A distributed architecture for interacting with NAO

F. Badeig Q. Pelorson S. Arias V. Drouard
I. Gebru X. Li G. Evangelidis R. Horaud

INRIA Grenoble Rhône-Alpes // Perception team
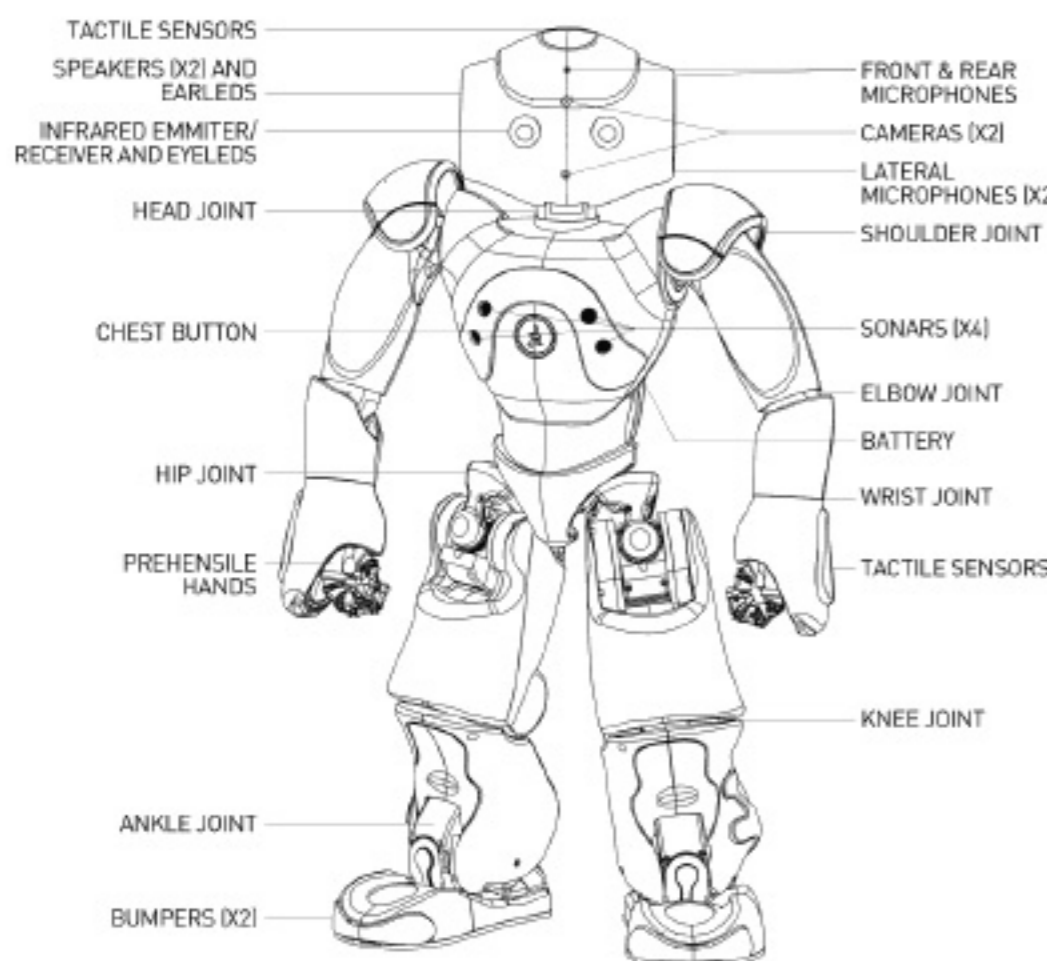{firstname.lastname}@inria.fr

1. NAO Specifications

Hardware specifications

- CPU: Intel Atom Z530 (2×1.6 GHz - 32 bits)
- RAM: 1 GB
- Servomotors: 24 degrees of freedom

Embedded software

- Face detection, Sound localization
- Motion control, Posture control
- Speech recognition (predefined words or sentences in several languages)
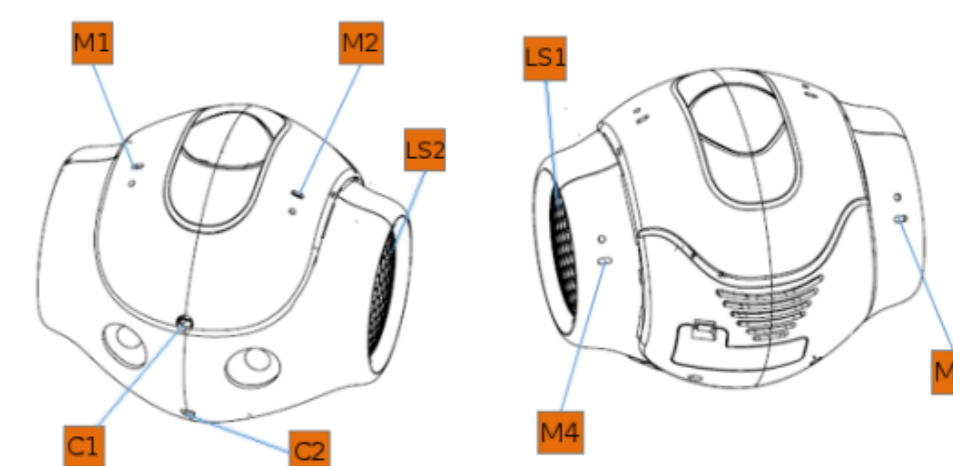

Current limitations:

