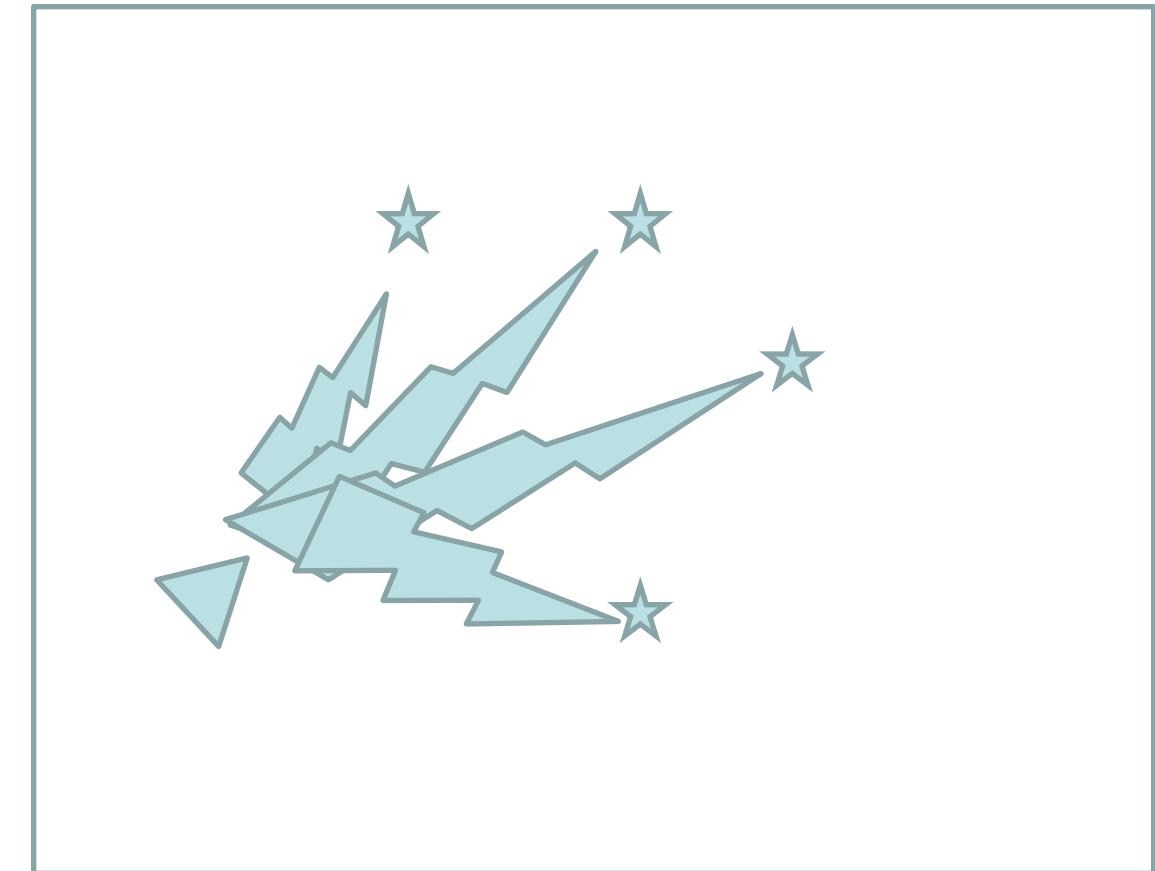
- Landmark extraction
- Data association
- State estimation
- State update
- Landmark update

- The SLAM process consists of number of steps.
  - Use environment to update the position of the robot.
  - Since the odometry of the robot is often erroneous we cannot rely directly on the odometry.
  - We can use laser scans or images of the environment to correct the position.
  - This is accomplished by extracting features from the environment and observing when the robot moves around.



# Simultaneous Localization and Mapping

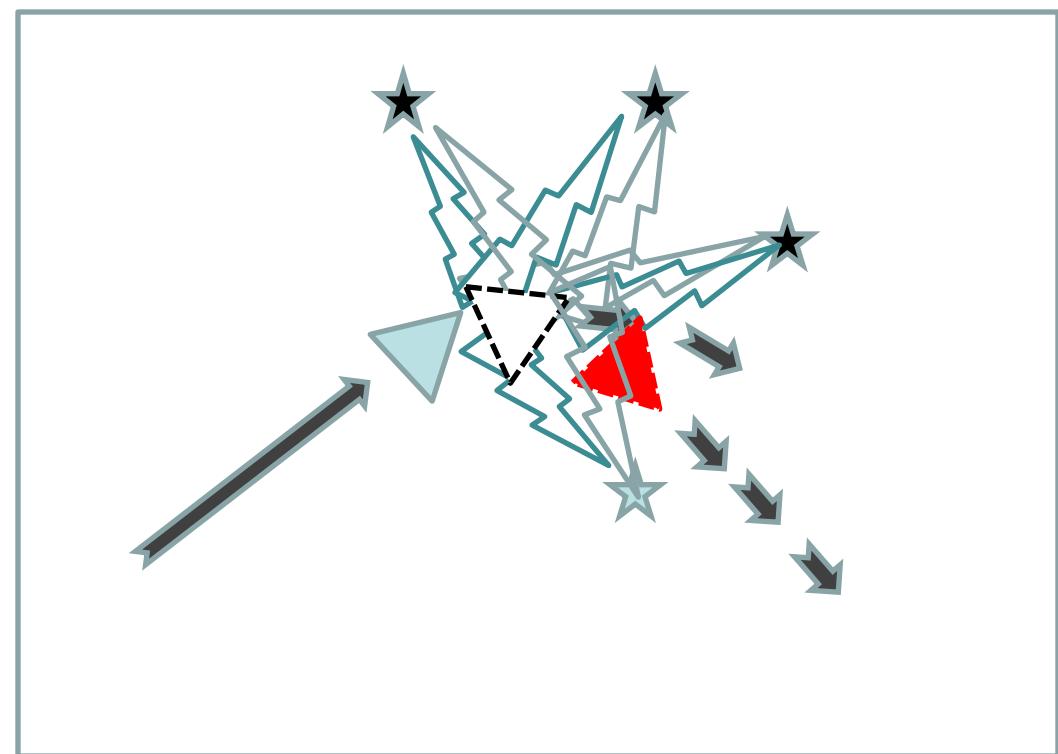
- SLAM
  - Odometry.
  - Laser scanning.
  - EKF
- The goal of the odometry data is to provide an approximate position of the robot.





# Simultaneous Localization and Mapping

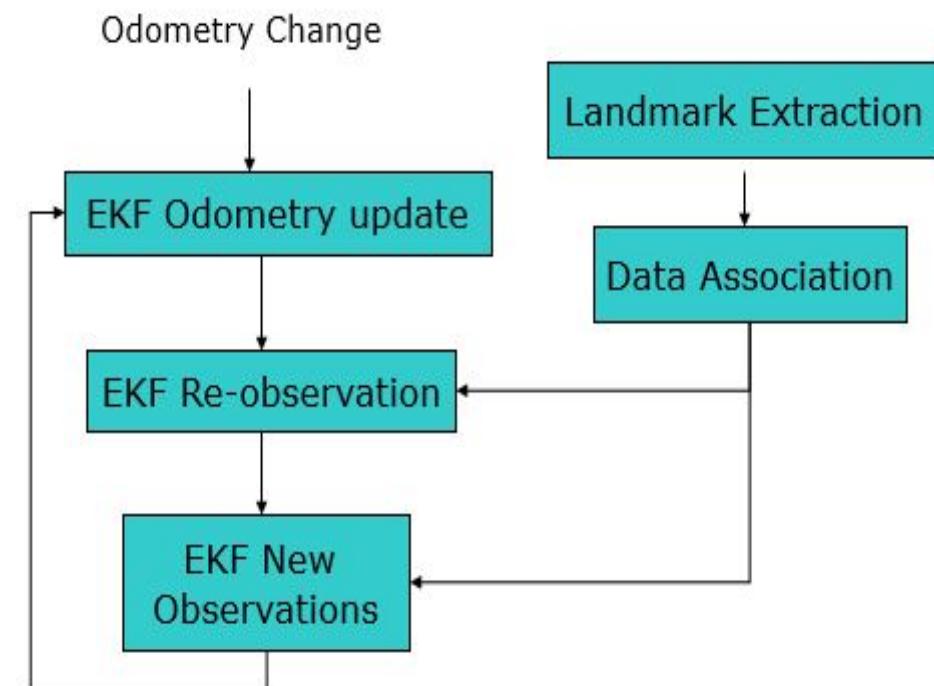
- The goal of the odometry data is to provide an approximate position of the robot.
  - Difficult part about the odometry data and the laser data is the timing.
    - Initial position after starting
    - Further calculation step
    - Final destination





# Simultaneous Localization and Mapping

- The EKF keeps track of an estimate of the uncertainty on the robots position.
- The uncertainty on these landmarks position in the environment.





