KIT-Department of Informatics
Prof. Dr.-Ing. Tamim Asfour

Exam Solution Sheet

Robotics III - Sensors and Perception in Robotics

October 2, 2020, 08:00 – 09:00

Family name:	Given name:	Matriculation number:
Bond	James	007

Question 1	10 out of 10 points
Question 2	9 out of 9 points
Question 3	8 out of 8 points
Question 4	10 out of 10 points
Question 5	8 out of 8 points

Total:	45 out of 45 points

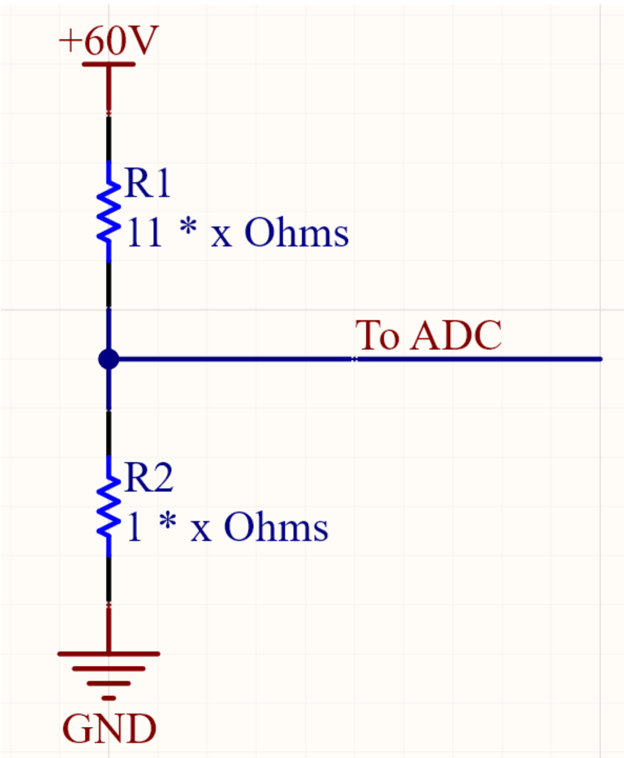
	Grade: **1,0**

Exercise 1 *Internal Sensors* (10 points)

1. - *high precision closed loop position and velocity control* → The motors need encoders to measure position and velocity.
   - *ready to be used by the software framework immediately* → No homing is allowed, so we need an absolute position encoder. Since the position control is required to be precise, we choose a magnetic off axis absolute encoder, as this is known from the lecture to be the most precise one.
   - *The arms should hence be able to precisely sense the torques* → We need torque sensors in each joint. The most precise known from the lecture are the strain gauge based sensors.
   - *At the same time the actuator units should be as small as possible* → Since the strain gauge sensors are also the smallest type according to the lecture, we choose this design.

2. (a) We choose the accelerometer, since we can calculate the slope from the gravity vector given by the accelerometer.

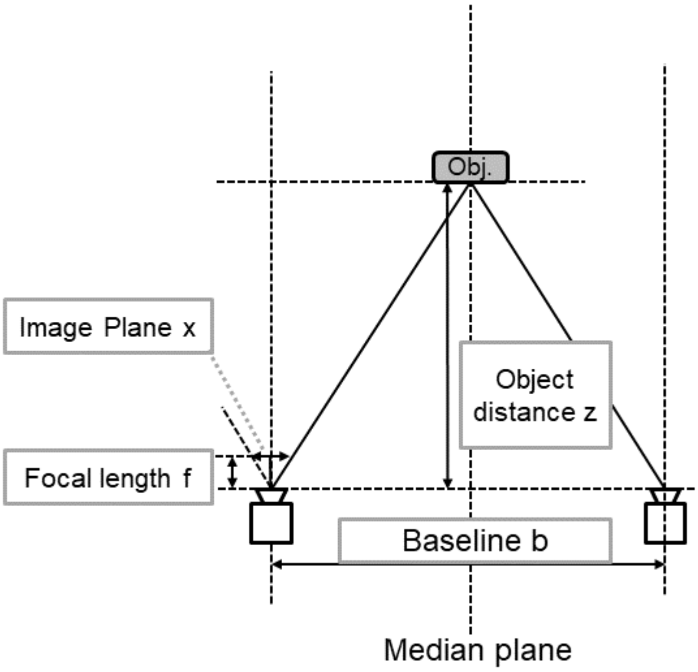
   (b) The accelerometer will not work well when the robot accelerates. A gyroscope will allow to refine the measurement by integration of the angular rate and using a Complementary filter.



3. The schematic shows a voltage divider. Since no value for the resistance was given in the question, we will assume $1\,\Omega$ for R_2 and use the formula $R_2 = \frac{U_2}{U}(R_1 + R_2)$ from the lecture where $U = 60$ Volt and $U_2 = 5$ Volt. This resolves to $R_1 = 11$ Ohms.

Exercise 2 *External Sensors* (9 points)

1. Stereo camera:



2. Disparity

$$d = \frac{b}{z} \times f = \frac{15 \times 10^{-2}\,\mathrm{m}}{1.5\,\mathrm{m}} \times 6 \times 10^{-3}\,\mathrm{m} = 0.6 \times 10^{-3}\,\mathrm{m}$$

3. Required bandwidth for stereo system: $2 \times 60\,\text{frames/s} \times 1000 \times 1000\,\text{pixels/frame} \times 24\,\text{bit/pixel} = 2 \times 1440\,\text{Mbits/s}$

   $1440\,\text{Mbits/s} > 430\,\text{Mbits/s}$ that are available $\Rightarrow$ It is not possible to connect one or two of the cameras to the bus.

