|  |  |  |  |
| --- | --- | --- | --- |
| **Group and title** | **Equipment or People** | **What to do and how evaluated** | **Partial tasks and Student Experience so far** |
| **Group 1. Radio-Controlled Tetrix Mobile Robot with hands and head.** | 1. Tetrix Kit, 2. Lion head robot of Martin Lukac 3. Laptop with RobotC and ESRA software. ESRA software to be used for motion generation synchronized with speech. 4. Tetrix Radio-Controller that works with RobotC 5. Humanoid Hand to be designed from scratch, use Tetrix and any other stuff. Write me what else you need. | **MAIN TASK.** A robot should be able to go places and transport items from place to place showing his simple emotions using hand, head and motion of the whole body. This robot will be used in research on new type of neural network developed at Ohio University (not this quarter). The goal of the class is to design a robot that will later on integrate much of our existing AI/Vision/Speech/Dialog software and be used in research. Work done the first quarter in ECE 478 is only a beginning.  **The final demo scenario:**  We see a mobile robot puppet with a head/neck that has more than 7 degrees of freedom and can show some preprogrammed behaviors, these behaviors show at least two emotions: “happy and energetic” and “unhappy sad depressed and with no energy.” The robot has one hand that can grab an item from a desk or chair. The robot is radio-controlled. It is a kind of a large toy. It is a mobile robot with one hand and neck/head.  Controlled by radio, the robot approaches the table and grabs a can. The robot says:  “ I am so happy to be a good servant. I am grabbing a can for you” or something like that. The robot does face/neck gestures or hand gestures showing that he is happy. This can be a complex and long behavior and its theatrical value counts. Next the robot drives to another desk and puts the can on the desk. The positioning of the can should be accurate. The robot says: “It was so nice to serve you” or something similar and shows happy gestures.  The robot tries to take a glass and his attempt fails. The robot says” I am so unhappy to break the glass. My wife will beat me up for this.” The robot shows unhappy gestures.  The robot should be able to grab few types of special items reliably.  You are free to extend this scenario, but the scenario should have base motions, hand motions and head/neck motions synchronized in some meaningful way.  The robot is completely controlled but some behaviors can be autonomous or partially automated (grabbing, leaving, shaking, whatever). The robot should use recorded human voice integrated in its behaviors. The voice should correspond to the head and hand. Remember that our metaphor here is theatre. This is not yet an AI project. It is only “robot design and test” project.  *Next quarter we will add dialog and non-recorder speech and perhaps vision. We will add also a more advanced architecture based on learning and reasoning.* | **Need to learn**. Motion and Behavior Programming in RobotC. Motion and speech generation with ESRA editor. Mechanical assembly with Lego NXT and Tetrix kits. Mechanical design of a humanoid hand that can grab a simple object. System integration using emotional software from previous groups.  **TASK 1:HEAD** Find Lion head/neck in the main lab. Fix it. Use existing motions of this head from Martin. Design your own head/neck motions using ESRA editor or other editor or by hand. Synchronize with speech and edit few funny behaviors. The head should represent whatever you want, human or animal or fairy tale character, but not a “robot-like” robot. The best would be some kind of a puppet that can show emotions. Concentrate on as good as possible representation of two emotions with the head: happy and unhappy. **Contact person:** Martin Lukac.  **lukacm@kameyama.ecei.tohoku.ac.jp**  **TASK 2: BASE** Find Tetrix kit and 3 robot bases in main lab. Adapt them or design your own new base. The base should allow to have the body with hands and the neck so it must be large and strong. Design the body, the base and the links to both arms. Program radio-controlled behaviors of the robot base. These should be all behaviors other than hand or head behaviors.  **TASK 3: HAND** You have to build one robot hand/arm/palm combination that will satisfy two goals: (1) will allow to take simple items such as plates, boxes, cans, etc. Not all possible items like a human hand. (2) will be as much as possible similar to anthropomorphic hand that will have a good match with the lion head. You can modify the lion robot as you want by putting latex mask or whatever. It will be not a lion anymore. Although you build only one arm, the robot should have a possibility and design to have two identical left/right arms useful for next projects.  **TASK 4: BASE.** Design software for robot path following.  **TASK 5:** **Hand.** Design software for grabbing item 1. Modify it to grabbing behaviors and putting behaviors of other items, item 2, item 3 etc.  **Task 6.** Integrate all these behaviors in a radio-controlled robot. Make the maximal use of all programmed behaviors and keys on the robot radio controller. Show the complete demo to the class and invited professors, students, families and actors. |
| **1** | Roger Bingham **roger.bingham@gmail.com** | Face/Head design and programming. Programming facial/neck gestures. Later on, programming associated hand gestures. | Undergraduate. C, C++, assmb, Java, familiar with industrial robots at work. |
| **2** | Caroline Stepnowski  kestpur@comcast.net | Face/Head Design and programming using ESRA editor. | Undergraduate. C++, ARM, Java, new to robots. |
| **3** | Christopher Cowan  **ChristopherLCowan@yahoo.com** | Face/Head Design concept and programming concept. Learn ESRA editor. How to integrate it.  We need a leader of the head group who will be also the leader of the whole project. | Undergraduate. Assmb. C++, Pascal, Basic. New to robots. |
| **4** | Matt Blackmore **blackmom@pdx.edu** | Hands design and programming in RobotC. | Undergraduate. What skills? |
| **5** | David Gaskin  **d.b.gaskin@gmail.com** | Mobile base design and programming in RobotC. | **Graduate,** C, C++, soccer playing robots. |
| **6** | Shaun Ochsner **scochsner@gmail.com** | Hand programming in RobotC. | Undergraduate. |
| **7** | Nathanael Makowski  [**ccnate\_m@hotmail.com**](mailto:ccnate_m@hotmail.com)  **nathanael.makowski@gmail.com** | Mobile base programming in RobotC.  We need a leader of the mobile base group who will be also the software integration person. Must know both RobotC and ESRA editor. | **Graduate.** C programming. Mechanical and/or spatial nature. Lego NXT, PID control. |
| **8** | Jacob Furniss  **Leader of hand group.**  **furnissj@cecs.pdx.edu** | The concept of the hand for this robot. How many DOF? How many servos? How many other motors? What kinds of motors? Drive design? Kinematics and dynamics. Interface with hand programmers. Take advise from them and give them advise. Integration of a working hand with grabbing and putting behaviors. This person does not need to write RobotC software but should understand it. | **Graduate**. Mechanical engineering. Controls, microcontrollers, little C, interested in motion control, autonomous functions, no ECE background. Good hands on skills. |
| **Group 2.**  **Autonomous Tetrix Mobile Robot with robotic arm and ESRA face.** | 1. Tetrix Kit, 2. Schreodinger cat robot of Arushi Raghuvanshi. 3. Laptop with RobotC and ESRA software 4. Radio-Controller of RobotC. The behaviors are autonomous. 5. Robotic Hand to be designed from scratch, use Tetrix and any other stuff. | Similar to Group 1 but the body and head of the robot are completed and need only to be integrated. The emphasis of this robot is to design or adapt the robot arm which is not human like but should be able to grab all kinds of items reliably. This is an autonomous robot, not radio-controlled. Otherwise, this group has much freedom to select and design robot architecture, sensors, mobile base, arm or arms and gestures/behaviors of a robot puppet. The emphasis is on autonomous behaviors and reliable arm design. A robot should be able to go places and transport items from place to place showing his simple emotions using hand, head and motion of the whole body. This robot will be used in research on new type of neural network developed at Ohio University. The goal of the class is to design a robot that will later on integrate much of our existing software and be used in research. Work done the first quarter is only a beginning.  **The final demo scenario:**  We see a robot puppet on a chair like a driver of a complex vehicle. He shows his actions and emotions with exaggerated behaviors. The body has any number of  degrees of freedom and can show some preprogrammed behaviors, these behaviors show at least two emotions: “happy and energetic” and “unhappy sad depressed and with no energy.” You can use the Cat robot or you can add degrees of freedom if you wish. This puppet has head with emotional features, hands and legs and also you can control the chair to add dramatism.  The robot has one “robotic arm” that can grab an item from a desk or chair. The robot is NOT radio-controlled. It is completely autonomous.  Autonomously, the robot approaches the table and grabs a can. (the position is known) The robot says:  “ I am so happy to be a good servant. I am grabbing a can for you” or something like that. The robot does all body face hand gestures showing that he is happy. This can be a complex behavior. Next the robot drives to another desk and puts the can on the desk. The robot says: “It was so nice to serve you” or something similar and shows happy gestures.  The robot tries to take a glass and his attempt fails. The robot says” I am so unhappy to break the glass. My wife will beat me up for this.” The robot shows unhappy gestures.  The robot should be able to grab many types of items reliably.  You are free to extend this scenario, but the scenario should have base motions, hand motions and head/neck motions synchronized in some meaningful way.  The robot is completely autonomous (grabbing, leaving, shaking, whatever). This is not yet a full AI project. It is only “simple autonomous robot design” project.  *Next quarter we will add dialog and non-recorder speech and perhaps vision. We will add also a more advanced architecture based on learning and reasoning.* | RobotC programming.  Sensors.  **TASK 1: MOBILE BASE.** Design the mobile base and body for the robot assuming the given here goal of this project.  **TASK 2: SCHROEDINGER CAT PUPPET as a driver.** You will be given the Schroedinger Cat robot with complete body. This robot will sit on some chair as a driver of the robot. You do not have to design this robot 0- only integrate it. It is controlled by ESRA and runs from a PC. There must be place on your robot base for this PC laptop. Your robot is autonomous, not radio-controlled. You can use all behaviors of the cat and design your own new behaviors, use voice and editor. Use sound/voice/speech editor that you can find on the internet.  **TASK 3: NON-HUMANOID ARM DESIGN** Find in the lab orDesign the non-humanoid arm for the robot. You can take the arm from the main desk in main lab, or adapt the Rhino arm (big one) or any other arm that you find. You can also design the arm from scratch or adapt any arm from the lab or other. My advise would be to take the little black arm that was already adapted with sensors and put it on front of the mobile base. Behind this arm should be a box with the Cat rocking on his chair. The Cat should be like a driver of the car and we should appreciate its gestures showing its actions and emotions. This is the most complex part of the project. You may decide to have two variants, fast variant and sophisticated robot arm. I allow to have actually two arms on the base. One in front and one in back. |
| **1** | Forrest Koran  **fkoran@pdx.edu** | mobile robot design and control. Programming of motions in RobotC. | Undergraduate. Interest in Tetrix. |
| **2** | Eric Casler  **Lead programmer** [eric.casler@gmail.com](mailto:eric.casler@gmail.com) | Complete arm design and programming of base functions typical for standard robotic arms. | Undergraduate. Computer Science, Java, C, C++, LISP, Python, Distributed robots, learning |
| **3** | Maher Hawash  **Project leader.**  **gmhawash@gmail.com** | Mobile base. Obstacle avoidance and path planning. Integration. Software supervision and test. | **Graduate.** High programming experience. |
| **4** | Takuya Nomura  [Takuya\_n\_acm@hotmail.com](mailto:Takuya_n_acm@hotmail.com) | ESRA robot programming and interfacing. | **Undergraduate, C, C++, no robot experience yet. 371, 372.** |
| **Group 3.**  **Constraint Satisfaction Problems for robotics on CUDA GPU system.** | No robot this quarter. Only software and theory. CUDA board. | The task of this project is the theoretical study of CSP algorithms for robotics. The lecture of the student should demonstrate his familiarity with advanced CSP and robotics concepts. | **TASKS:** Learn Constraint Satisfaction problems such as SAT, Petrick Function and Graph Coloring. Find what are the applications of CSP in robotics and robot vision. How to program them on CUDA. |