# Simultaneous Localization and Mapping

- In the literature one can see several implementations on cooperative SLAM that use stereo vision camera systems acquiring the data for fusing the cooperative data.
  - Covariance Intersection (CI) is a method of data fusion which combines two or more states estimates.
  - In addition to this, CI gathers sensor measurements from different platforms having an unknown correlation between them.



# Simultaneous Localization and Mapping



- Any cooperative work for robotic applications that involves swarm of systems requires the systems to communicate their sensor data with each other.
- Cooperative SLAM is one such operation in which multiple systems need to communicate their sensor data with each other in order to build a common map of their surroundings and calculate the locations of the other systems at the same time.



# Simultaneous Localization and Mapping



- Concerns:
  - If SLAM operations by one system involves high computational complexity and high power consumption, then the level of the computational complexity with multiple systems will be undeniably increased, in parallel to the power consumption.
  - Laser or sonar based data collection.
  - Possible to have another source for gathering the data.



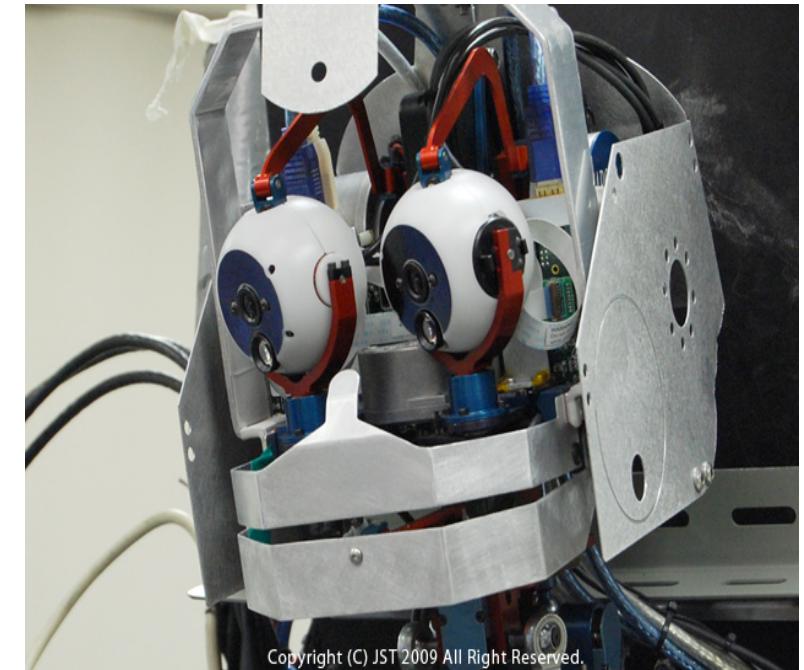
# Vision Capabilities

- Our interaction with the environment requires a system that feels and senses precisely.
- Robot: “*a goal oriented machine that can sense , plan and act.\**”
- Complexity of processing the received data is high.
  - General view with wide angle or focusing on details.
  - Motion detection.
  - Low light conditions.



# Vision Capabilities

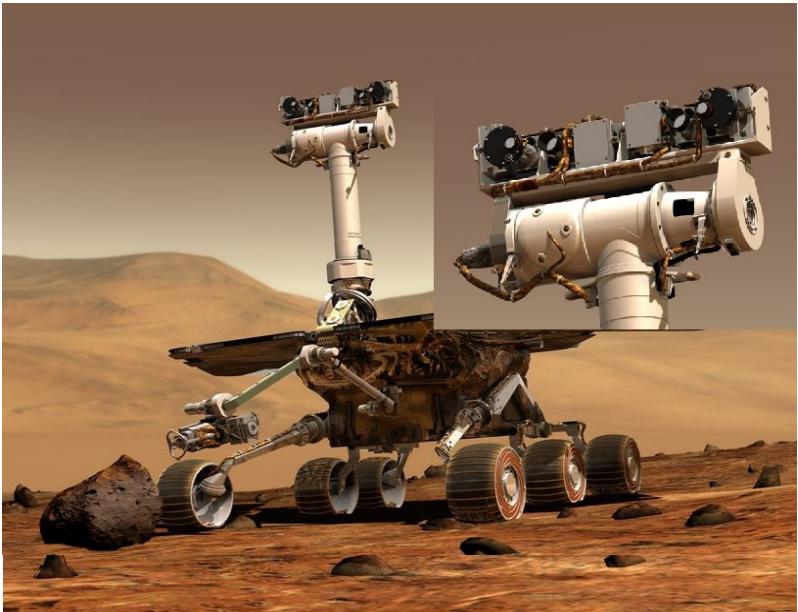
- Classical approaches:
  - Human like vision.
  - High cost, state of art cameras.
  - Wide and detailed.
- Nowadays:
  - Streaming, high processing power.
  - Combined sensory data.
  - Motion detection and visualization.



Courtesy of Japan Science and Technology Agency (JST)



