

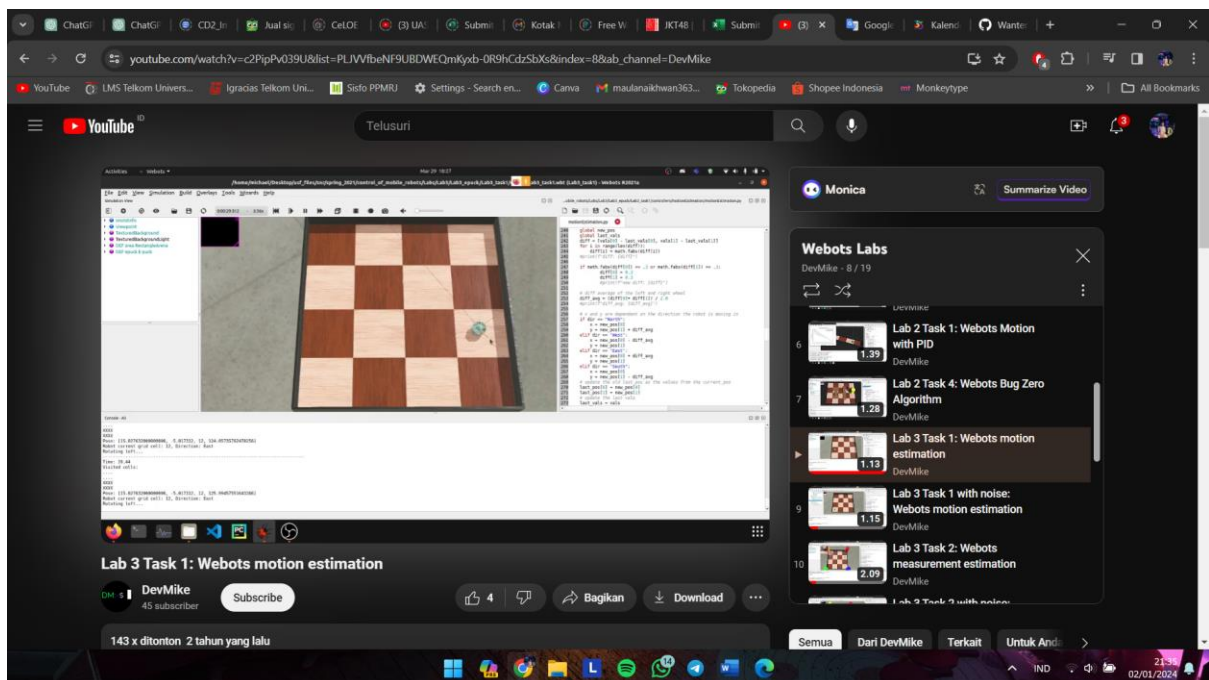
MUHAMMAD IKHWAN MAULANA

1103203219

### Playlist Path Planning 8-19

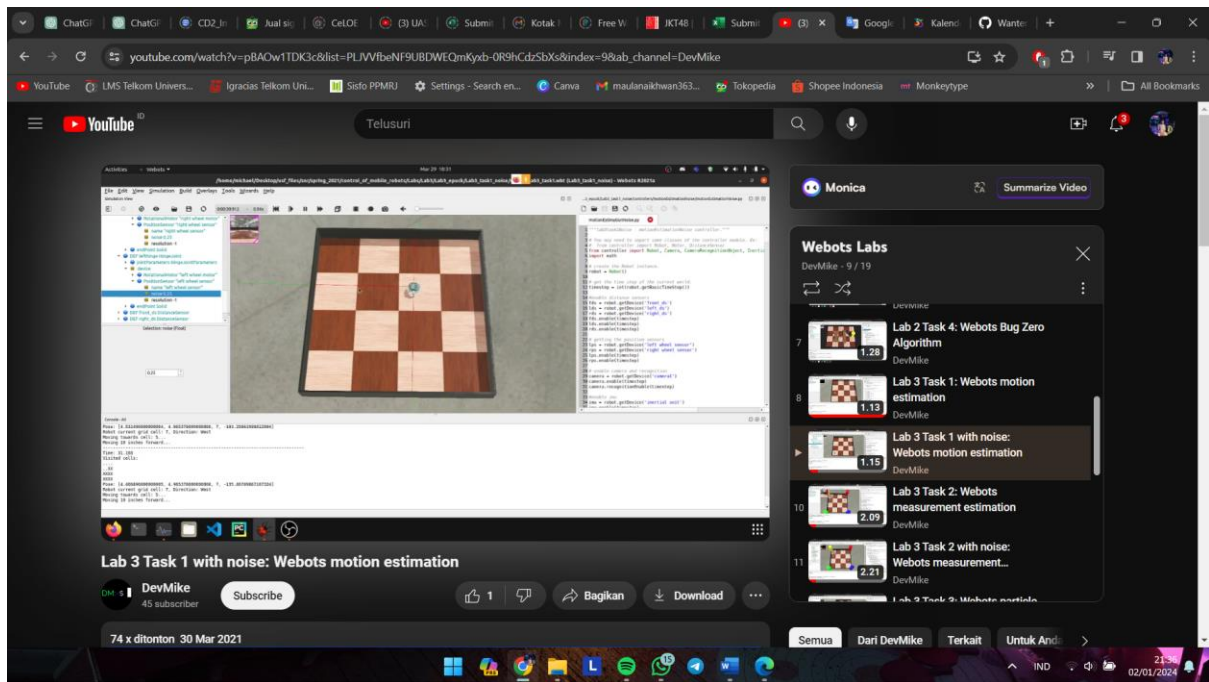
- Lab 3 Task 1: Webots motion estimation

Robot ini menggunakan sensor posisi untuk mencari perbedaan pembacaan saat ini dan pembacaan sebelumnya untuk menentukan jarak linier yang telah ditempuh dari titik