Communication Networks and Computer Vision Based Control

Nicholas Tovar

Ventura College
Physics/Computer Science

Mentors

Yonggang Xu, James Riehl

Professor

João Hespanha

Communication Networks and Computer Vision Based Control

Control over a network

Use Client/Server protocols to interpret and process data over the network

- Data consists of video/captured images and periodic sonar readings.
- Networking using Ethernet(100 Mbits/s) or Wireless(11 Mbits/s) connections

Examine the limitations of real time video processing on a lower bandwidth network

- Emphasis on real-time video interaction.
- Creating/Using algorithms for overall more efficient client-side video processing

What is my Role?

Create our own re-usable code to control the robot

Network Connection

- Network programming connection using sockets

Camera manipulation and remote viewing

- PTZ (Pan-Tilt-Zoom) camera motion control
- Frame-by-frame image capture
- Client-side image processing
- Pattern recognition algorithm

Sonar sensor feedback and motion

- Sonar information retrieval and calculations
- Direct Integer or Byte Command Based movement

```
ArRobotPacketReceiver sip;
ArRobotPacket* packet;
sip.setDeviceConnection( &conn);

ArUtil::sleep(100);

if((packet = sip.receivePacket(0)) != NULL)
  {

     int x = 0;

     unsigned char byteOne;
     unsigned char byteTwo;
     unsigned char sonarNum;
     unsigned char tempInt[2];
     unsigned char numReadings;
     unsigned short* sonarReading;

     numReadings = (packet->getBuf())[22];

     while(x < numReadings)
     {

          sonarNum = ((unsigned char) (packet->getBuf())[23 + (3 * x)]);
          printf(" Sonar number: %x\n",sonarNum);

      byteOne  =  ((unsigned char) (packet->getBuf())[24 + (3 * x)]);
          printf("First Byte: %x\n",byteOne);

      byteTwo =  ((unsigned char) (packet->getBuf())[25 + (3 * x)]);
          printf("Second Byte: %x\n",byteTwo);


      tempInt[0] = byteOne;
      tempInt[1] = byteTwo;
      sonarReading = (unsigned short*)tempInt;
```

Laboratory Process/Equipment

System Schematic

Internet &
802.11b

Micro-Controller

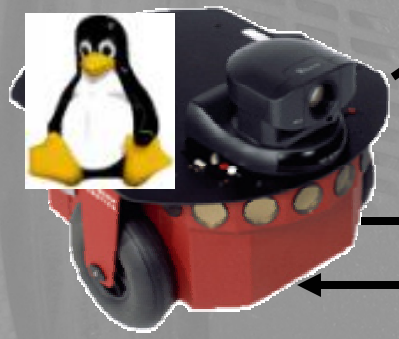
Sonar

Encoders

Motors

SIP
(Server Info Packet)

CIP
(Client Info Packet)



Commands (PTZ)

Frames

Sony PTZ Camera

Beginning Steps

Images and Sonar

```
Sonar number: 4
First Byte: d5
Second Byte: e

Sonar number: 4
First Byte: d7
Second Byte: e

Sonar number: 4
First Byte: d6
Second Byte: e
Sonar Reading: 3798
Sonar Time Stamp: 10:19:24:685
numReadings = 2

Sonar number: 4
First Byte: d8
Second Byte: e

Sonar number: 4
First Byte: d8
Second Byte: e
Sonar Reading: 3800
Sonar Time Stamp: 10:19:24:777
numReadings = 3

Sonar number: 4
First Byte: d7
Second Byte: e

Sonar number: 4
First Byte: d8
Second Byte: e

Sonar number: 4
First Byte: d9
Second Byte: e
Sonar Reading: 3801
Sonar Time Stamp: 10:19:24:872
numReadings = 2

Sonar number: 4
First Byte: d8
Second Byte: e

Sonar number: 4
First Byte: d8
Second Byte: e
Sonar Reading: 3800
Sonar Time Stamp: 10:19:24:965
numReadings = 3

Sonar number: 4
```

