

NAMA : Ratika Dwi Anggraini

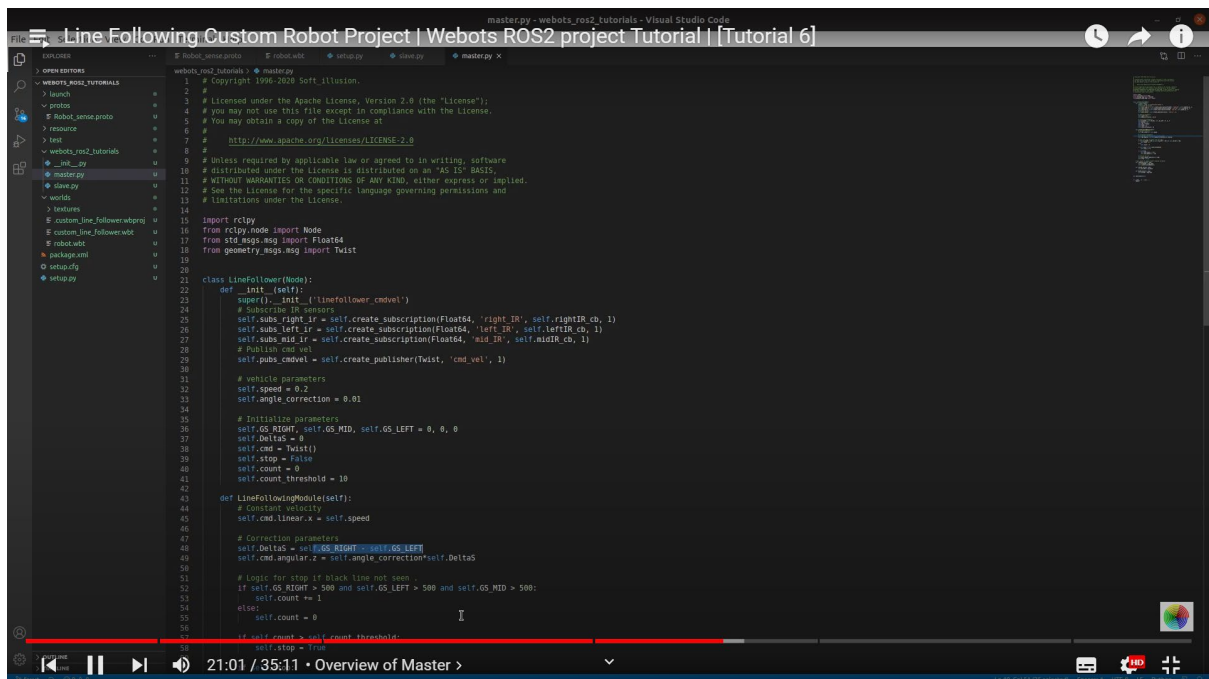
NIM : 1103201250

TUGAS 5 ROBOTIKA

LINE FOLLOWING CUSTOM ROBOT PROJECT

WEBOTS ROS2 PROJECT TUTORIAL

Dalam video tutorial keenam dari seri v-bots, pembuat konten memperkenalkan konsep teknik master-slave. Proyek yang diimplementasikan melibatkan robot kustom yang mengikuti garis menggunakan tiga sensor jarak. Video memberikan gambaran umum tentang proyek tersebut, dengan node master yang berlangganan topik sensor dan melakukan perhitungan untuk menghasilkan perintah kecepatan. Sementara itu, node slave melakukan penerbitan dan berlangganan topik untuk berinteraksi dengan v-bots. Tutorial ini mengajak pemirsa untuk menerapkan pengetahuan yang telah dipelajari dari tiga video sebelumnya, yang mencakup pembuatan layanan, penerbit, dan pelanggan. Pemirsa juga diberikan akses ke file proyek sehingga dapat bereksperimen tanpa harus menulis kode. Pembuat konten juga menjelaskan cara menambahkan sensor jarak pada robot kustom dan memperkenalkan konsep proto untuk memudahkan penggunaan robot dalam berbagai dunia. Pada bagian selanjutnya, tutorial membahas pembuatan node slave, yang bertanggung jawab untuk mengelola sensor dan mengatur kecepatan roda berdasarkan perintah yang diterima.



```
Line Following Custom Robot Project | Webots ROS2 project Tutorial | [Tutorial 6]
# Copyright 1996-2020 Soft Illusion.
# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
# http://www.apache.org/licenses/LICENSE-2.0
# Unless required by applicable law or agreed to in writing, software
# distributed under the license is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.

import rospy
from rospy import Node
from std_msgs.msg import Float64
from geometry_msgs.msg import Twist

class LineFollower(Node):
    def __init__(self):
        super().__init__('linefollower_cmdvel')
        # Subscribe to sensors
        self.subs_right_ir = self.create_subscription(Float64, 'right_IR', self.rightIR_cb, 1)
        self.subs_left_ir = self.create_subscription(Float64, 'left_IR', self.leftIR_cb, 1)
        self.subs_mid_ir = self.create_subscription(Float64, 'mid_IR', self.midIR_cb, 1)
        # Publish cmd vel
        self.pubs_cmdvel = self.create_publisher(Twist, 'cmd_vel', 1)

        # Vehicle parameters
        self.speed = 0.2
        self.angle_correction = 0.01

        # Initialize parameters
        self.gos_right, self.gos_mid, self.gos_left = 0, 0, 0
        self.deltas = 0
        self.cmd = Twist()
        self.stop = False
        self.count = 0
        self.count_threshold = 10

    def LineFollowingModule(self):
        # Constant velocity
        self.cmd.linear.x = self.speed

        # Correction parameters
        self.deltas = 0
        self.cmd.angular.z = self.angle_correction * self.deltas

        # Logic for stop if black line not seen
        if self.gos_right > 500 and self.gos_left > 500 and self.gos_mid > 500:
            self.count += 1
        else:
            self.count = 0

        if self.count > self.count_threshold:
            self.stop = True
```

Pada bagian terakhir dari tutorial ini, kita melihat bagaimana menulis node master. Node ini berperan sebagai "pikiran" dari sistem, yang berlangganan ke semua sensor dan membuat keputusan mengenai arah yang harus diambil oleh robot, kemudian mempublikasikannya ke topik command velocities. Dalam node master ini, terdapat beberapa variabel dan logika untuk mengatur kecepatan linear dan sudut robot berdasarkan perbedaan pembacaan sensor kanan dan kiri. Terdapat juga logika untuk menghentikan robot jika tidak mendeteksi garis

hitam, dan robot akan kembali bergerak ketika garis hitam terdeteksi kembali. Kemudian, tutorial membahas file launch yang digunakan untuk memulai proyek dan menyiapkan dunia simulasi. Setelah itu, penjelasan tentang setup.pi dijelaskan, yang memuat berbagai konfigurasi dan parameter untuk memastikan proyek dapat berjalan dengan baik. Setelah menjelaskan teori, penonton diajak untuk melakukan build proyek dan menjalankannya menggunakan perintah ROS2. Terakhir, demonstrasi singkat dilakukan dengan menunjukkan hasil simulasi robot yang mengikuti garis hitam berkat logika yang telah diimplementasikan dalam node master dan slave.