awalnya di sel grid 16 mulai dari koordinat 15.0, -15.0. Dari sini, robot dapat melacak seberapa jauh ia telah bergerak dari titik awal untuk melacak koordinatnya guna menentukan di kotak mana ia berada, dan dapat terus bergerak hingga mengunjungi setiap kotak.



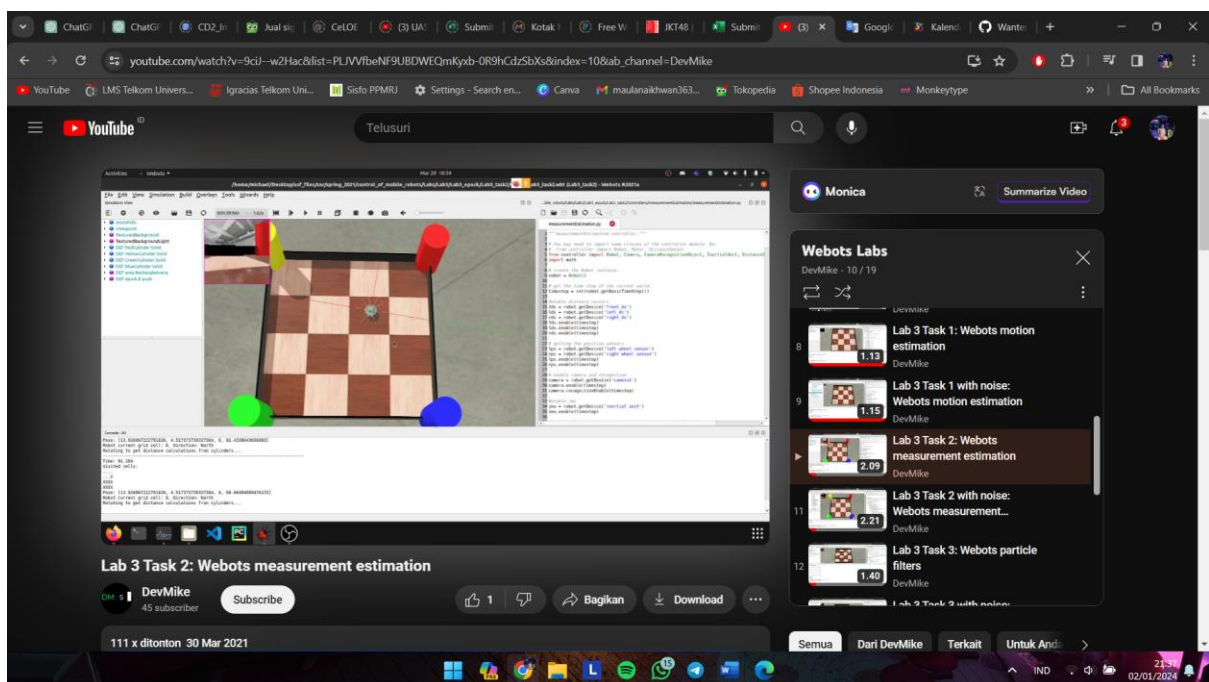
- Lab 3 Task 1 with noise: Webots motion estimation