| **1** | Eric Paul | This is a one-person advanced project in theory of parallel programming for CSP problems that may be useful in robotics. | Graduate. |
| **Group 4. Parallel Processing for Stereo Vision on CUDA GPU system.** | 1. DIM Robot of Dmitriy Labunsky. 2. CUDA board. 3. PC computer in Erik’s room near the main lab. 4. Erik’s software for stereovision 5. Dmitriy‘s software for DIM robot. | The task of this project is the practical study of stereovision algorithms for robotic applications.  The demo should show that the DIM robot can correctly recognize distances to some object located in front of it. The final lecture of the students should demonstrate their understanding of stereovision in robotics and good methodology for testing the DIM robot vision system. | **TASK 1.** Learn the robot DIM of D. Labunsky. Use its two cameras for stereovision. Use algorithm by Eric Paul implemented on CUDA for stereovision. Integrate system and demonstrate its hand-shaking behavior. Because the robot has no hands yet, the screen should show the distance to items located near to robot using various colors. |
| **1** | Danny Volis  [dannyjvoils@gmail.com](mailto:dannyjvoils@gmail.com) | Design a stereo-vision system for DIM robot that will use CUDA board. He needs access to Erik’s room now. | Undergraduate. Currently working on a robot vision project with the SRV1:  <http://www.surveyor.com/SRV_info.html>  I use this robot to send a video stream to my linux box. There I process the images and send drive commands back to the robot. The heart of the system is a Bayesian network, but I would like to learn how to use OpenCV to preprocess images and do things like edge/blob detection.  I am most interested in using parallel computing to do robot control. I messed around with MPI for a while, but that seems better suited for things like simulation using batch jobs. I also experimented with postix threads, but couldn't find any good examples of how to create thread objects in C++. So for now, I'm working with a serial C++ library that I designed for the robot. |
| **2** | Vamsi Parasa |  | Graduate |
| **Group 5. Interfacing iSOBOT.** | 1. iSOBOT from Marek Perkowski with controller 2. PC computer or laptop 3. Possibly, other controllers as found on internet about iSOBOT interfacing | The task of this project is the design of the interface that will allow to control iSOBOT from a laptop or PC. Demo should show that it can be done. | **TASK 1:** Design an interface between a PC and the iSOBOT by modifying the transmitter or otherwise. In any case, the goal is to control the robot by infrared using some way of programming its actions. Communicate with Group 6 on language and editor. |
| **1** | Clint Lieser,  clint.lieser@gmail.com | The concept of the controller “hack” to be able to interface the PC to the robot to send commands by radio. | Undergraduate. ECE555/ECE556 Neural Networks, ECE559 Genetic Algorithms, ECE551/ECE552           Control Systems, CS541  Artificial Intelligence, CS421/422 Compiler Design, Embedded Firmware Design in C and Assembly (2 years), FPGA Design using Verilog (3 years), Circuit Board Design and Debug (many years). |
| **2** | Erik Witkamp [erikwitkamp@gmail.com](mailto:erikwitkamp@gmail.com) | **Help with the design and test of the interface.** | Undergraduate. CMP. 371,372, hobby closed –loop. |
| **3** | Pheng Vang  **vangpheng@gmail.com** | Help with the design and test of the interface. Software and test of complete new motions transmitted to the robot. | Undergraduate. Embedded systems. 371,372. Hobby robots. Robot arm or tetrix. |
