UAS ROBOTIKA DAN SISTEM CERDAS

TECHNICAL REPORT

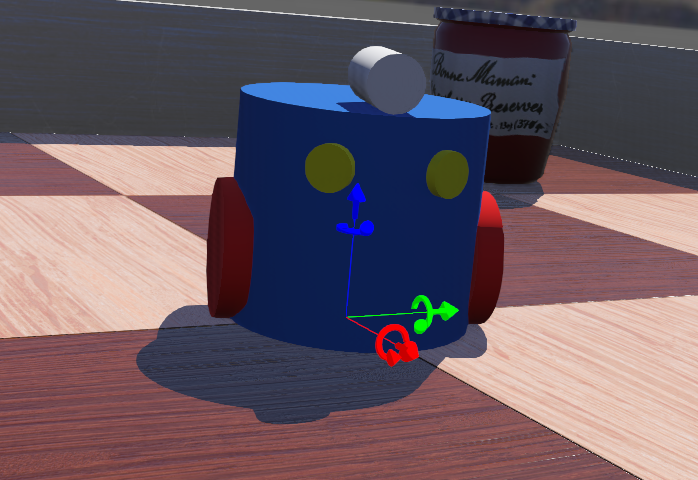
HACKING PROGRAAMING BOOK

Nama: Jean Jeasen T

Nim: 1103201257

Kelas:TK43G08

Topic; Camera recognizing



Robot Camera recognizing ini berfungsi untuk mengetahui lingkungan sekitar nya ada apa saja , yang berada di sekitar.

Source code:

|  |
| --- |
| #include <stdio.h>  #include <webots/camera.h>  #include <webots/camera\_recognition\_object.h>  #include <webots/motor.h>  #include <webots/robot.h>  #define SPEED 1.5  #define TIME\_STEP 64  int main() {  WbDeviceTag camera, left\_motor, right\_motor;  int i, j;  wb\_robot\_init();  /\* Get the camera device, enable it and the recognition \*/  camera = wb\_robot\_get\_device("camera");  wb\_camera\_enable(camera, TIME\_STEP);  wb\_camera\_recognition\_enable(camera, TIME\_STEP);  /\* get a handler to the motors and set target position to infinity (speed control). \*/  left\_motor = wb\_robot\_get\_device("left wheel motor");  right\_motor = wb\_robot\_get\_device("right wheel motor");  wb\_motor\_set\_position(left\_motor, INFINITY);  wb\_motor\_set\_position(right\_motor, INFINITY);  /\* Set the motors speed \*/  wb\_motor\_set\_velocity(left\_motor, -SPEED);  wb\_motor\_set\_velocity(right\_motor, SPEED);  /\* Main loop \*/  while (wb\_robot\_step(TIME\_STEP) != -1) {  /\* Get current number of object recognized \*/  int number\_of\_objects = wb\_camera\_recognition\_get\_number\_of\_objects(camera);  printf("\nRecognized %d objects.\n", number\_of\_objects);  /\* Get and display all the objects information \*/  const WbCameraRecognitionObject \*objects = wb\_camera\_recognition\_get\_objects(camera);  for (i = 0; i < number\_of\_objects; ++i) {  printf("Model of object %d: %s\n", i, objects[i].model);  printf("Id of object %d: %d\n", i, objects[i].id);  printf("Relative position of object %d: %lf %lf %lf\n", i, objects[i].position[0], objects[i].position[1],  objects[i].position[2]);  printf("Relative orientation of object %d: %lf %lf %lf %lf\n", i, objects[i].orientation[0], objects[i].orientation[1],  objects[i].orientation[2], objects[i].orientation[3]);  printf("Size of object %d: %lf %lf\n", i, objects[i].size[0], objects[i].size[1]);  printf("Position of the object %d on the camera image: %d %d\n", i, objects[i].position\_on\_image[0],  objects[i].position\_on\_image[1]);  printf("Size of the object %d on the camera image: %d %d\n", i, objects[i].size\_on\_image[0], objects[i].size\_on\_image[1]);  for (j = 0; j < objects[i].number\_of\_colors; ++j)  printf("- Color %d/%d: %lf %lf %lf\n", j + 1, objects[i].number\_of\_colors, objects[i].colors[3 \* j],  objects[i].colors[3 \* j + 1], objects[i].colors[3 \* j + 2]);  }  }  wb\_robot\_cleanup();  return 0;  } |

Diatas ini adalah source code yang dibutuhkan untuk memoeritahkan camera untuk menganalisa lingkungan sekitar nya .

Berikut adalah penejlasan untuk source code nya

|  |
| --- |
| Inisialisasi perangkat: Pada awal program, perangkat-perangkat seperti kamera dan motor diinisialisasi menggunakan fungsi wb\_robot\_get\_device().  Pengaktifan kamera: Kamera diaktifkan menggunakan fungsi wb\_camera\_enable(), yang mengaktifkan kamera dengan interval waktu TIME\_STEP. Ini memungkinkan perangkat kamera mengambil citra secara periodik.  Pengaktifan pengenalan objek kamera: Pengenalan objek kamera diaktifkan menggunakan fungsi wb\_camera\_recognition\_enable(). Ini memungkinkan robot untuk mengenali objek-objek yang terlihat melalui kamera.  Loop utama: Di dalam loop utama, yang dijalankan dengan wb\_robot\_step(TIME\_STEP), objek-objek yang terlihat oleh kamera diambil dan ditampilkan informasinya.  Mendapatkan jumlah objek yang terkenali: Dengan menggunakan wb\_camera\_recognition\_get\_number\_of\_objects(), program mengambil jumlah objek yang dikenali oleh kamera.  Mendapatkan informasi objek: Dengan menggunakan wb\_camera\_recognition\_get\_objects(), program mendapatkan informasi detail tentang setiap objek yang terkenali, seperti model objek, ID objek, posisi relatif objek, orientasi relatif objek, ukuran objek, posisi objek di gambar kamera, ukuran objek di gambar kamera, dan informasi warna objek.  Menampilkan informasi objek: Informasi objek yang diterima ditampilkan melalui printf() untuk masing-masing objek yang terkenali.  Proses ini diulang terus menerus selama simulasi berjalan, dan program diakhiri dengan membersihkan perangkat-perangkat yang digunakan melalui wb\_robot\_cleanup() |