AR DRONE

On-Board Marker Detection and Following using AR-Drone

Group Members:

Name	Roll Number
Muhammad Hassan Nadeem	2015-10-0063
Danial Nawaz	2015-10-0177

Postgraduate Students

- Mobile Robotics
- Embedded Systems

Undergraduate Students

- Final Year Project
- Microeconomics
- Network Security
- Audio Video
- Mobile Robotics
- Embedded Systems

Mathematical Analysis

- **Postgraduates** have 3*2 = **6 credit hours** worth of course work.
- **Undergraduates** have 3*6 = **18 credit hours** worth of course work.
- Simple Mathematical analysis:
  - *Undergraduates have 18/6 = **3 times the course load**.*
- Conclusion:
  - *For equal quality of work, **Undergraduates should get 3 times the grade points** compared to Postgraduates.*

CROSS-COMPILATION

Porting OpenCV 2.4.10 to AR Drone 2.0

OpenCV in AR Drone

- **Nightmare** to port OpenCV to AR Drone.
- Numerous Problems in
  - *Setting up cross compilation environment*
  - *Linking shared libraries*
  - *FFmpeg support*
  - *etc. etc.*

- Every thing is documented and available on my Blog : blog.HassanNadeem.com

CAPTURING VIDEO STREAM

Video Stream

- Front Camera is 720p (1280x720) , 30 FPS.
- AR Drone sends TCP stream of Video on PORT 5555 encoded in either MPEG or H264.
- We use FFMPEG libraries with x264 to decode the stream for OpenCV.
- Drone configured to stream 360p video at 15 FPS.

AR DRONE API

AT Commands

AT Commands

- AR Drone can receive AT commands on UDP port 5556.
- AT commands are strings encoded as 8-bit ASCII characters.
- You can pass integer, float and string arguments with AT commands.
- Integers are encoded as strings with their decimal representation.
- 32-bit Floating point numbers are encoded as 32-bit integers.
- Example AT Command to move drone forward with 50 % velocity:
  - **AT***PCMD=63,1,0, 1056964608,0,0

AT Commands Summary

- List of Available AT-Commands:

AT command	Arguments[1]	Description
AT*REF	input	Takeoff/Landing/Emergency stop command
AT*PCMD	flag, roll, pitch, gaz, yaw	Move the drone
AT*PCMD_MAG	flag, roll, pitch, gaz, yaw, psi, psi accuracy	Move the drone (with Absolute Control support)
AT*FTRIM	-	Sets the reference for the horizontal plane (must be on ground)
AT*CONFIG	key, value	Configuration of the AR.Drone 2.0
AT*CONFIG_IDS	session, user, application ids	Identifiers for AT*CONFIG commands
AT*COMWDG	-	Reset the communication watchdog
AT*CALIB	device number	Ask the drone to calibrate the magnetometer (must be flying)

AT*PCMD

- **Send progressive commands - makes the drone move (translate/rotate).**
- Argument 1 : the sequence number
- Argument 2 : flag enabling the use of progressive commands and/or the Combined
- Yaw mode (bitfield)
- Argument 3 : drone left-right tilt - floating-point value in range [-1 to 1]
- Argument 4 : drone front-back tilt - floating-point value in range [-1 to 1]
- Argument 5 : drone vertical speed - floating-point value in range [-1 to 1]
- Argument 6 : drone angular speed - floating-point value in range [-1 to 1]

NAVIGATION DATA

What is Navigation Data?

- Information containing drone status, roll, pitch, yaw, altitude, x,y,z velocities etc.
- Sent periodically every 5ms.
- Encoded data sent over UDP port 5554 in binary format.

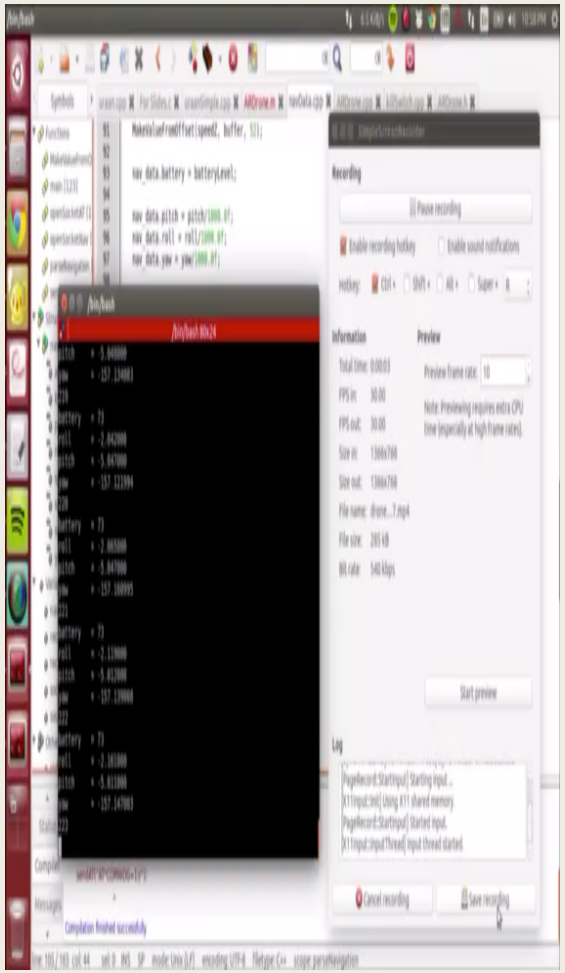
Header 0x55667788	**Drone state**	**Sequence number**	**Vision flag**	**Option 1**			...	**Checksum block**		
				id	size	data	...	cks id	size	cks data
32-bit int.	32-bit int.	32-bit int.	32-bit int.	16-bit int.	16-bit int.	... ...	... ...	16-bit int.	16-bit int.	32-bit int.