| **Group 6. Genetic Algorithm for motion generation for iSOBOT.** | 1. iSOBOT from Aditya with controller 2. PC computer or laptop 3. GA software from internet or any resources written in C, Java, or any other language, but the same as the editor | The task of this project is to investigate the following research issues for the first time.   1. Can GA be used to invent new motions for iSOBOT? 2. What are the best GA parameters for this task? 3. Can we add new operators to GA that will allow more interesting or better in some way motions for GA? 4. How the fact that the robot falls down can be used in the design of new motions and specifically in GA?   The demo should show several new motions automatically created by the new software and the new editor to design new motions in a run. The final lecture should show that students have an advanced understanding of iSOBOT programming and evolutionary computing in robot applications. | **TASK 1:** Learn all commands and modes of iSOBOT. Design by hand some new motions: pushups, tai-chi, Swedish gymnastics, dance, speech with gesticulation.  **TASK 2:** Learn about Genetic algorithm. Encode all iSobot commands as symbols. Use your GA (from Internet or written from scratch) to develop new motions of iSOBOT from the old ones and from random base of motions generated randomly. Test these new automatically generated motions on the robot, inputting them by hand.  **TASK 3:** Design a graphic editor interface to your GA program that will allow to design motions as strings of characters.  **Task 4.** Create a data base of motions. The editor should allow to give information about these motions to the user, call them and modify.  **Task 5.** According to the theory that I will give you, design new crossover and mutation operators, especially tuned to iSOBOT.  **Task 6**. Test your editor on the new interface designed in this class that would allow you to directly load motions to the robot.  **HELP.** Mathias Sunardi,  You can re-use any editor that we designed in the class or that you find on internet. Just write that you do so. Quay Williams, Mathias Sunardi, ESRA and Hear-To-Heart are good examples. |
| **1** | Aditya Bhutada.  Work closely with groups 7 and 5 | Genetic algorithm for iSobot. New operators. Feedback from audience by phones. Design of the editor program. | Graduate. Verilog etc. |
| **Group 7.**  **KHR-1 and**  **Emotion-based Motion**  **Transformations.** | 1. KHR-1 robot from Mathias Sunardi or Marek Perkowski. 2. Laptop or PC 3. Visual C, Visual Basic, Python or any software. Best to extend Mathias’ work | The task of this project is the design of smooth continuous motions automatically using a modified GA with new operators.  The task of this project is to investigate the following research issues (perhaps for the first time, check on internet the previous work on GA for KHR).   1. Can GA be used to invent new motions for KHR1? 2. What are the best GA parameters for this task? 3. Can we add new operators to GA that will allow more interesting or better in some way motions for GA? 4. How the fact that the robot falls down can be used in the design of new motions and specifically in GA?   The demo should show several new motions automatically created by the new software and the new editor to design new motions in a run. The final lecture should show that students have an advanced understanding of KHR-1 programming and evolutionary computing in robot applications. The students should have some understanding of spectral methods and Fourier analysis in robot-motion generation and transformation applications. | **TASK 1:** Learn all commands of KHR-1. Design by hand some new motions: pushups, tai-chi, Swedish gymnastics, dance, speech with gesticulation. **Communicate and get help from Mathias Sunardi.**  **TASK 2:** Learn about Genetic algorithm and evolutionary computation. Encode all KHR-1 commands as genes in chromosomes. Use your GA (from Internet or written from scratch) to develop new motions of JHR-1 from the old ones and from random base of motions generated randomly. Test these new automatically generated motions on the robot, inputting them by hand.  **TASK 3:** Modify the graphic editor interface of Mathias Sunardi to include your GA program that will allow to design motions.  **Task 4.** Create a data base of motions. The editor should allow to give information about these motions to the user, call them and modify.  **Task 5.** According to the theory from MS thesis of Mathias Sunardi, design new crossover and mutation operators, especially tuned to KHR-1.  **Task 6**. Test your editor by directly loading motions to the KHR-1 robot.  **HELP.** Mathias Sunardi,  You can re-use parts of any editor that we designed in the class or that you find on internet. Just write that you do so. Quay Williams, Mathias Sunardi, ESRA and Hear-To-Heart are good examples. |