# Vision Capabilities



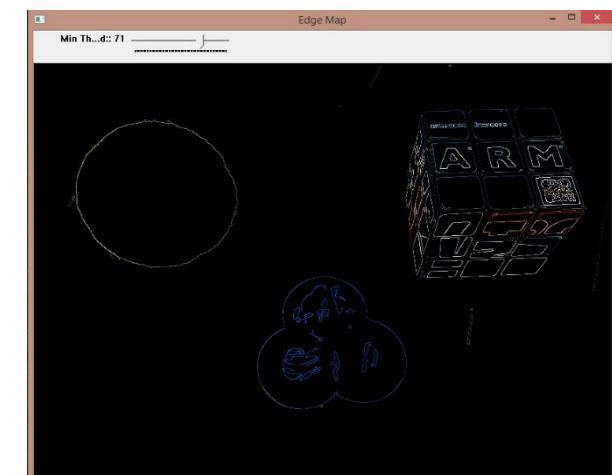
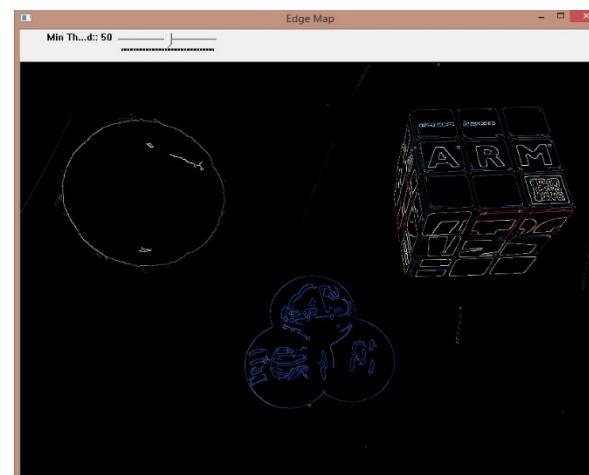
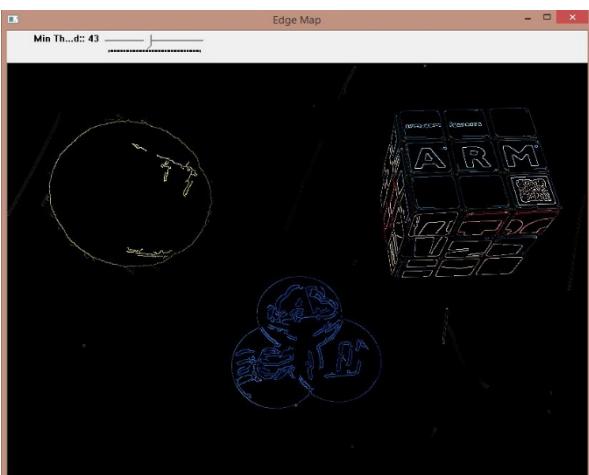
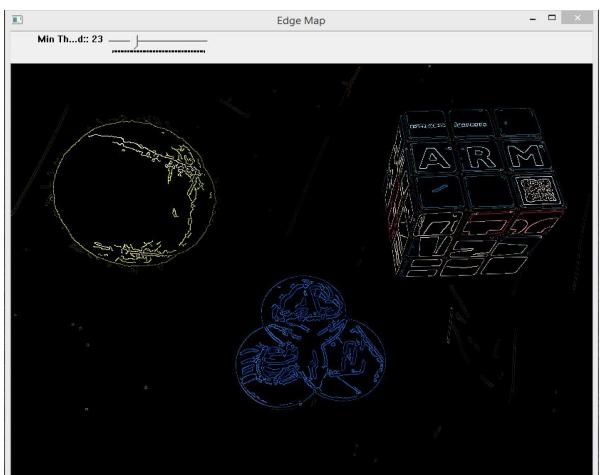
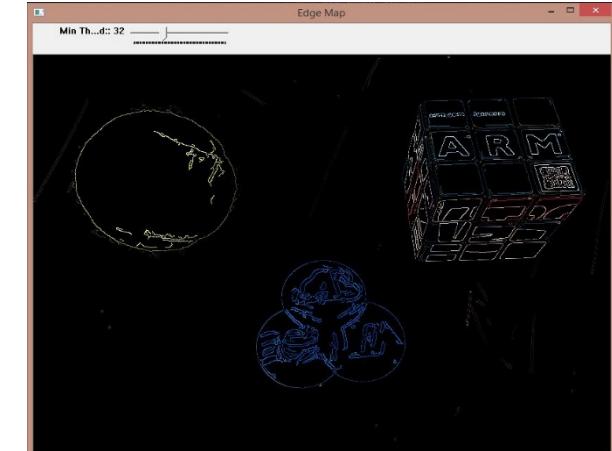
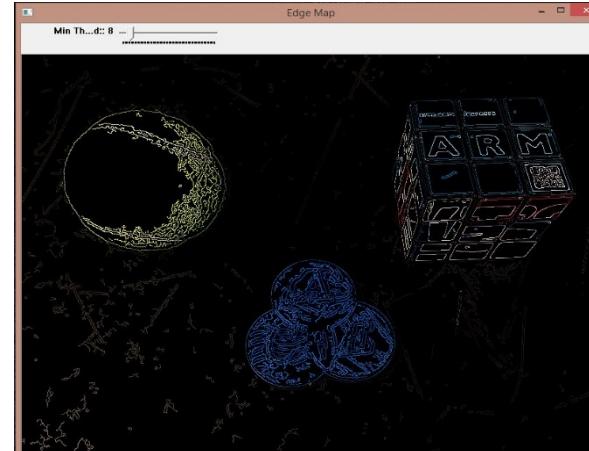
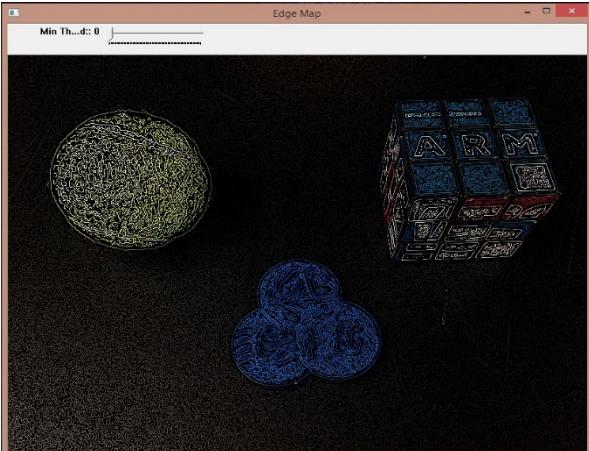
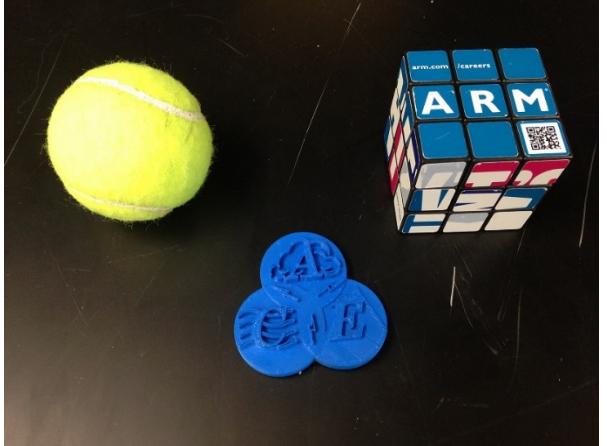


# Vision Capabilities

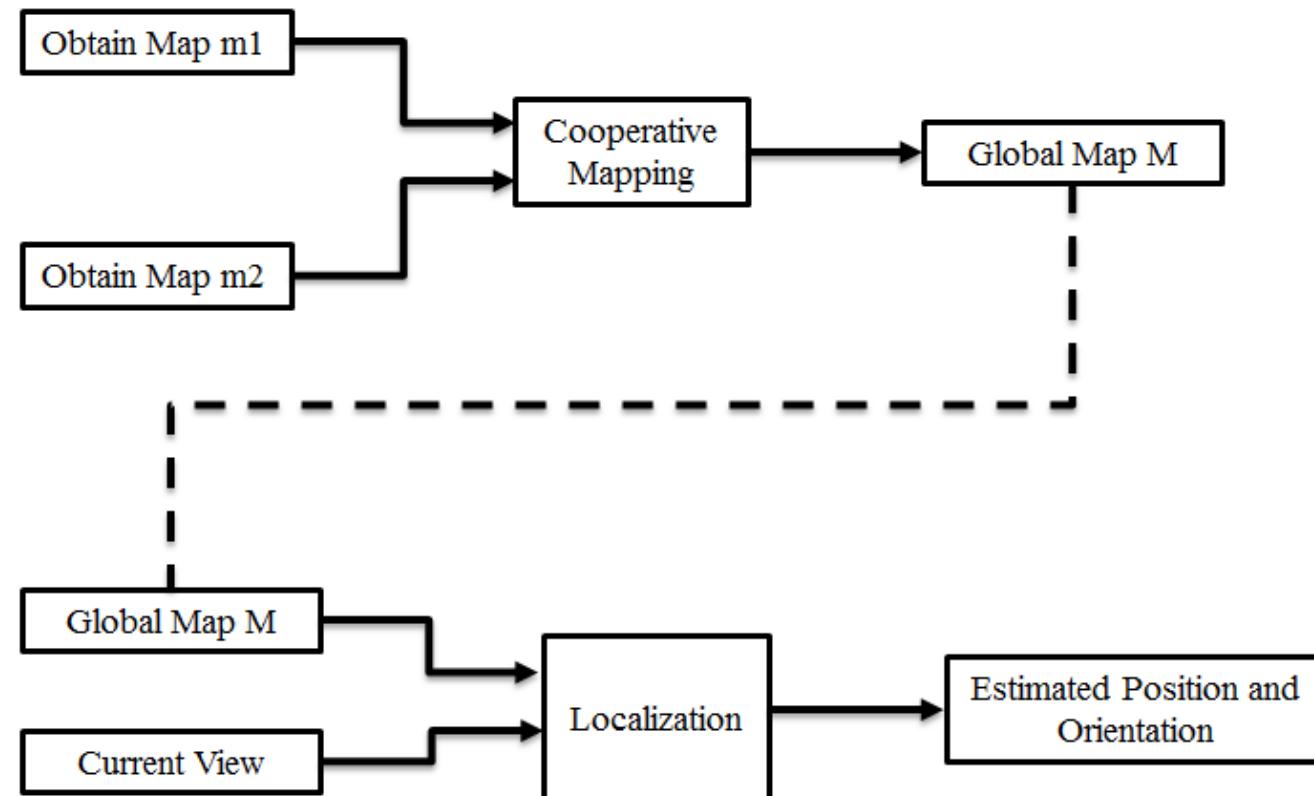
- Microsoft's Kinect, widely used vision source in the robotics. Now we are using ASUS Xtion Pro RGB-D camera for land rovers and quadcopters.
- An RGB camera and a depth sensor with an infrared laser projector, a monochrome CMOS sensor, and for the voice a microphone have implemented.
- Enough components to feature:
  - Capturing motions in 3D.
  - Face/feature recognition.
- The depth scale of the objects is visualized by colors from white to blue and in between, as close to far respectively.



# Vision Capabilities



# Objective of the Presentation





# Vision-based SLAM- visual SLAM (vSLAM)

- Refers to the problem of using images, as the only source of external information, in order to establish the position of a robot, a vehicle, or a moving camera in an environment, and at the same time, construct a representation of the explored zone.