Extracting Navigation Data

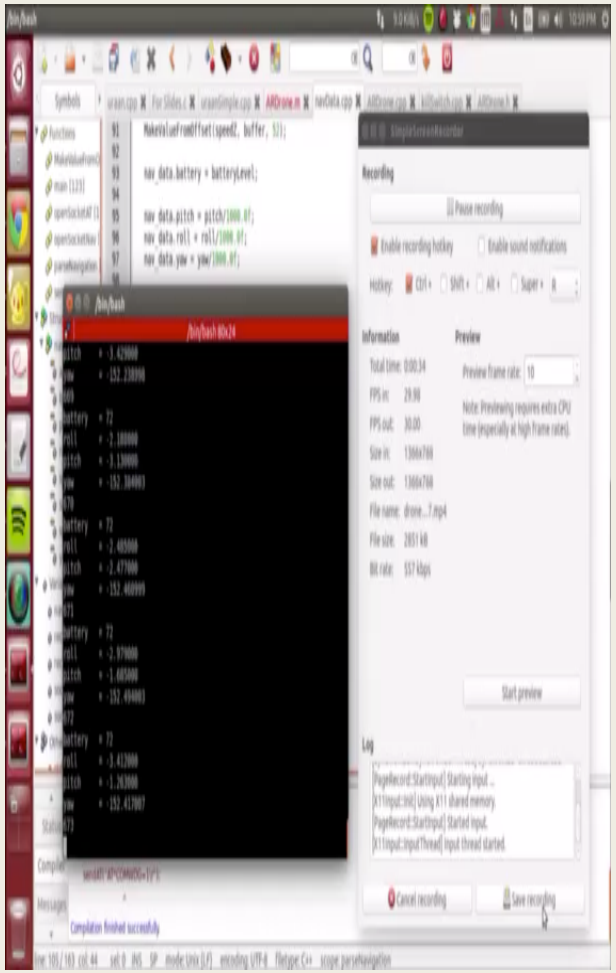
- A dedicated thread running that
  - *Requests*
  - *Receives*
  - *Decodes and*
  - *Publishes*

    Navigation Data.

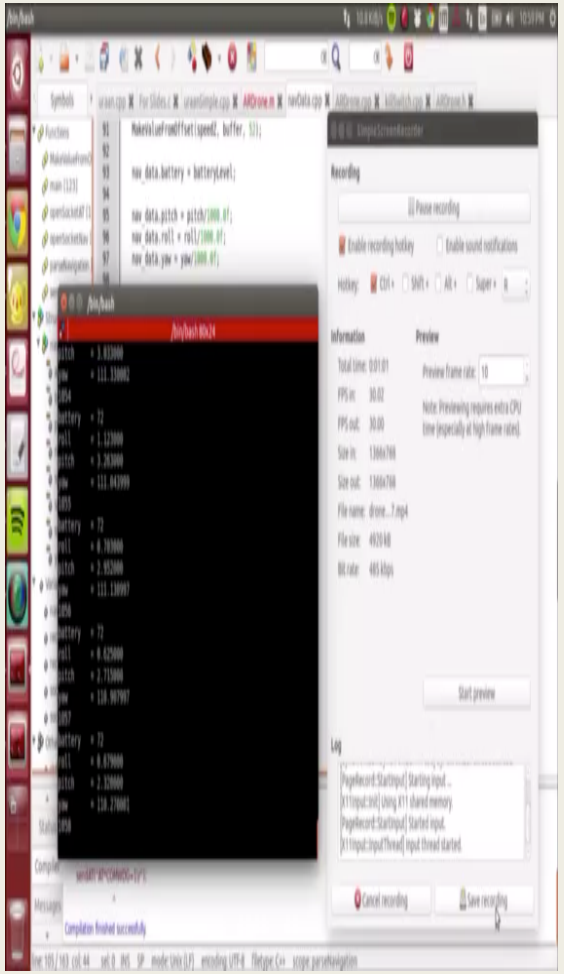
Nav_Data Demo



ROLL



PITCH



YAW

How are we using Navigation Data?

- We are not!
- The velocity commands given to the Drone are in percentages of maximum inclination.
- We planned to use Navigation Data to close the loop on velocity commands.
- So that linear and angular velocities can be specified in m/s and rad/sec.
- Not able to do so due to lack of time. (See Slide # 3)

MULTIPLE THREADS

And Boost Libraries

Multi-threaded Application

- AR-Drone requires velocity commands to be published every **30ms** for smooth movements.
- AR-Drone publishes Navigation Data every **5ms**.
- Decoding and processing an image takes over **500ms**.
- Need for multithreading for a smooth ride.