Robot ini juga mengalami kebisingan pada unit pengukuran inersia dan sensor jaraknya, karena robot ini terus melacak bagaimana cara menambahkan perbedaan nilai pada koordinat x dan y berdasarkan arah yang dituju, perlu ada solusi bagi IMU untuk melakukannya. menentukan apakah ia bergerak ke Utara, Timur, Selatan, atau Barat. Saya memecahkan masalah ini dengan membuat kamus python dirs = {"North": 0, "West": 0, "East": 0, "South": 0}, untuk setiap langkah gerak saya akan menambahkan arah sesuai dengan Nilai IMU, saya kemudian akan berasumsi bahwa robot sedang menuju ke arah yang memiliki hitungan tertinggi, melakukan hal ini memecahkan masalah saya tentang arah mana yang dituju dari satu sel ke sel lainnya dan dengan demikian memecahkan masalah koordinatnya juga! Menyetel ulang kamus untuk setiap arah baru yang coba dilalui juga sangat penting untuk mencegah estimasi arah yang miring.



- Lab 3 Task 2: Webots measurement estimation

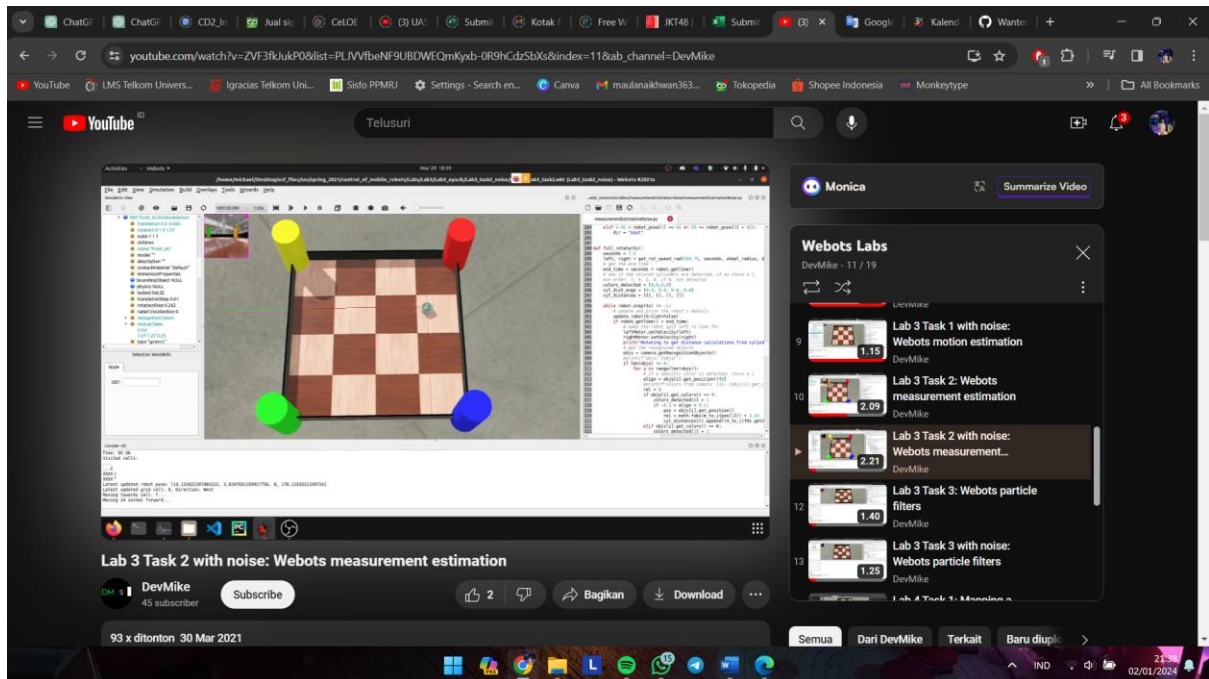
Scrip Webots ini mendemonstrasikan cara menggunakan trilaterasi di setiap sel untuk menentukan lokasi robot di peta grid. Robot ini berputar 360 derajat di setiap sel untuk mendeteksi keempat silinder dan menggunakan sistem persamaan dan trilaterasi untuk menentukan posisi robot di peta grid.



- Lab 3 Task 2 with noise: Webots measurement estimation

Dunia Webots ini memperkenalkan kebisingan di kedua sensor jarak yang membuatnya lebih sulit untuk memperkirakan seberapa jauh jarak robot dari setiap silinder. Untuk menyiasatinya, saya menggunakan jarak relatif kamera, meratakannya dengan nilai sensor jarak depan, lalu merata-ratakan banyak nilai tersebut untuk setiap silinder untuk mendapatkan radius dari robot ke setiap silinder. Hal ini memungkinkan pembacaan sensor yang lebih akurat. Selain itu, saya juga mengubah fungsi

trilaterasi saya untuk menggunakan 3 silinder terjauh dari robot untuk menghindari penggunaan jarak yang terlalu dekat dengan silinder untuk mendapatkan pembacaan yang akurat.