# vSLAM

- vSLAM method will be used in order to detect and identify features in the images that are grabbed by the RGB-D camera.
  - In the past vSLAM has been used in conjunction with the cloud to help agents navigate in their environment.
  - This was done through the use of RG-Chromaticity to process and find features in the environment- was used to inspect each pixel for RGB intensity and match it to images stored in the database.
- This allows robots to remember the features in order to build a map of the world.
  - Previously, this approach was implemented on a Pioneer2 land rover robot and tested in the UTSA building.



# vSLAM

- The most common method for feature detection in SLAM systems has been SIFT, ORB and SURF.
  - These algorithms are widely used for visual odometry, for the building of 3D maps, and to detect objects or points of interest.
- All three of these techniques are useful in their own right for object recognition. The only difficultly with these algorithms is that each of them require a large amount of images to be stored.
  - This could be handled by the cloud by off-loading images and running a SIFT or SURF algorithm on the cloud. This is a method that will be tested once the test bed for the system is created.

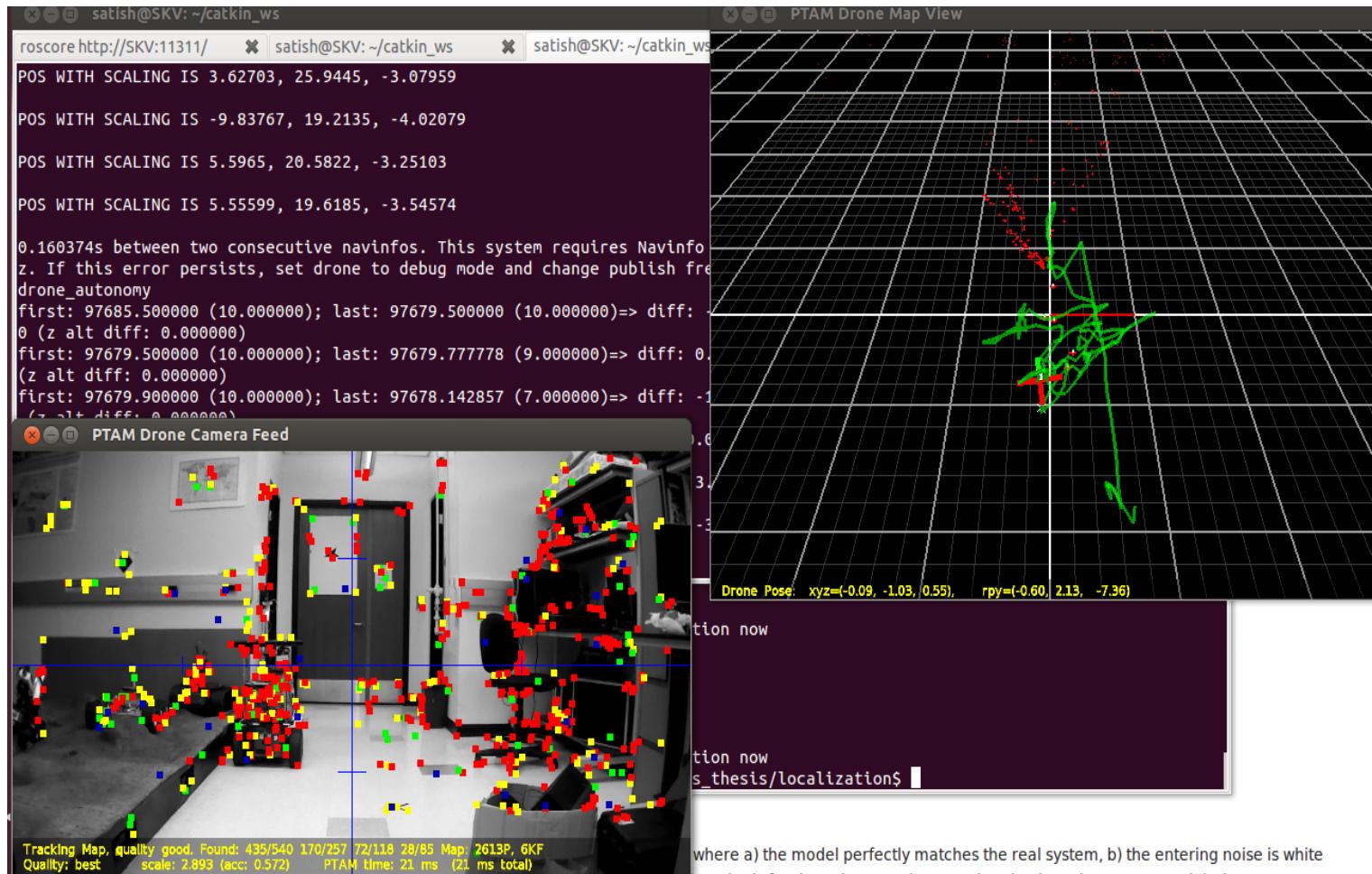


# Vision-based SLAM

- Another method to find features in the mobile systems environment is done through the use of Point Clouds and the Iterative Closest Point (ICP) algorithm.
- Point clouds allow for systems to use depth information from the RGB-D data stream that the sensor returns.
- The ICP method is meant to find the smallest distance between two points in a point cloud. This can be done by generating a point cloud and then comparing the individual points.