Boost Libraries

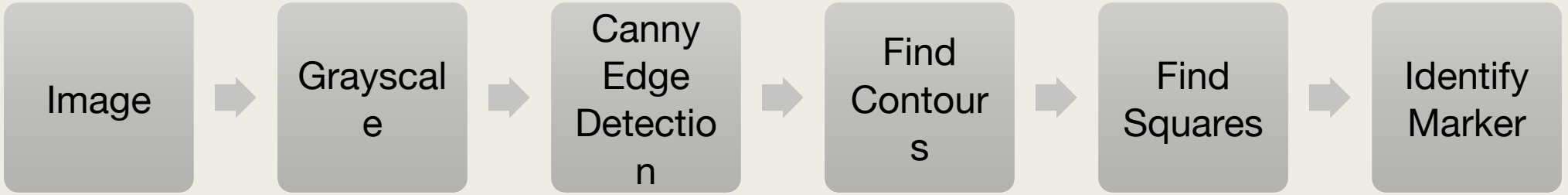
- We have used Boost libraries for threading.
- **Boost** is a set of libraries for the C++ programming language that provide support for tasks and structures such as linear algebra,pseudorandom number generation, **multithreading,** image processing, regular expressions, and unit testing.
- Becoming de facto standard.
- Abstracts hardware.
- Easy to learn and **Boosts** productivity.
- Fully cross-platform – Very easy to cross-compile.

MARKER DETECTION

Creating Marker

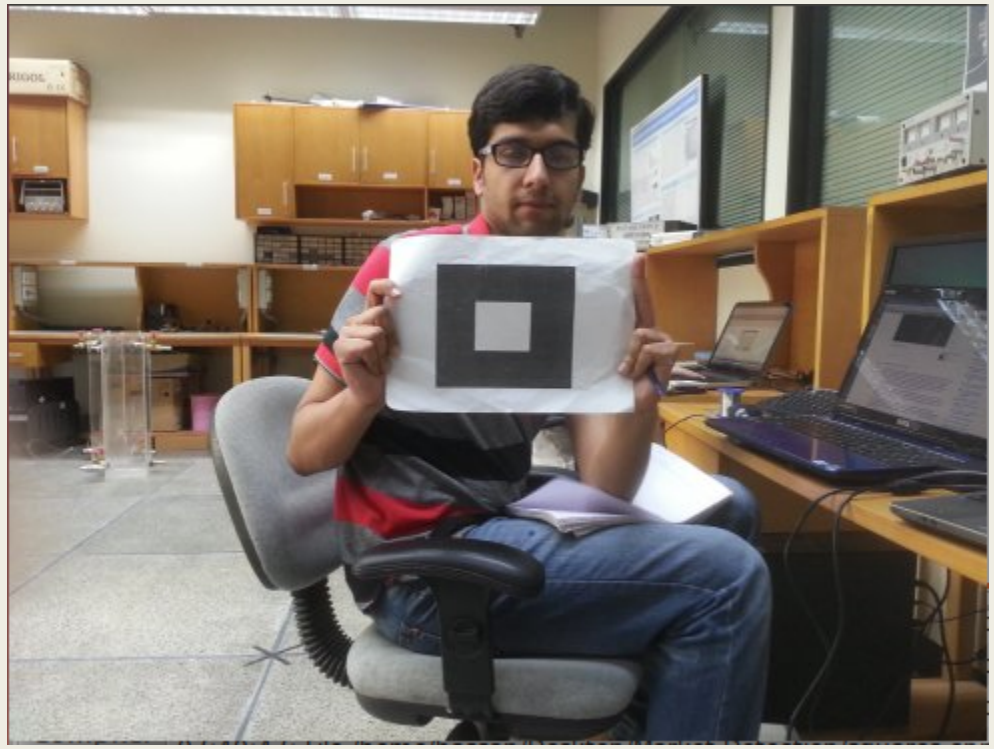
- Ready-made marker and QR-Code detection libraries too intensive for AR Drone processor.
- Need for a simple marker that is easy to detect with low false positives.
- So we created our own marker.
  - *A square within a square.*
- And wrote code to detect that marker.