4. Stereo vision problematic:

   - Homogeneous surfaces, such as a white wall (absence of features)
   - Objects far from camera (only good for mid-range distances)

   Active depth problematic:

   - IR-depth camera in outdoor environment not usable due to high intensity IR radiation in sunlight

5.
$$d = \frac{c\,\Delta t}{2}$$

   c: signal propagation speed / speed of light

6. RADAR sensor, ultrasonic sensor

Exercise 3 *Feature Extraction* (8 points)

1. Point of Interest Detection

   (a) Methods :

   - Moravec Operator
   - Harris Corner
   - Canny Edge Detector
   - Scale Space Extrema
   - Maximally Stable Regions
   - Neural Networks
   - others are possible

   (b) Descriptors:
   Points of Interest only mark an x-y-coordinate in an image. This is not enough to identify features in images, as there is no invariance w.r.t. rotation, translation, scaling, lighting conditions, etc.
   Examples are: SIFT, SURF, MSER, etc..

2. Pose Estimation

   (a) Object Detection:

   - Brute Force
   - Nearest Neighbors
   - RANSAC

   (b) Methodology:
   Different solutions are possible:

   - Monocular: any Method that solves the "Perspective n-Point" Problem, i.e. 2D-3D correspondences are necessary. The resulting system of linear equations can be used to estimate the pose.
   - Stereo Camera System: use correlation and stereo triangulation, i.e. identify the corresponding features in both images and calculate the world coordinates of the features. Afterwards, the locations of the features of known objects need to be fitted to the estimated feature positions to calculate the pose
   - RGBD: The features are already projected to 3D space when using point cloud, therefore no 2D-3D problem needs to be solved. The locations of the features can be directly used to estimate the pose of the object (through fitting).

3. RGB-D Pose Estimation:
   ICP, RANSAC or Neural Networks can be used to estimate the pose of the objects from depth data only.

   Point Clouds are unordered, therefore standard computer vision approaches cannot be used for them (i.e. supervised learning methods have a hard time with unordered sets)

Exercise 4 *Scene Understanding* (10 points)

1. (a) Point cloud of the objects: 1.5 p.

    - Option 1:
        - In the RGB image, segment single-colored table via *either* MSER (Maximally Stable Extremal Regions) *or* HSV color histograms .
        - Remove points corresponding to segmented pixels from point cloud.
    - Option 2:
        - Fit plane to table using RANSAC.
        - Remove plane inliers.
    - Notes:
        - Changing lighting conditions, color not known $\Rightarrow$ RGB color threshold/histograms are not sufficient.
        - RGB-D camera $\Rightarrow$ known pixel-point correspondence

   (b) Number of objects and centroids: 1.5 p.

    - Segment the remaining point cloud using {Euclidean Clustering, Region Growing, LCCP, PointNet}.
    - Number of objects = number of segments.
    - Object centroids: mean of points in segment *or* center of axis-aligned bounding box (AABB) of segment

2. Method and strategy: 2 p.

    - Method: Support relations between objects / support graph
    - Strategy:
        - Remove objects which are supported by eggs until eggs does not support any objects.
        - Do not grasp objects which are supporting other objects.
        - If the support graph contains a cycle: *safe bimanual manipulation*, i.e. secure supported object with one hand while grasping it with the other.

3. (a) Six spatial relations: 2 p.

    - Fork *left of* Plate / Plate *right of* Fork
    - Knife *right of* Plate / Plate *left of* Knife
    - Scrambled Eggs *on* Plate / Scrambled Eggs *inside* Plate / Plate *under* Scrambled Eggs
    - Fork *on* Table / Table *under* Fork
    - Plate *on* Table / Table *under* Plate
    - Knife *on* Table / Table *under* Knife

   (b) Discriminative and generative models of spatial relation: 3 p.

    - Discriminative:
        - Classifies which relations are present in a scene.
        - Input: Scene / object locations

  - Output: Relations (present/not present)
- Generative:
  - Generates scenes with given relations.
  - Input: Initial scene/objects, relations
  - Output: New scene / object location(s)

Exercise 5 *Active Vision and Gaze Stabilization* (8 points)

1. Yarbus key observation: 2 p.
   - Patterns of eye movements are similar (but not identical) when different people view the same painting or when a single individual was shown the same painting a number of times, with between one and two days separating the recording
   - Different pattern of eye movements depending on the task
2. Reafference Principle: 3 p.
   - Description
     - Receive copies of the motor commands (efference copy)
     - Predict the expected sensory outcome of self-induced motions (predicted reafference)
     - These reafferences are then subtracted from the actual sensor measurements
     - Thus isolating the sensory consequences of externally induced perturbations (called exafference)
   - Terms and Relations:
     - afference = measured sensor values
     - reafference = predicted sensor values
     - exafference = afference - reafference

   Example: tickling
3. Gaze stabilization

   (a) Vestibulo-Ocular Reflex (VOR). Advantages: High Sampling Rate and easy to implement. Limitations: Controls only the eye joints, cannot compensate for external perturbations and requires an IMU.

   (b) Optokinectic reflex (OKR). Advantages: can stabilize the image in a dynamic environment. Limitations: low input frequency and controls only the eye joint.

   (c) Inverse kinematics (IK). Advantages: fast. Controls both head and eye joints Limitations: Requires a target point Can only measure and thus stabilize self-induced perturbations Depends on the accuracy of the robot's kinematic model

   The OKR is the most versatile and can be used for all scenarios. A combination of different gaze stabilization methods is possible.