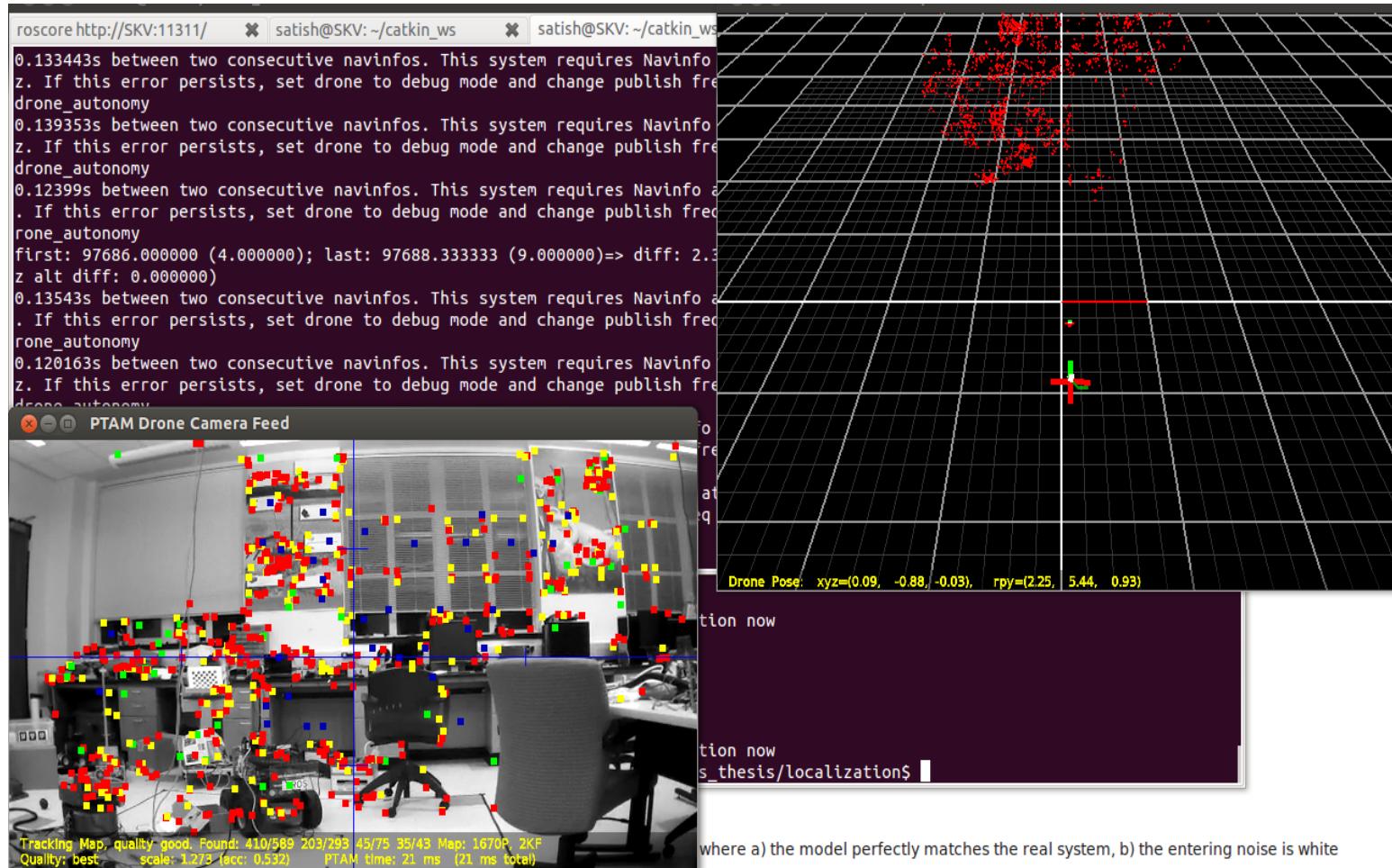


# A Case Study @ACE Labs



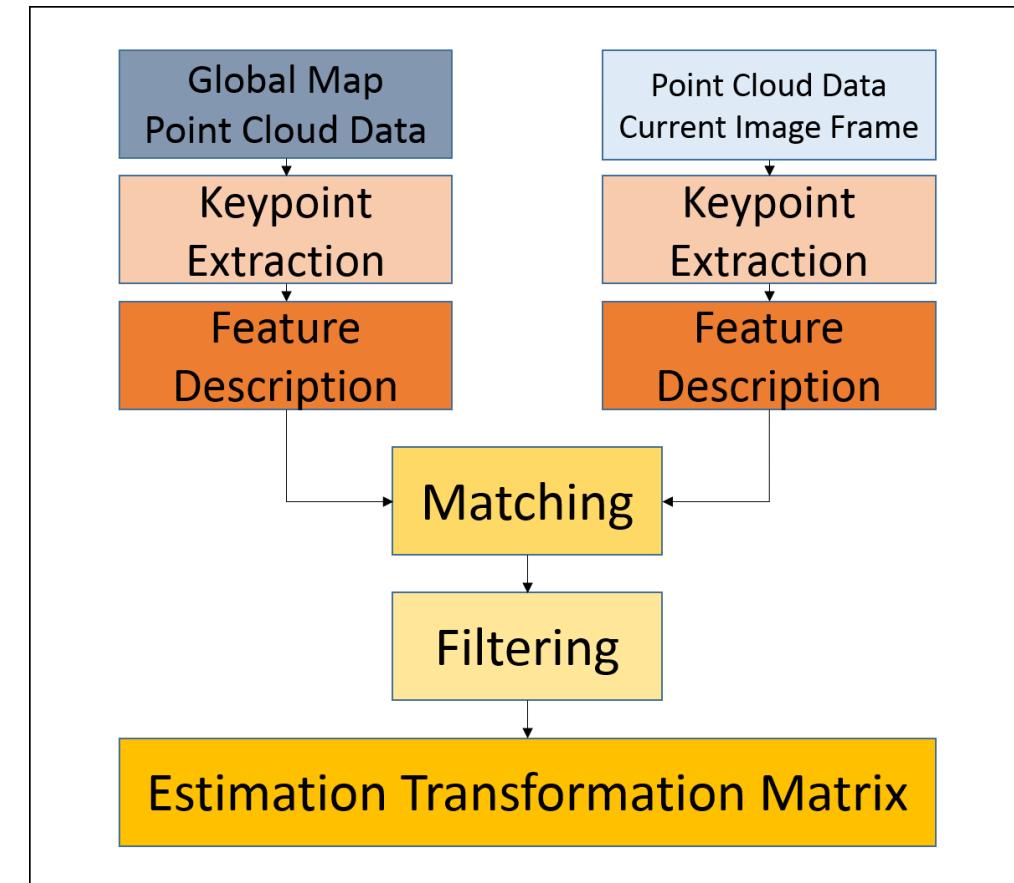
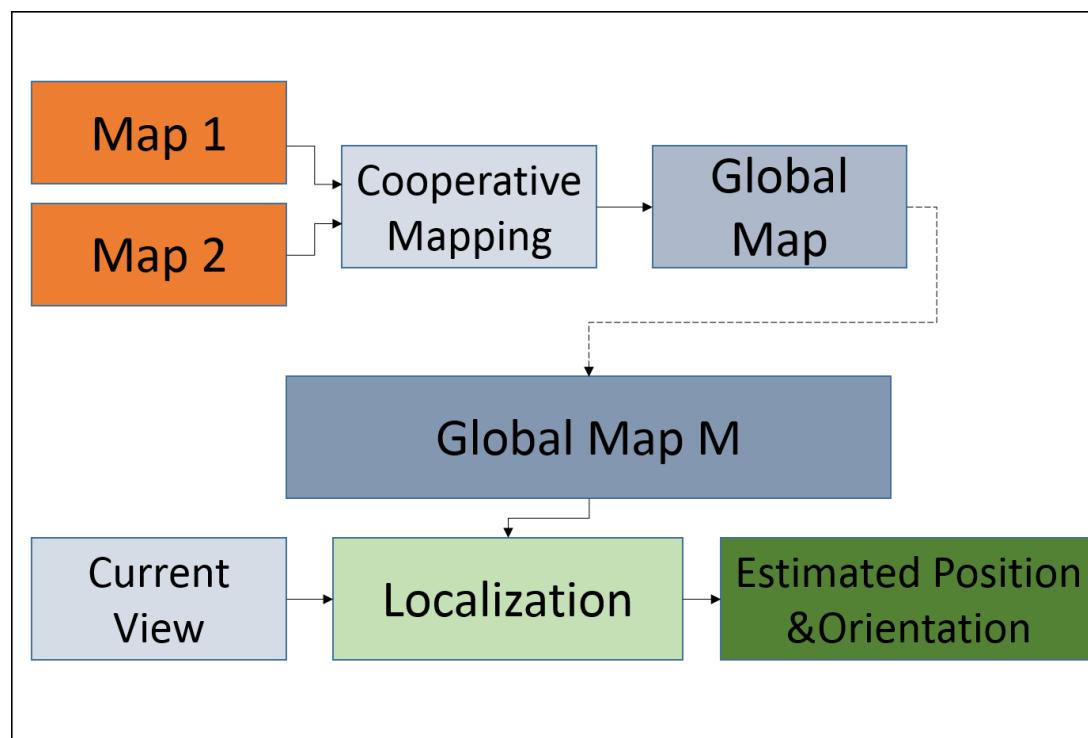