Creating Marker: Big Picture



Marker Detection Process

- Original Image

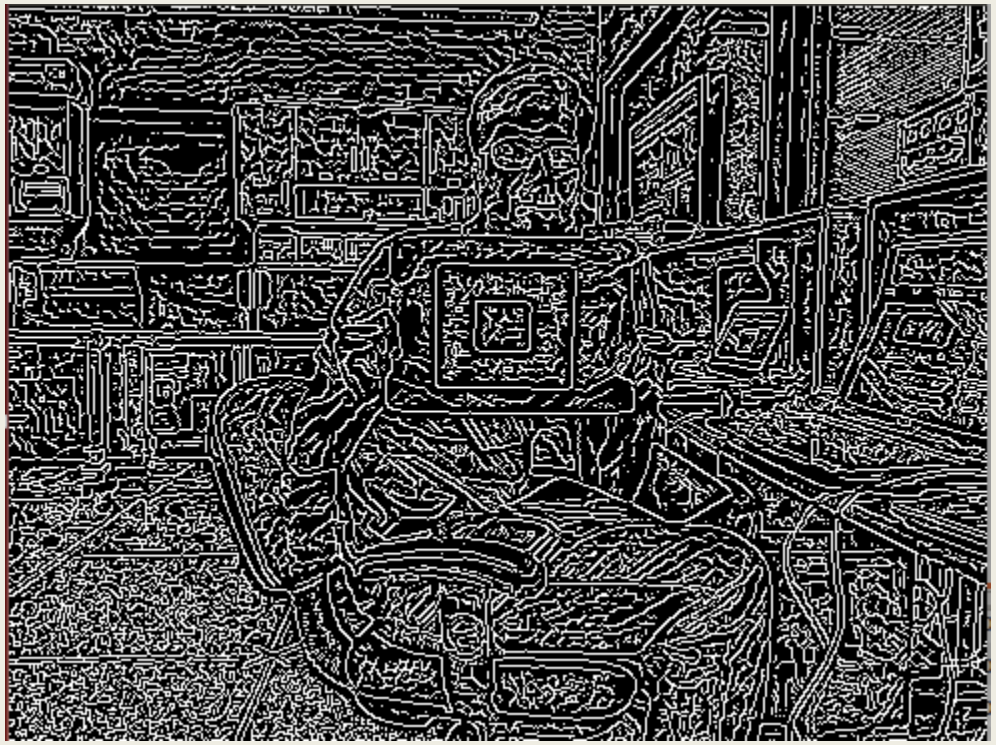


Marker Detection Process

- Grayscale

Marker Detection Process

- Canny Edge Detection



Marker Detection Process

- Find all contours in a given image

  Find all contours in the given image
- Approximate

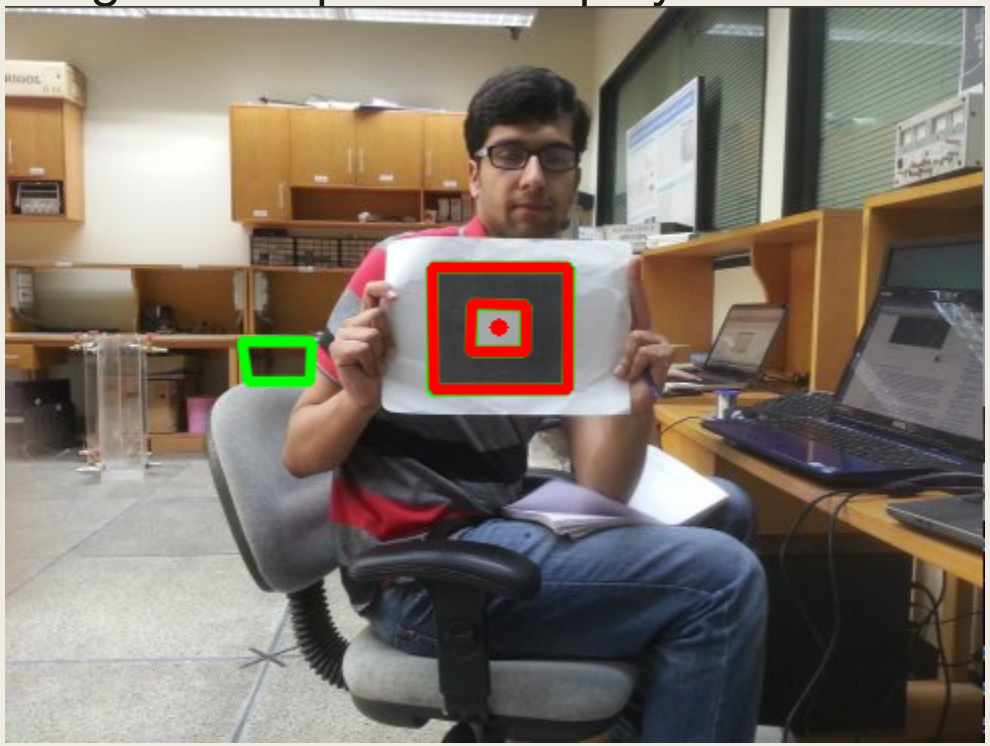
  approximate bad shapes to the closest matching shape
- Find Squares

  apply conditions to find shapes with 4 edges and angle in specific range
- Square filtering