| **1** | Shubham Chopra **group leader**  **shubham\_24@hotmail.com** | Spectral theory of motion. Interpolation schemes for motion. Generators of motion. | **Undergraduate**.C, C++, Matlab, follower robot, parallel port. Robot competitions. Programing robots interest. |
| **2** | Nimrod Mc Dade  [nimrod@comcast.net](mailto:nimrod@comcast.net) | Improving and extending motion editor | Undergraduate Senior. C, ASSM, Visual Basic. |
| **3** | Charlie Wu  [charliewu123@gmail.com](mailto:charliewu123@gmail.com) | Generators of motion. Design, programming and test of groups of symbolic motions. Creating a library of motions | Undergraduate. C, C++, Java, VBasic, 3D games, Direct X, 3-D modeling. XML. Programing robots interest. |
| **4** | Allen Grimm  **Arm design leader and Group Leader.** grimm@pdx.edu | Arm concept. Artificial Intelligence and system architecture of the entire project | **Graduate.** Lego. Other robotics. |
| **Group 8.**  **Pioneer robot as a Greeter to PSU Visitors.**  Cancelled | 1. Pioneer PEOPLEBOT robot, previously designed woman greater. 2. Previously designed woman head (oriental). 3. Previously designed hands (several variants available) | This robot will concentrate on actions of a greeting female person, like a receptionist that will have entertaining gestures and dialog about how great is to be a PSU student. This robot was designed twice in the past but this time it must be more reliable and hopefully more intelligent. Look to previous class webpages under PEOPLEBOT or Greeter. Students who worked on this project before include: Jeff Allen, Jake Biamonte, Uland Wong, Alex Perez, Stefan Gebauer, Normen Giesecke, ADD MORE students. | **Research and Development Topics:** Programming in C/C++. Vision. Speech and dialog with Alice. Integration of the system.  **Topic 1. MOBILE BASE.** Assemble again the three components of Pioneer Robot from Activmedia as they were in original robot mobile base. Test how the original software for obstacle avoidance works. Try to understand what this software does and how it can be improved/modified for our purposes. **Help** **from:** last class. I will insert names later on.  **Topic 2. HANDS.** Find the woman body that was used by previous students. Find hands of the last class. Put it together as it was. Be sure the rotation in waist works. Test hand motions. Possibly modify arms and palms. **Help from:** previous class. The hands can be also taken from the Bohr robot which is in the lab. Remember that in this project hands are only to show gestures, they do not need to be able to grab items.  **Topic 3. HEAD.** Find the oriental female head in the adjacent room (Hawash knows). Head designer David-Ng <david-ng@shaw.ca> Find documentation to this head. Test on existing software. Add more speech dialogs about how great is to study at PSU or whatever you like. Program new facial gestures. **Help** **from:** Martin Lukac. Use ESRA software for speech with facial gestures. You need also a woman voice to record. **Help from** Arushi Raghuvanshi  **Topic 4.** Integration of hands, head, body and mobile base. ***Maher Hawash – leader with everybody helping.***  **Topic 5.** Add behaviors starting from sensors and ending with speech patterns and gestures. Show a complete demo of the robot on the corridor and meeting area in new engineering building.  ***Topic 6*** *(next quarter).*  *Vision (reused), speech communication based onAlice (reused). Speech recognition (reused). Integration of speech and vision to the robot.* |
| **Group 9.**  **Advanced hand/palm design for a humanoid robot.** |  |  | 1. Should find similar projects on Internet. Including my webpage. Michele Folgheraiter. 2. Read about human palm, hand and controls/muscles. 3. Propose the kinematics design. 4. Compare to previous hands from the class. Should be better in at least one respect. 5. Mechanical design and control of simple motions from RobotC. No advanced programming, only basic motions for each DOF. |
| **1** | Lillian Olney  Integrate with robot DIM  teknosushii@gmail.com | Design a complete arm for the palm that I showed in class. Where to put servos, how to control fingers, how many degrees of freedom. Your demo should only show that you can reliably control fingers, palms and arm. The minimum is 2 DOF for shoulder, 3 DOF for arm/hand and 3 DOF for fingers. Should be similar to humans in its motions. I will show you videos. |  |
| **Group 10.**  **Highway Safety Robot** |  |  |  |
| **1** |  |  |  |