# A Case Study @ACE Labs





# A Case Study @ACE Labs



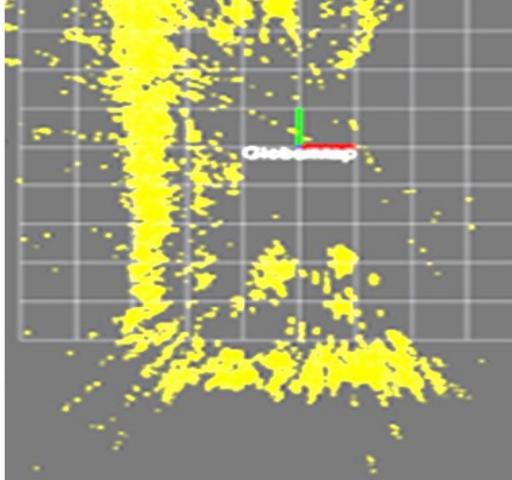
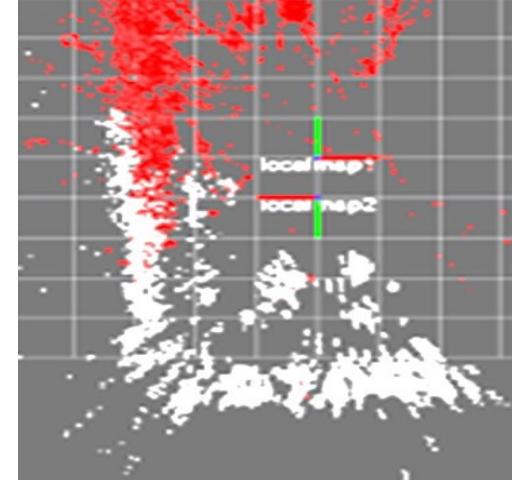
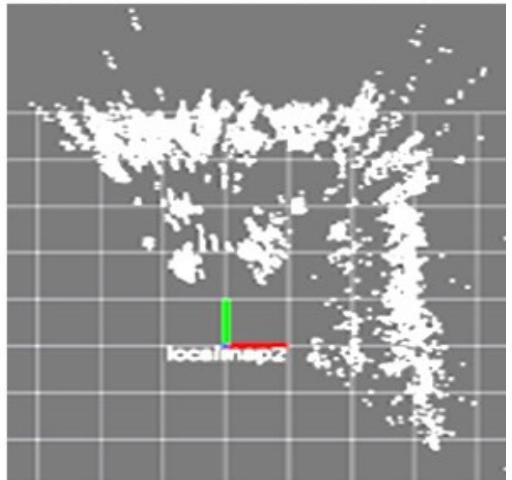
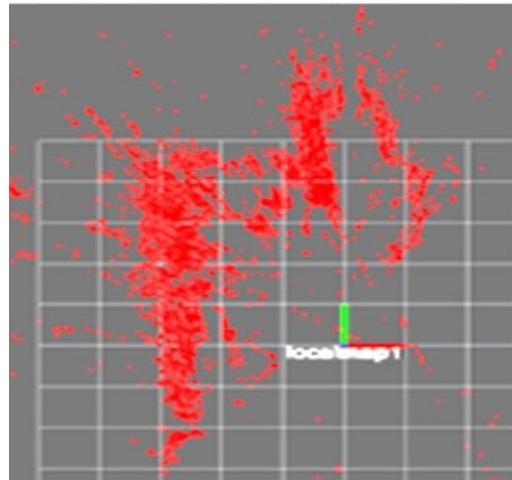
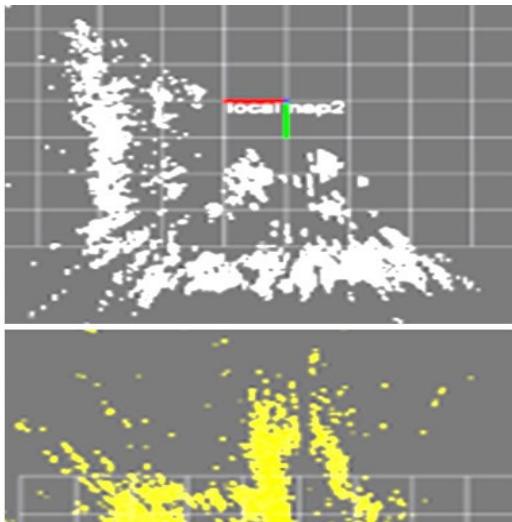
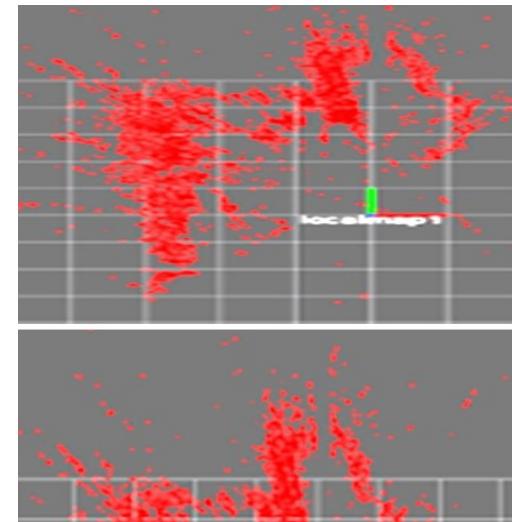
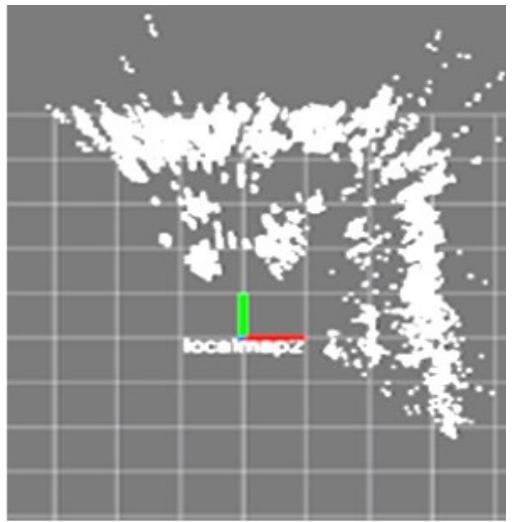
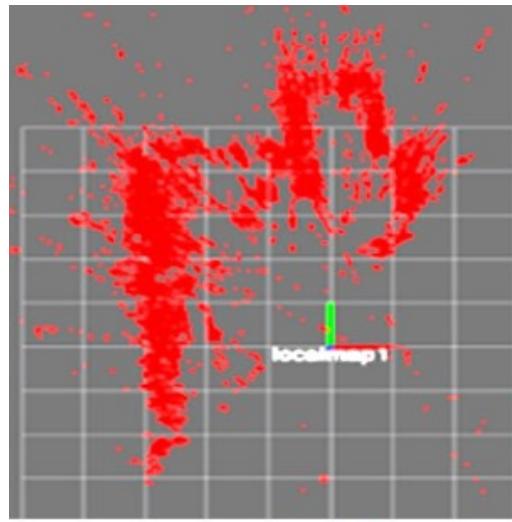


# A Case Study @ACE Labs

- Algorithm:
  - 1: Obtain maps using appropriate ROS package
  - 2: Apply point cloud smoothing on maps to remove the noisy data and make it more organized
  - 3: Define a global map  $M$
  - 4: Apply transformation matrix on maps
  - 5: Use the map forming to get  $M$
  - 6: Apply point cloud registration including transformations to obtain the merged map
  - 7: Apply point cloud registration technique to merge the two local maps into a global map