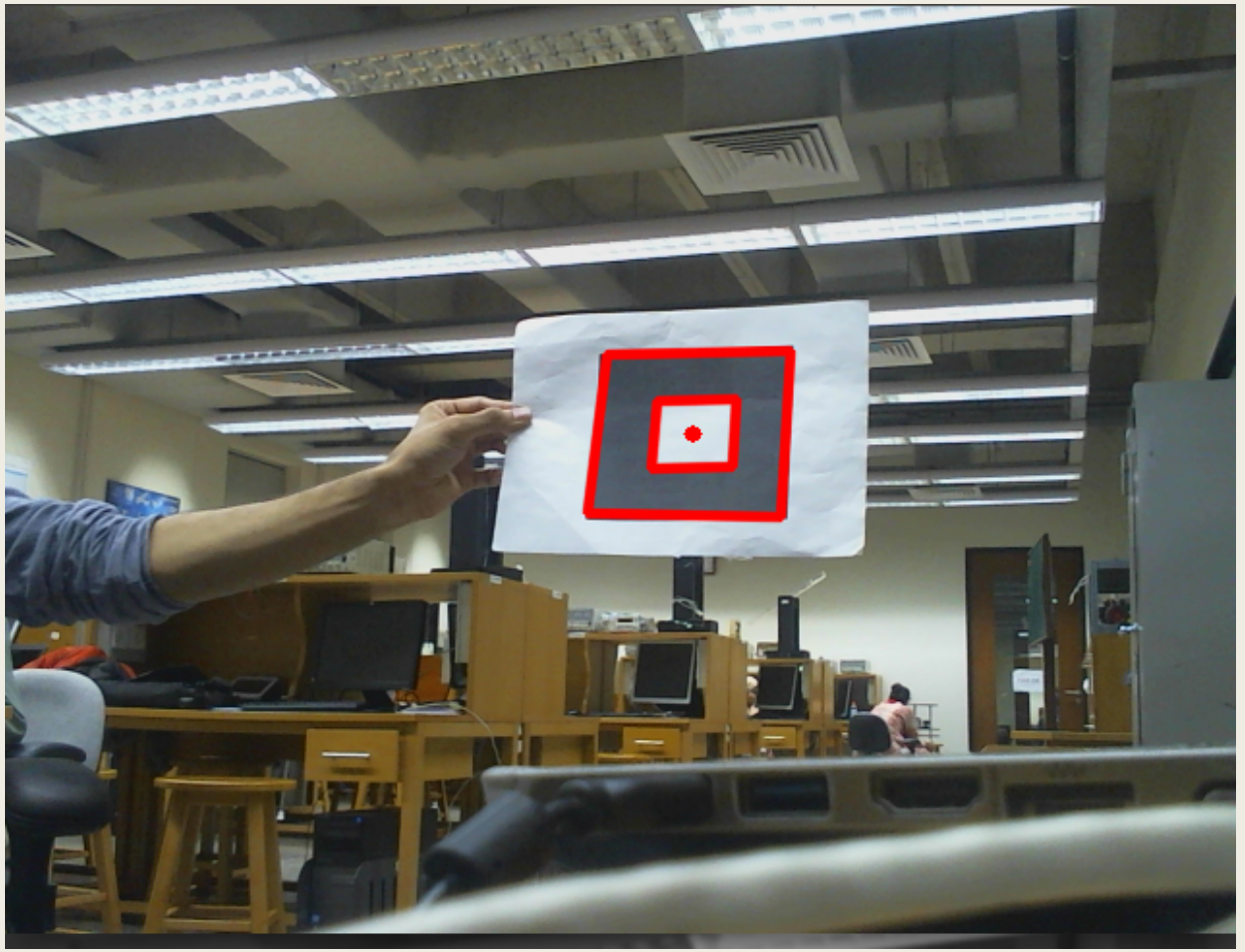
  filter squares to our specific condition

Marker Detection Process

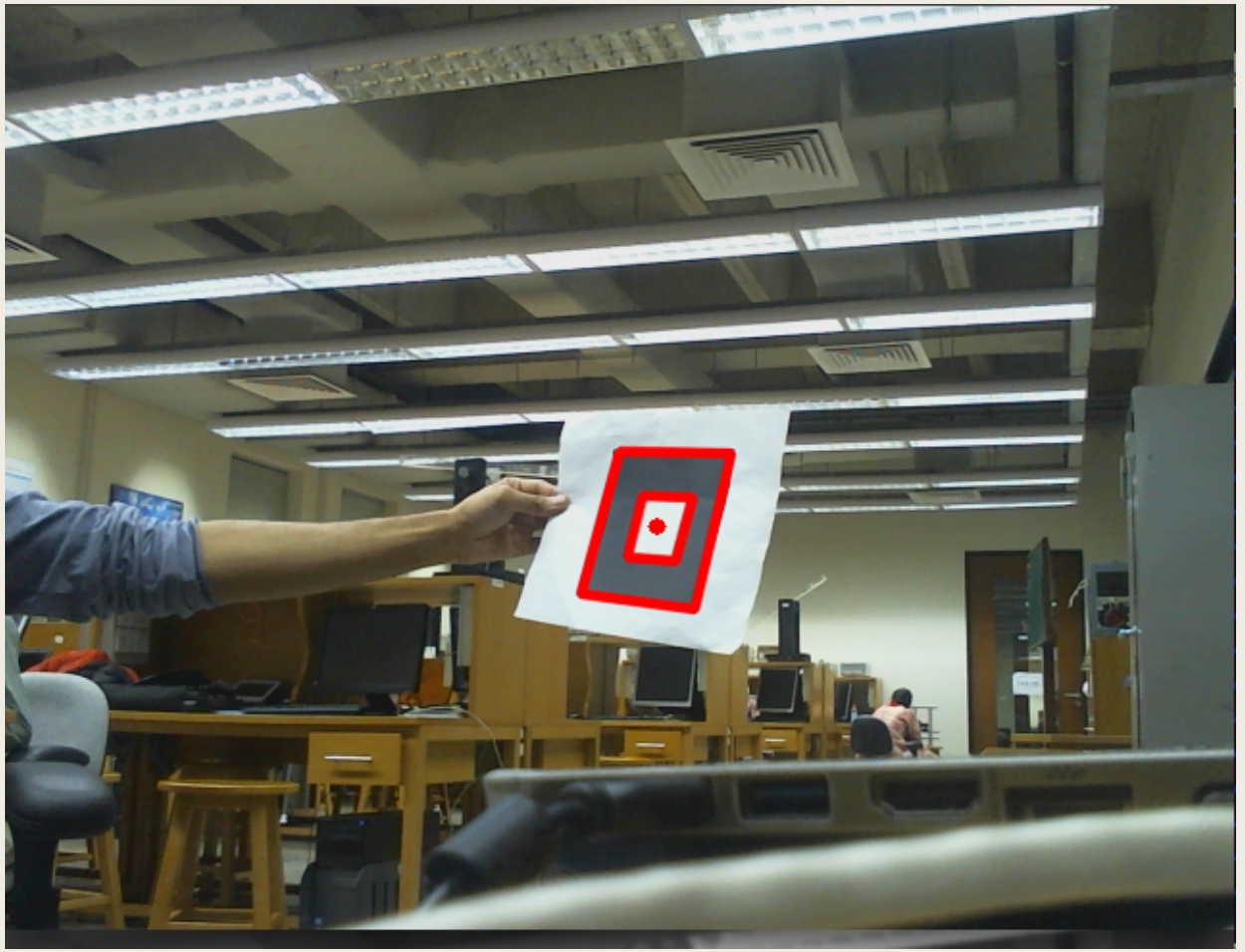
- Find center of image using moments
- Draw center of image and square to display window