1- Limited on-board computing resources

2- Require advanced embedded-programming skills
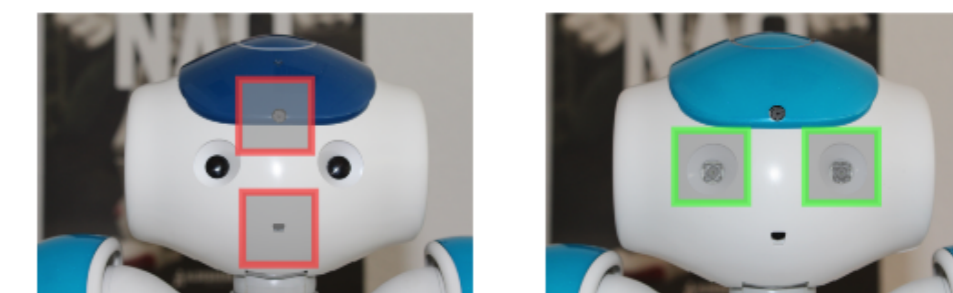
3- Only few functionalities available

Challenges: Easy and fast software development for NAO

1- Using various programming environments (C++, Matlab, python)

2- Combining external toolboxes (OpenCV, etc.) with embedded software
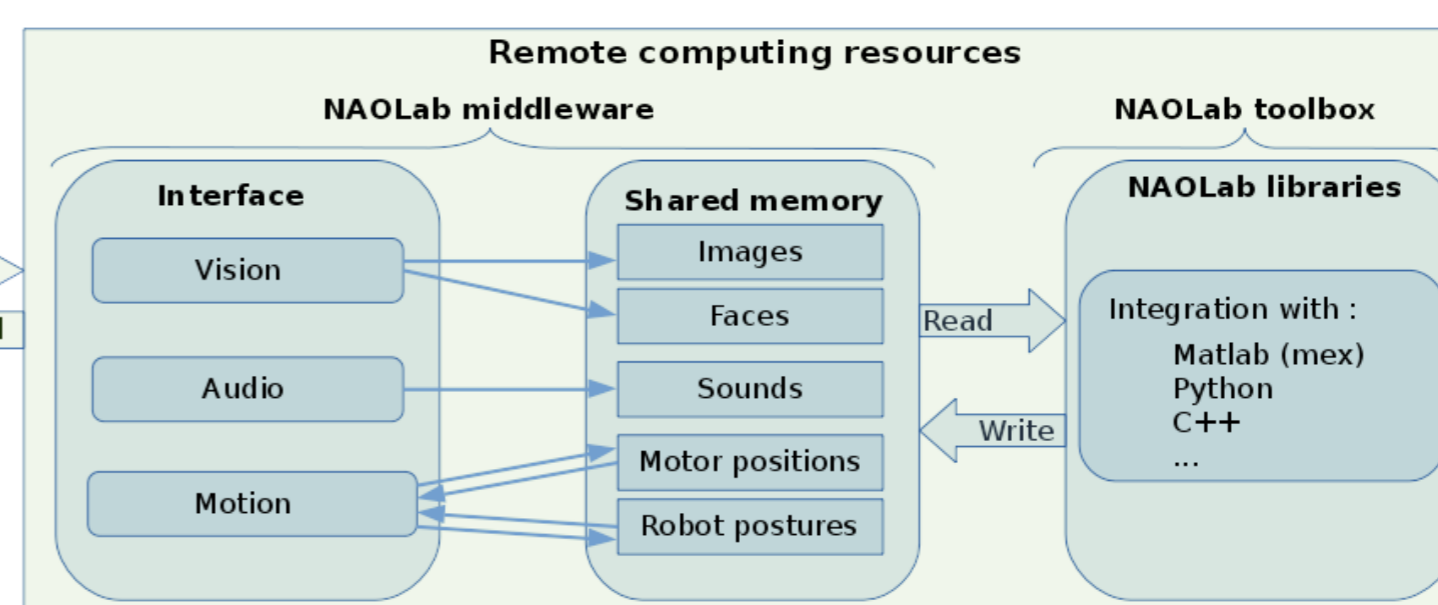
Microphone configuration

Camera configurations

Standard   New

2. Proposed solution

A middleware to **access** and **control** NAO

[Advantage] Deploy advanced algorithms without embedded-software constraints

3. Toolbox to interact with NAO

Provides the following features:

1- The middleware complexity is transparent to the users

2- A user-friendly interface is provided

(NAOLab C++ and python libraries)

NAOLab C++ library				
Vision	Access:	getImage()	getImages()	getFaces()
	Control:	setResolution()	setCamera()	
Audio	Access:	getSound()		
	Control:	textToSpeech()	playAudioFile()	
Motion	Access:	getAllMotorInfo()	getMotorInfo()	getPosture()
	Control:	moveHead()	moveMotor()	moveToPoint() setPosture()

4. Example: Sound source localization (SSL) in Matlab

Requirements: Vision (grab images from left camera + face positions), Audio (grab sound buffers from microphones), Motion (control head motors)

INPUT   ALGORITHM (Matlab)   OUTPUT

Vision data

Audio data

front left

front right


```
while(1)
    imageData = getImage( );
    faceData = getFaces( );
    soundData = getSound( );
    [isSound,theta] = SSL(soundData);
    if(isSound)
        [value,idx] = min(abs(faceData(1,:)-theta));
        newYaw = faceData(1,idx);
        newPitch = faceData(2,idx);
        moveHead(newYaw,newPitch);
    end
end
```

Visual-Audio Alignment

Localized sound source

NAO Motion

NAO turns its head towards the sound direction

Download our code