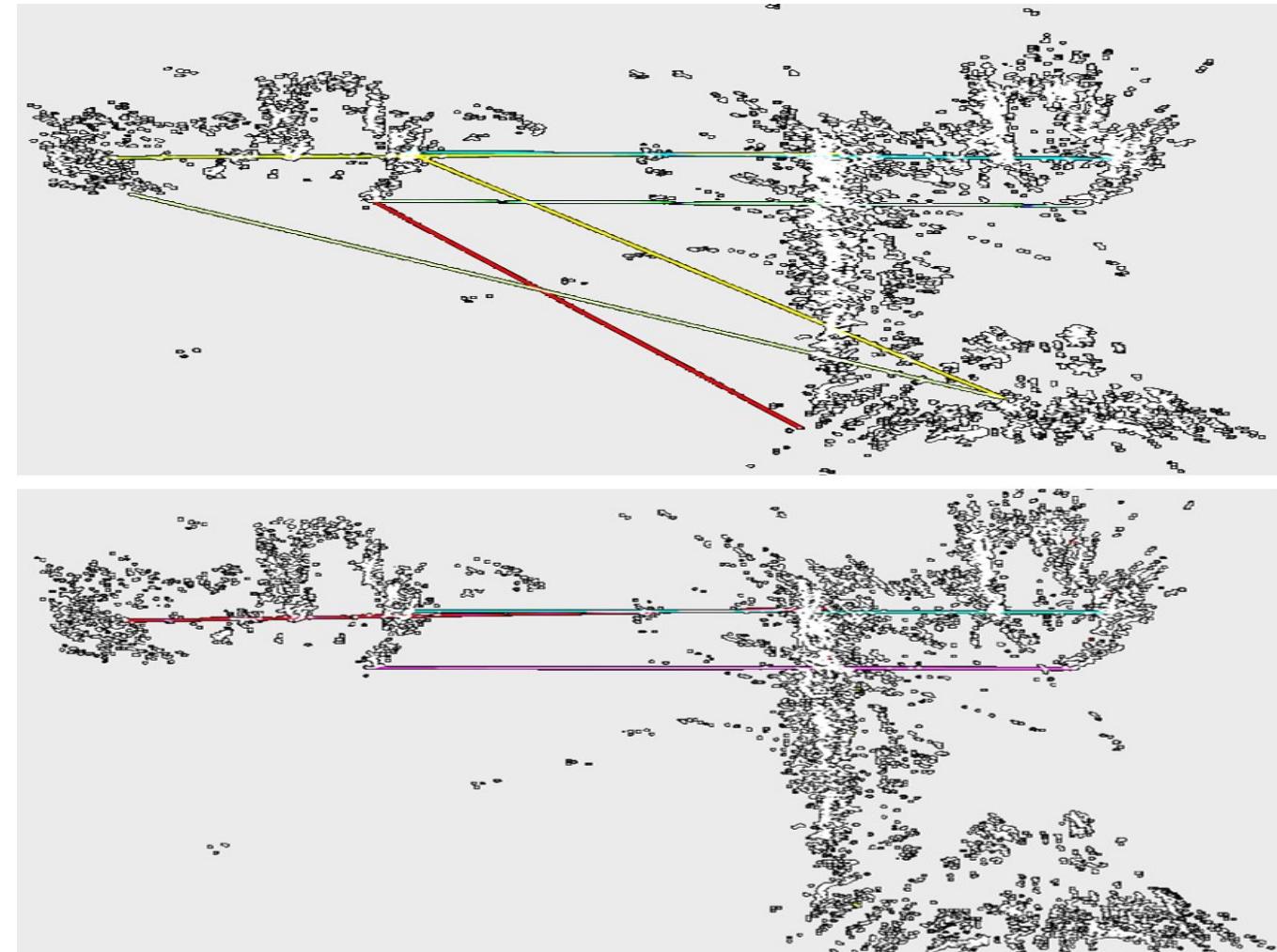
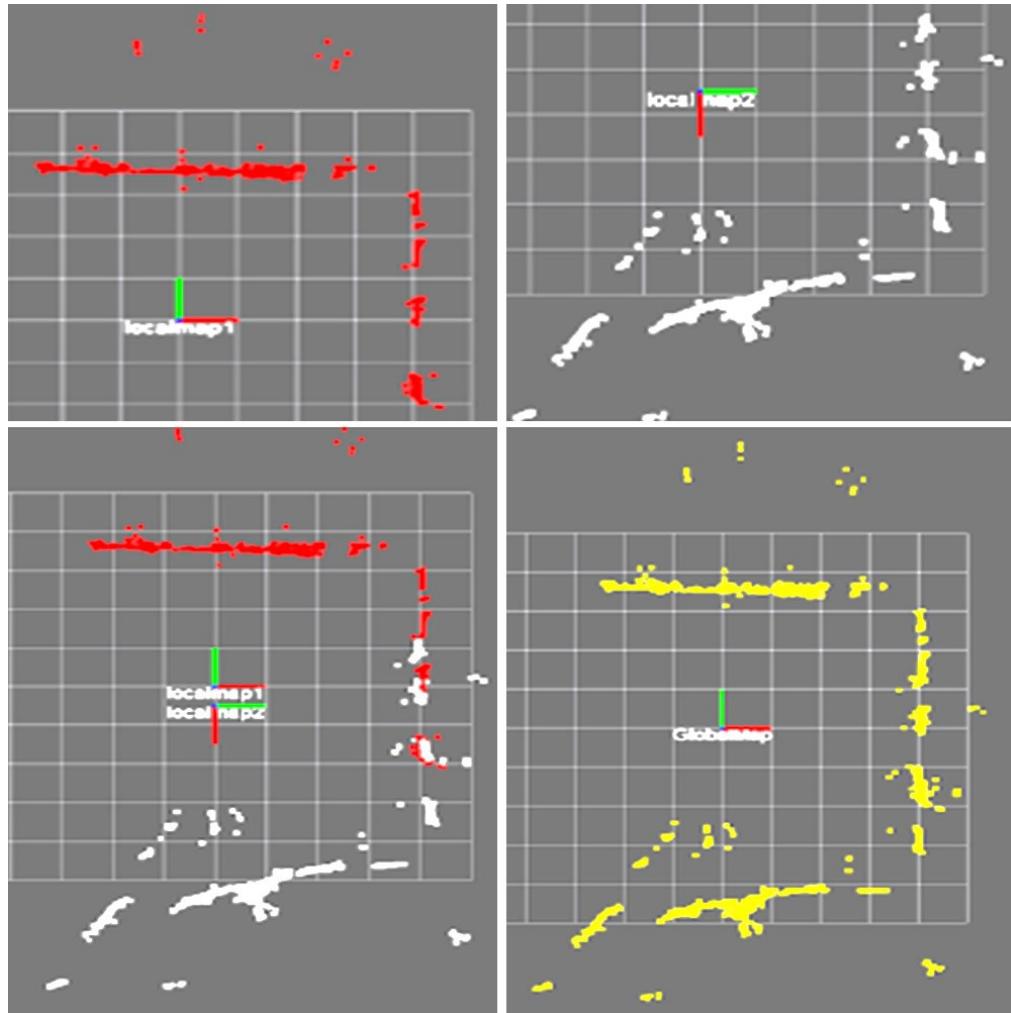


# A Case Study @ACE Labs

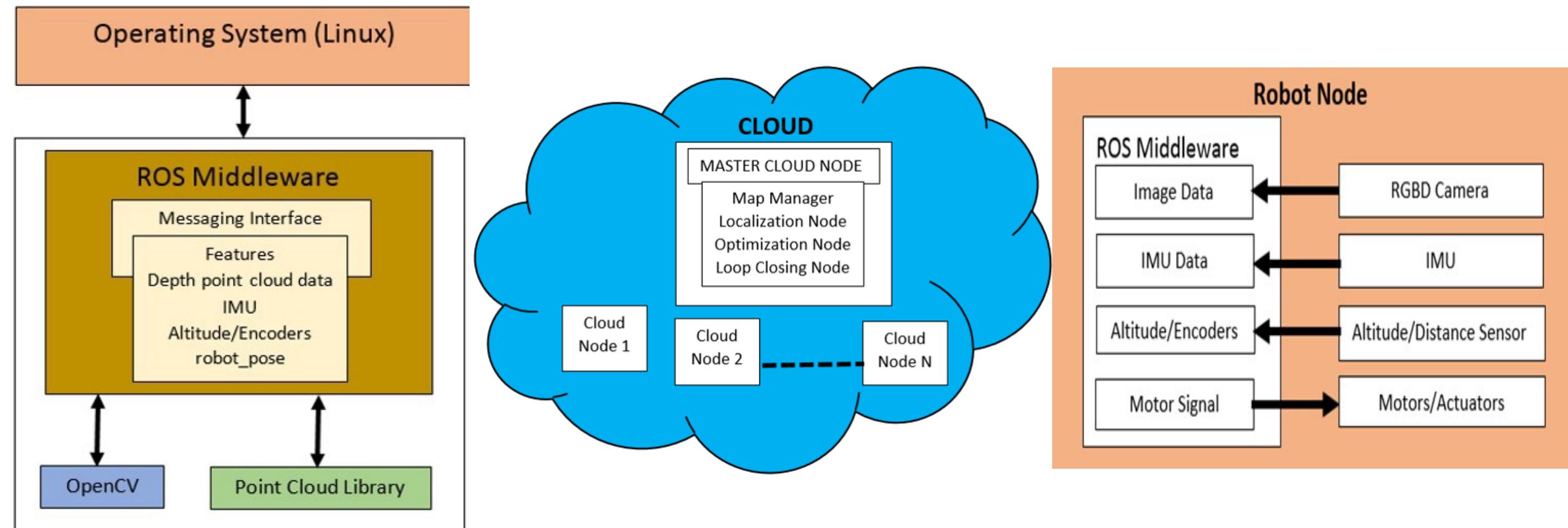




# A Case Study @ACE Labs



# A Case Study @ACE Labs- Proposed Architecture





# Proposed Architecture @ACE Labs

- The vSLAM system that we are proposing to design has an ASUS Xtion Pro Live in order to implement a the algorithm and parse images to a cloud node.
- Our system will consist of a Turtlebot2, an RGB-D sensor, and an Odroid equipped with ROS.
  - ROS will allow us to be able to control the Turtlebot2 and process all image(RGB-D) data as well as sensor data.
- This setup will allow TurtleBot2 to use a vSLAM algorithm for navigation.



# Proposed Architecture @ACE Labs

- Next, we will find features for the vSLAM algorithm, and pass any features detected to a program designed for localization so the system can know where it is in its environment.
- Our algorithm will decide if an old feature is the same as a newly detected feature.
- This is important for the system, since it will have to generate a map of the environment for future use.
- This process will be repeated as the system navigates towards the specified goal in order to create a more complete map.



# Proposed Architecture @ACE Labs





# Conclusions

- Priceless experience for hands-on robotics project.
- Interesting research topics are found.
  - Multidisciplinary robotic research topics are on going in literature, which are popular in several fields includes medical and social sciences.
- Hardware and software design methodologies are reviewed.
  - ROS packages for new quadcopters.
  - 3D design, calibration and printing tools.
  - Autodesk Inverter, 123D, Blender and Cura.



# Conclusions

- The work done till now builds the map after gathering the data by performing the data processing operations of smoothing and registration.
- Also, the localization module estimates the location of the quadcopter in the global map built offline.
  - Hence, one of the immediate future work involves building a framework to perform the operation of cooperative mapping and localization simultaneously.
- The next future work will be to test the algorithm of cooperative mapping with a RGB-D sensor like the ASUS Xtion Pro which might signify the importance of the algorithm in a better way with faster map building operations and much better accuracy.
  - Also, the localization operation will be much faster and more accurate.