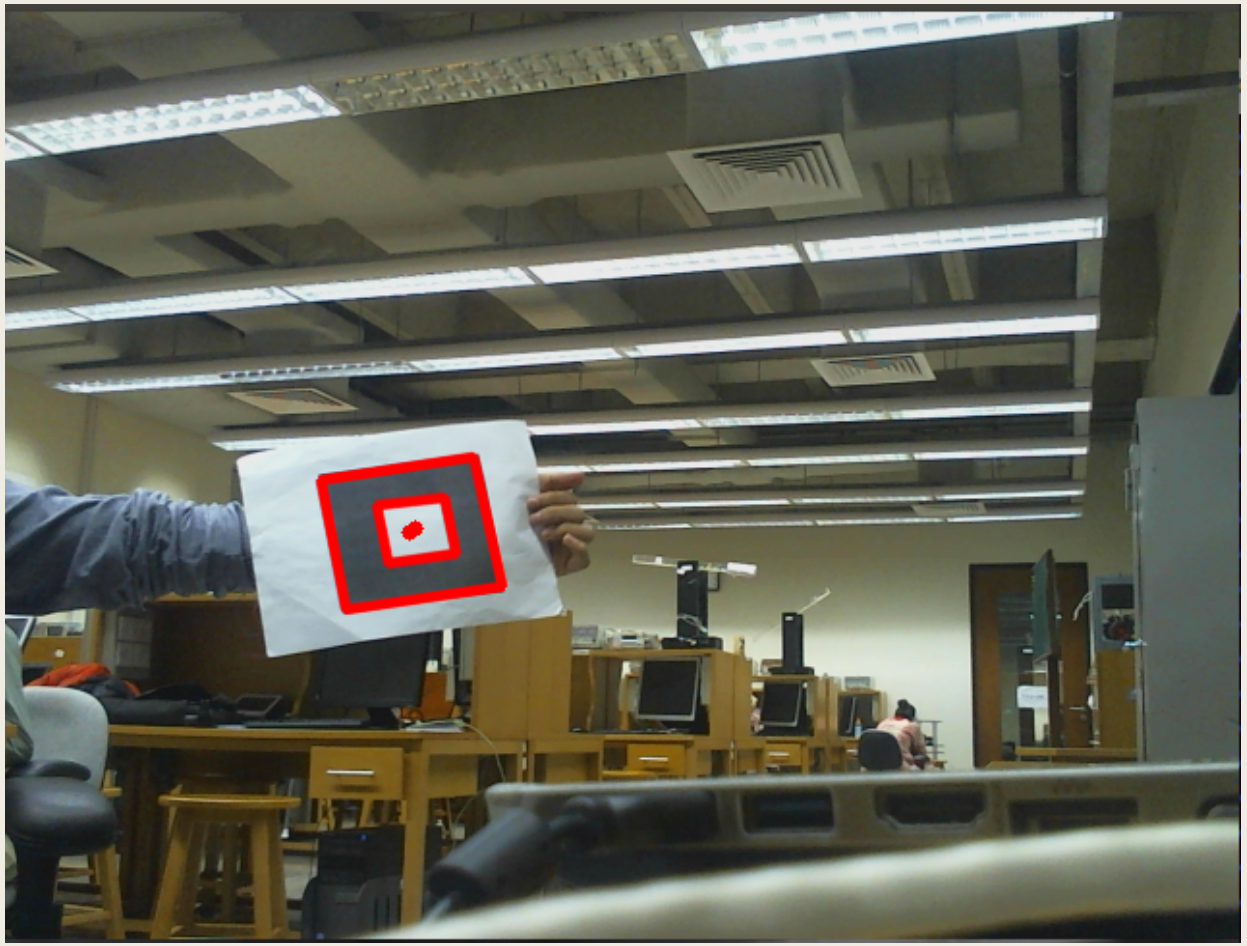
Robustness



Robustness

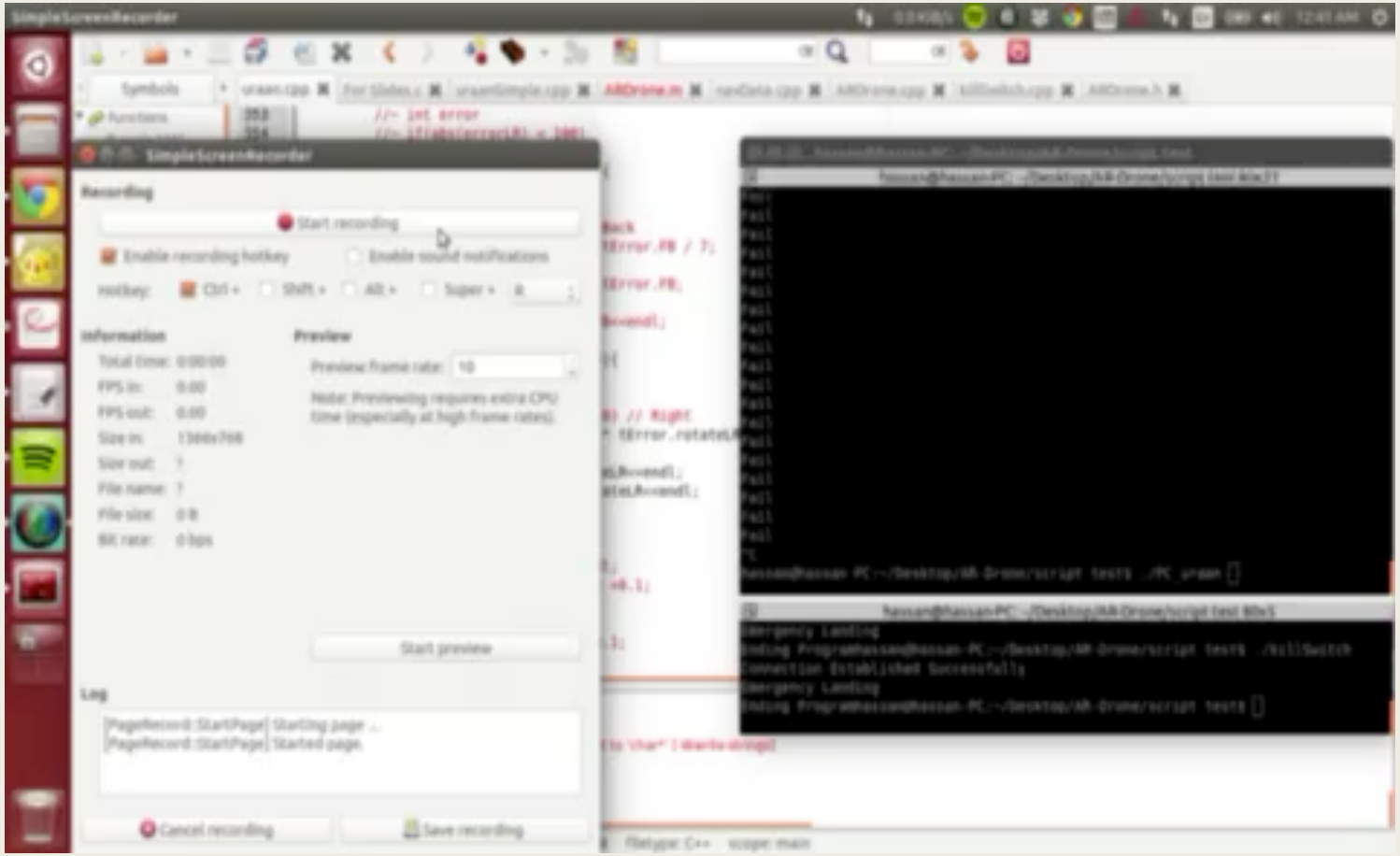


Robustness

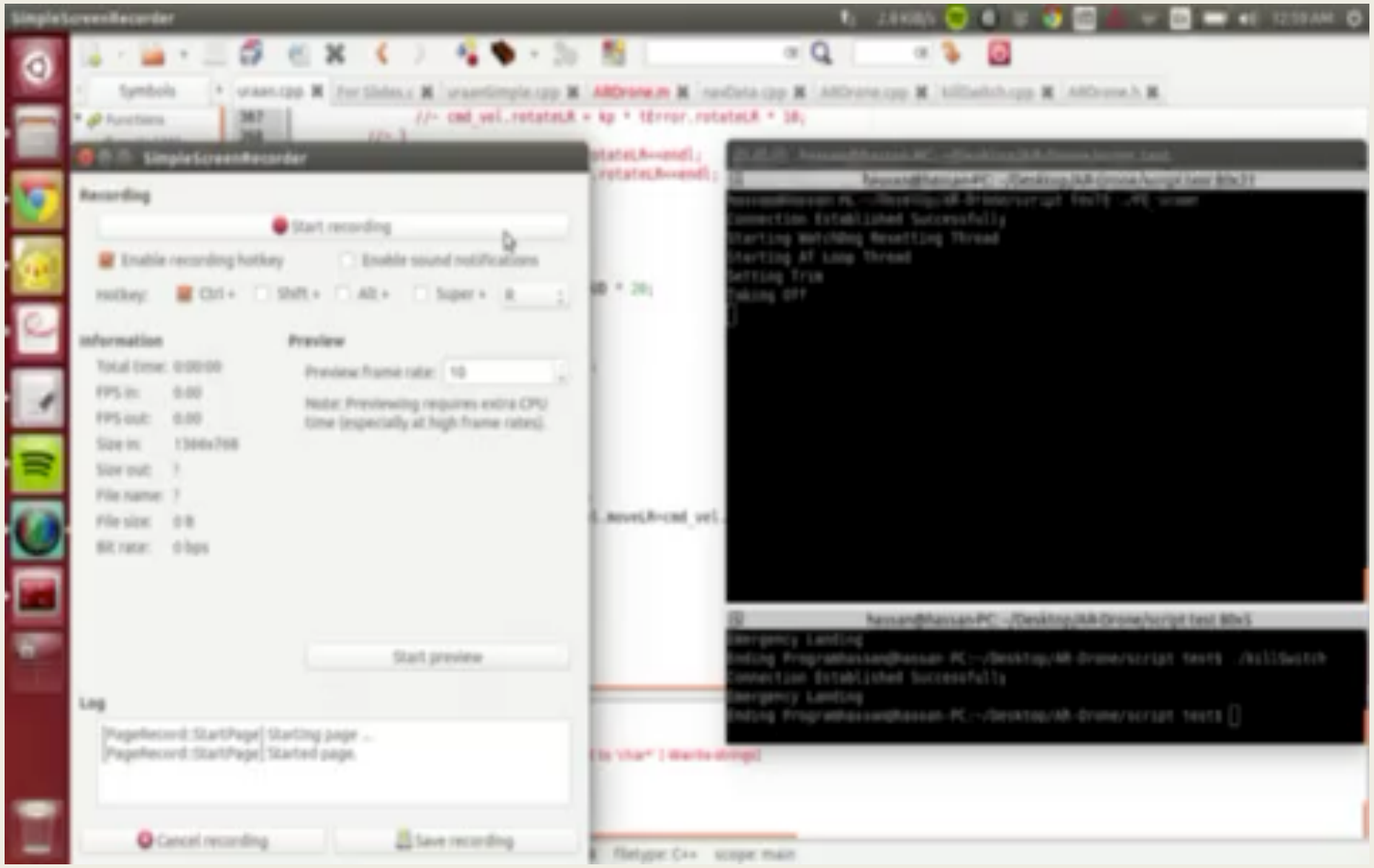


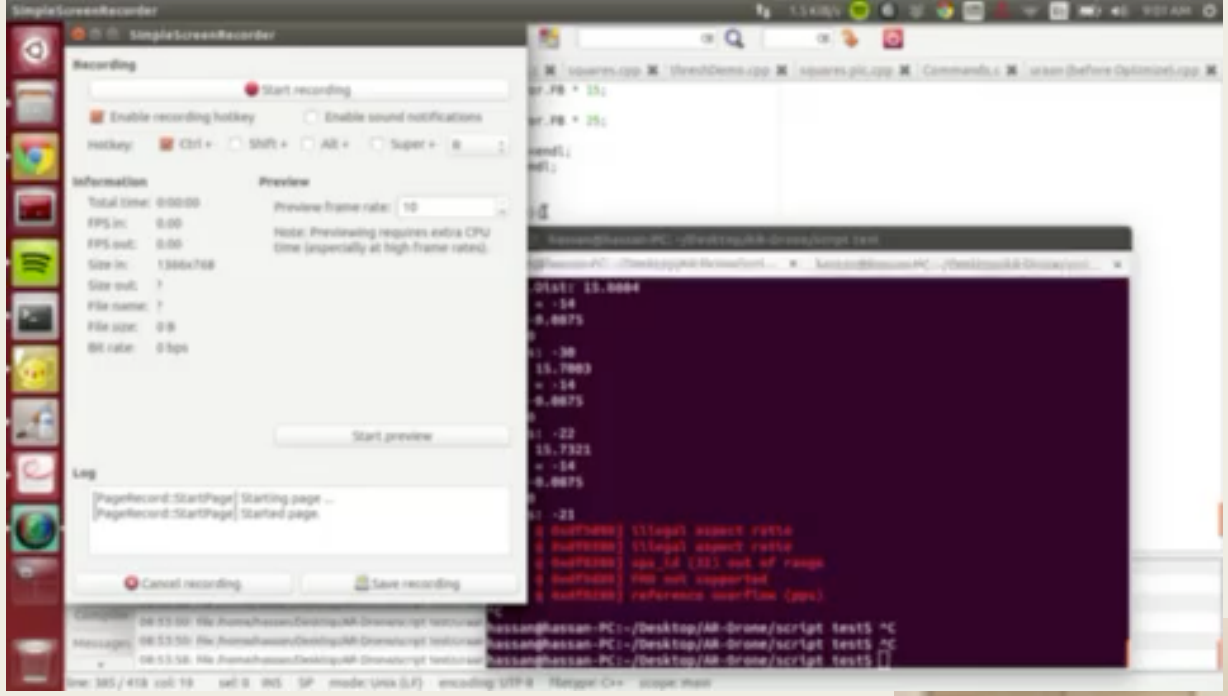
DEMO

Left Right


SimpleScreenRecorder
Recording
Start recording
Enable recording hotkey
Enable sound notifications
Information
Preview
Start preview
Log
Cancel recording
Save recording

Up Down


SimpleScreenRecorder
Recording
Start recording
Enable recording hotkey
Enable sound notifications
Information
Preview
Start preview
Log
Cancel recording
Save recording


SimpleScreenRecorder
Recording
Start recording
Enable recording hotkey
Enable sound notifications
Hotkey:
Ctrl +
Shift +
Alt +
Super +
Information
Preview
Preview frame rate:
Start preview
Log
Cancel recording
Save recording
hassan@hassan-PC:~/Desktop/AR-Drone/script test5 ^C

CONCLUSION

What have we achieved?

- Definition of Drone:
  - An unmanned aircraft or ship that can navigate autonomously, without human control or **beyond line of sight**.
- AR Drone is now truly a Drone.
- Can be used beyond line of sight.
- Truly autonomous.

PROBLEM?