

LEGO car course topics

Xiebing Wang, Xiang Gao, Biao Hu, Kai Huang

Chair of Robotics and Embedded Systems
Department of Informatiks
Technische Universität München

Topic 1: autonomous driving of lego car

Contents

- Assemble the lego car.
- Run our designed hardware and software.
- Apply some more advanced autonomous techniques - vision-based or control strategy.

Topic 1: autonomous driving of lego car

Contents

- Assemble the lego car.
- Run our designed hardware and software.
- Apply some more advanced autonomous techniques - vision-based or control strategy.

Difficulties

- Debugging hardware/software design.
- The movement of lego car is not predictable.
- Advanced autonomous techniques.

Topic 1: autonomous driving of lego car

Contents

- Assemble the lego car.
- Run our designed hardware and software.
- Apply some more advanced autonomous techniques - vision-based or control strategy.

Difficulties

- Debugging hardware/software design.
- The movement of lego car is not predictable.
- Advanced autonomous techniques.

Expected team size

3 members

Topic 2: autonomous driving of iron car

Contents

- Being familiar with the existing system in this car.
- Solve the existing problem.
- Apply some more advanced sensors or algorithms to make the car more autonomous.

Topic 2: autonomous driving of iron car

Contents

- Being familiar with the existing system in this car.
- Solve the existing problem.
- Apply some more advanced sensors or algorithms to make the car more autonomous.

Difficulties

- Debugging hardware/software design.
- Safety is very important since the car is fast and heavy.
- The computing power is limited but the computing demand is high.
- The driving situations can be very complex to analyze at runtime.

Topic 2: autonomous driving of iron car

Contents

- Being familiar with the existing system in this car.
- Solve the existing problem.
- Apply some more advanced sensors or algorithms to make the car more autonomous.

Difficulties

- Debugging hardware/software design.
- Safety is very important since the car is fast and heavy.
- The computing power is limited but the computing demand is high.
- The driving situations can be very complex to analyze at runtime.

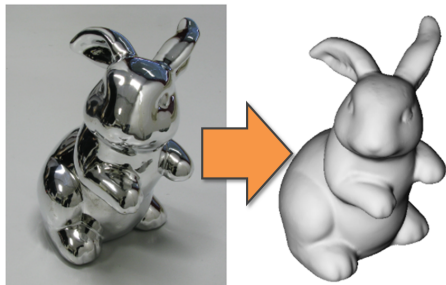
Expected team size

5 members

Topic 3: REMODE: Probabilistic, Monocular Dense Reconstruction in Real Time by OpenCL

Background

- Constructing 3D scene using only one camera (a video).
- Existing work with open source code - cuda parallel code.
- Real-time.



Topic 3: REMODE: Probabilistic, Monocular Dense Reconstruction in Real Time by OpenCL

Task

- Understand the existing work.
- Port this CUDA-based open-source to be OpenCL-based code.
- Evaluate the performance of the two different programs.

Topic 3: REMODE: Probabilistic, Monocular Dense Reconstruction in Real Time by OpenCL

Task

- Understand the existing work.
- Port this CUDA-based open-source to be OpenCL-based code.
- Evaluate the performance of the two different programs.

Why this topic

- Very useful, applicable in many situations.
 - 1 Using cellphone to construct the 3D object.
 - 2 3D printer.
 - 3 Low cost.
- The autonomous car needs such technique to know the distance between itself and the front object.

Topic 3: REMODE: Probabilistic, Monocular Dense Reconstruction in Real Time by OpenCL

Why OpenCL

- Can run in heterogeneous system (CPU, DSP, FPGA, and GPU).
- Can run in cellphone.
- The results of using OpenCL and CUDA are useful for industry.

Topic 3: REMODE: Probabilistic, Monocular Dense Reconstruction in Real Time by OpenCL

Why OpenCL

- Can run in heterogeneous system (CPU, DSP, FPGA, and GPU).
- Can run in cellphone.
- The results of using OpenCL and CUDA are useful for industry.

Difficulties

- Understand this technique by reading the paper.
- Be familiar with CUDA and OpenCL programming.
- Fully parallelize the execution of program.
- Evaluating the performance.

Topic 3: REMODE: Probabilistic, Monocular Dense Reconstruction in Real Time by OpenCL

Why OpenCL

- Can run in heterogeneous system (CPU, DSP, FPGA, and GPU).
- Can run in cellphone.
- The results of using OpenCL and CUDA are useful for industry.

Difficulties

- Understand this technique by reading the paper.
- Be familiar with CUDA and OpenCL programming.
- Fully parallelize the execution of program.
- Evaluating the performance.

Expected team size

- 3-4 members