# Conclusions

- Cooperative SLAM operations involves dealing with high amount of data due to the huge number of point clouds involved in building the Global Map using the local maps obtained from the quadcopters.
- As the mapping area increases, using the cooperative operation becomes more sensible so as to map the area faster and also with better accuracy with data fusion.
- Also, better sensors lead to better accuracy.
  - This implies that a cooperative SLAM operation with better sensors will lead to tremendous amount of point cloud data which will practically involve huge amounts of computation and high processing power.



# Conclusions

- This calls in for the use of Cloud Computing which takes care of the high computation and processing power requirements.
- As shown in the results we were able to localize a UAV as it built a map of its surroundings. Using this method, in conjunction with the cloud localization that has been done in the past, we can build shared maps more efficiently and effectively.



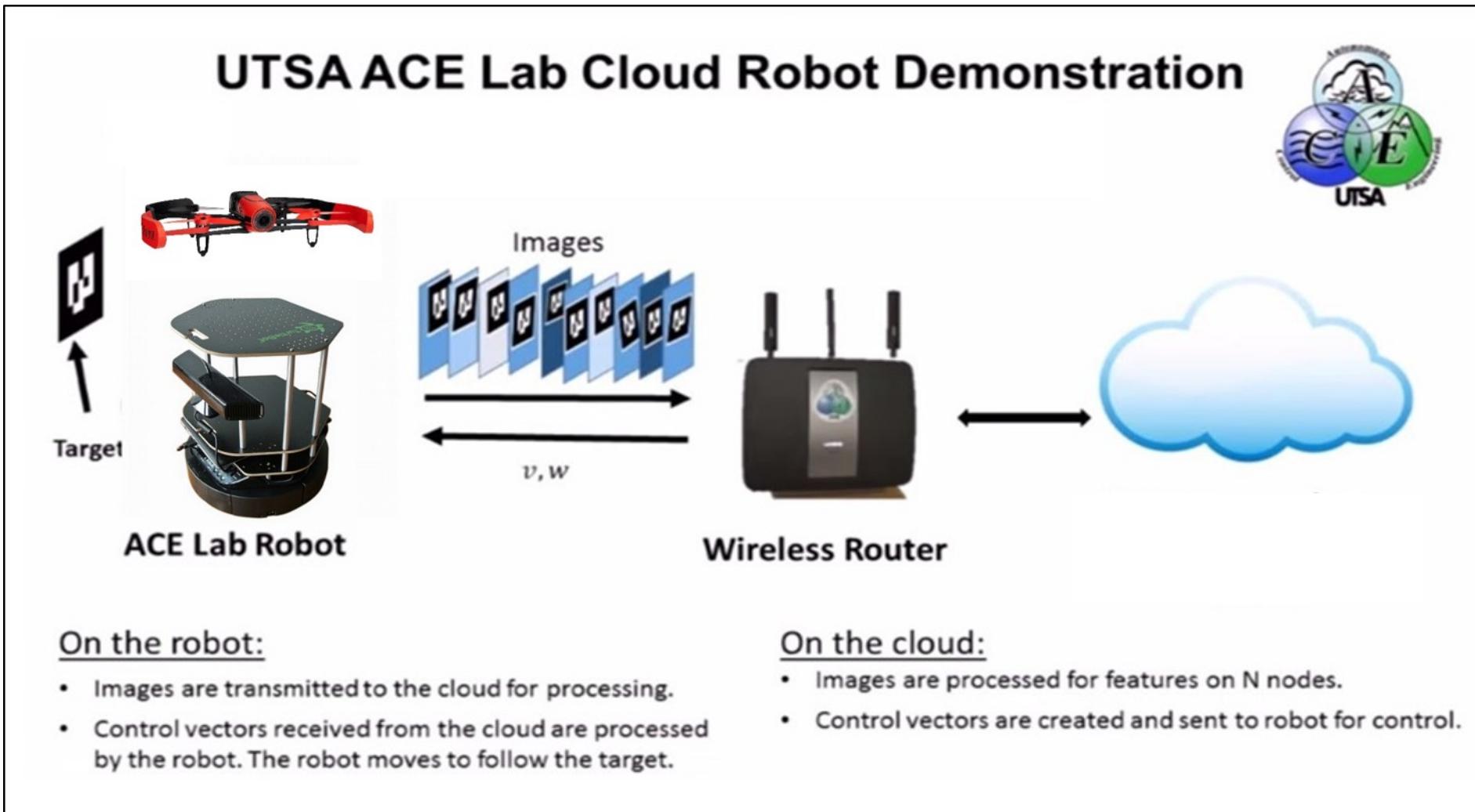
# Future Works

- Improving the vision control and feature detection in cloud back-end.
- Developing a vSLAM library for ACE Labs.
- Using the experience in TECHLAV's objectives.
- Cooperative control in cloud back-end.
- Implementing the previous works into LSASVs.





# Future Works



# References

From journals, conference papers and online sources:

- International Federation of Robotics, statistics. Retrieved March 04, 2015, from <http://www.ifr.org/industrial-robots/statistics/>
- Robot: from Merriam-Webster online dictionary. Retrieved March 04, 2015, from <http://www.merriam-webster.com/dictionary/robot>
- X. Li and N. Aouf, "Experimental research on cooperative vslam for uavs," in Computational Intelligence, Communication Systems and Networks (CICSyN), 2013 Fifth International Conference on. IEEE, 2013, pp. 385–390.
- R. Arumugam, V. R. Enti, L. Bingbing, W. Xiaojun, K. Baskaran, F. F. Kong, K. D. Meng, G. W. Kit et al., "Davinci: A cloud computing framework for service robots," in Robotics and Automation (ICRA), 2010 IEEE International Conference on. IEEE, 2010, pp. 3084–3089.
- F. Endres, J. Hess, N. Engelhard, J. Sturm, D. Cremers, and W. Burgard, "An evaluation of the rgb-d slam system," in Robotics and Automation (ICRA), 2012 IEEE International Conference on. IEEE, 2012, pp. 1691–1696.
- P. Newman and K. Ho, "Slam-loop closing with visually salient features," in Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on. IEEE, 2005, pp. 635–642.
- J. Du, C. Mouser, and W. Sheng, "Design and evaluation of a teleoperated robotic 3-d mapping system using an rgb-d sensor."
- F. Endres, J. Hess, J. Sturm, D. Cremers, and W. Burgard, "3-d mapping with an rgb-d camera," *Robotics*, IEEE Transactions on, vol. 30, no. 1, pp. 177–187, 2014.
- M. Paton and J. Kosecka, "Adaptive rgb-d localization," in Computer and Robot Vision (CRV), 2012 Ninth Conference on. IEEE, 2012, pp. 24–31.
- P. Benavidez, M. Muppudi, P. Rad, J. Prevost, M. Jamshidi, and L. Brown, "Cloud-based realtime robotic visual slam," in Systems Conference (SysCon), 2015 9th Annual IEEE International, April 2015, pp. 773–777.
- G. Bradski, Dr. Dobb's Journal of Software Tools.
- R. B. Rusu and S. Cousins, "3D is here: Point Cloud Library (PCL)," in IEEE International Conference on Robotics and Automation (ICRA), Shanghai, China, May 9-13 2011.
- M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "Ros: an open-source robot operating system," in ICRA workshop on open source software, vol. 3, no. 3.2, 2009, p. 5.
- P. Benavidez, M. Muppudi, and M. Jamshidi, "Improving visual slam algorithms for use in realtime robotic applications," in World Automation Congress (WAC), 2014, Aug 2014, pp. 1–6.
- D. Wang and D. Liu, "Sift-preserving compression of mobile-captured license plate images for recognition," in Wireless Communications and Signal Processing (WCSP), 2014 Sixth International Conference on. IEEE, 2014, pp. 1–5.
- Y. Qin, H. Xu, and H. Chen, "Image feature points matching via improved orb," in Progress in Informatics and Computing (PIC), 2014 International Conference on. IEEE, 2014, pp. 204–208.
- M. Du, J. Wang, J. Li, H. Cao, G. Cui, J. Fang, J. Lv, and X. Chen, "Robot robust object recognition based on fast surf feature matching," in Chinese Automation Congress (CAC), 2013. IEEE, 2013, pp. 581–586.



# Thank You for Your Time