Real-time Examples

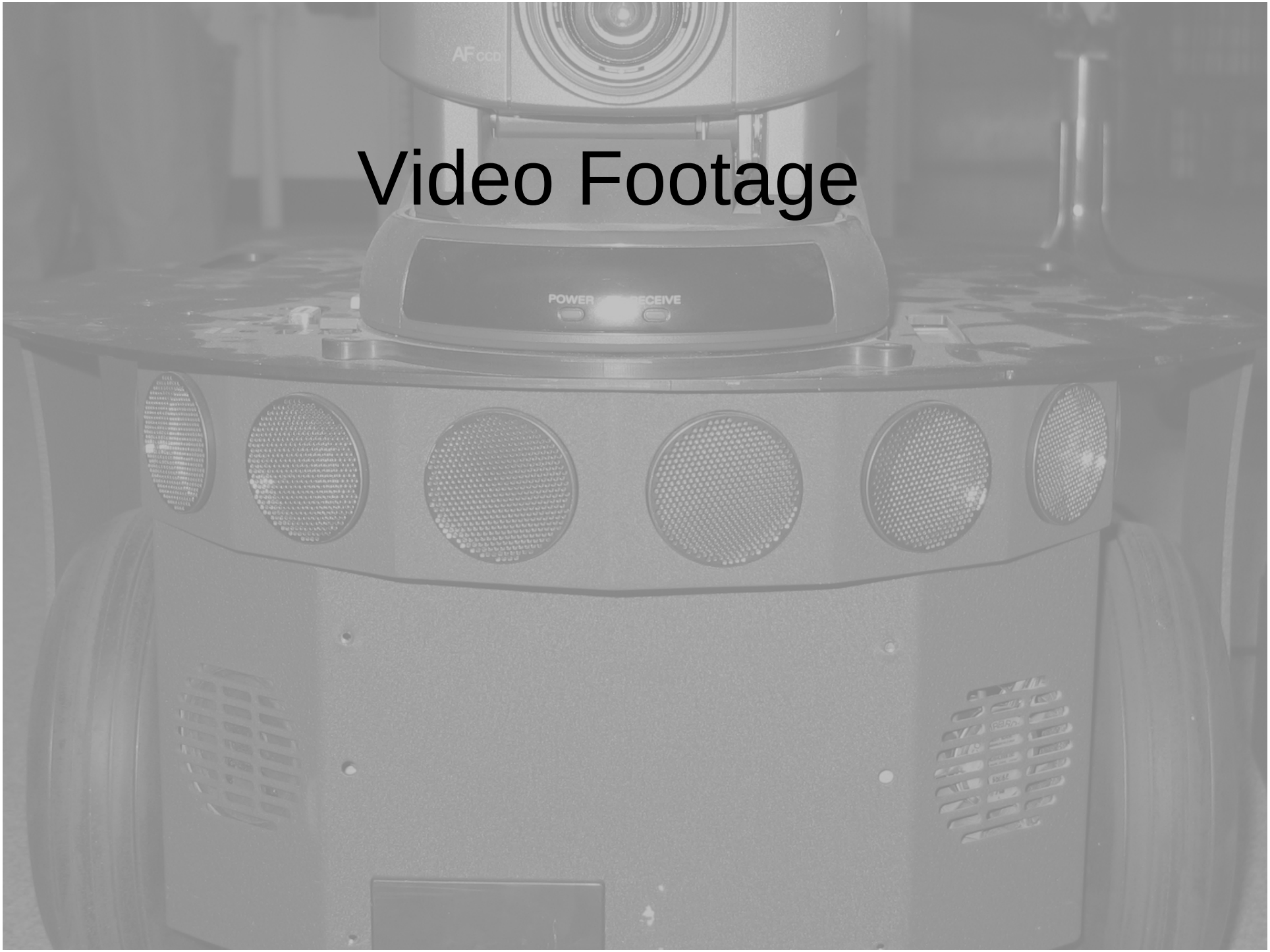
Real-time Video integrated with motion and sonar

Video and Network algorithm Incorporation

- High Resolution -> 640x480, 24 bit color (900KB)
- Compressed file -> 320X240, 8 bit grayscale (75KB)
- Resultant file is 12 times smaller

Object Avoidance using sonar

- Sonar directed movement.
- Robot avoids obstacles by turning when objects are detected within a given range.


Video Footage

What I have Accomplished

Summary of Achievements

Programming, C++

- Created my own “Basic Movement” class
- Coded an “Improved Sonar” class
- Utilized programming code that favored direct packet communication
- Employed a pre-existing video and pattern recognition algorithm

Understanding/Investigative

- Analyzed and understood robot manufacturer code and robot specs
- Examined how the robot interacts with the network

What's Next?

Future Plans

- Combine Object Avoidance with vision control
- Integrate pattern recognition and image processing
- Refine sonar control