- Lab 3 Task 3: Webots particle filters

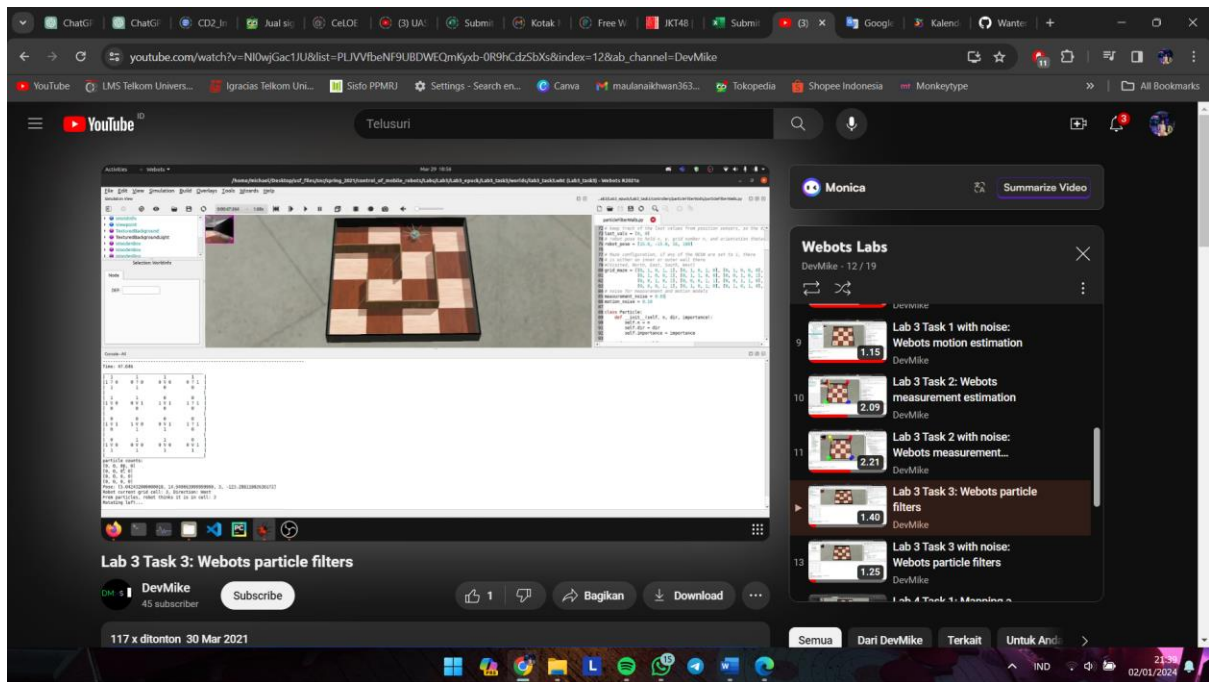
Skrip ini akan menginstruksikan robot untuk melintasi labirin menggunakan algoritma yang saya buat untuk menyelesaikan labirin ini dan masa depan. Itu juga menerapkan filter partikel yang terdiri dari 80 partikel. Ia akan melihat dinding di sekelilingnya, dan melihat gerakan mana yang mungkin dilakukan, ke kiri, ke depan, atau ke kanan. Ia kemudian akan melakukan estimasi pengukuran pada semua partikel dengan menggunakan dinding yang baru saja didapatnya.

Selanjutnya, fungsi ini akan mendapatkan bobot setiap sel, menormalkannya, dan kemudian mengambil sampel ulang partikel berdasarkan bobot yang dinormalisasi tersebut. Robot kemudian akan bergerak terlebih dahulu, melihat apakah sel kiri tersedia, jika ada maka akan bergerak ke kiri. Jika tidak ada belok kiri maka akan maju jika tersedia. Jika jalan ke depan tidak tersedia, ia akan memeriksa apakah jalan ke kanan tersedia, jika tersedia maka ia akan berbelok ke kanan dan menuju ke sel itu. Jika tidak ada gerakan ke kiri, depan, atau kanan, ia akan menghadap ke selatan dan melihat apakah ke Selatan tersedia atau tidak, jika demikian, ia akan bergerak ke Selatan. Jika tidak, robot akan berbelok ke kiri, kini menghadap Timur, dan berpindah ke kotak kanan.

Dari titik ini, nilai kontrol disimpan dan digunakan sebagai parameter untuk fungsi motion\_update. Setelah setiap gerakan, robot akan menghadap Utara lagi dan mengulangnya hingga setiap sel dikunjungi.

Skrip ini merupakan model pengukuran 5% noise pada sensor yang akan diterapkan saat estimasi pengukuran diterapkan pada setiap partikel di setiap sel saat menentukan kemungkinan bahwa partikel tersebut menunjukkan keadaan yang sama seperti robot.

Skrip ini juga menggunakan model gerak partikel yang bergerak maju 90% dan diam 10% saat pergerakan partikel memungkinkan.



- Lab 3 Task 3 with noise: Webots particle filters

Skrip ini akan menginstruksikan robot untuk melintasi labirin menggunakan algoritma yang saya buat untuk menyelesaikan labirin ini dan masa depan. Itu juga menerapkan filter partikel yang terdiri dari 80 partikel. Ia akan melihat dinding di sekelilingnya, dan melihat gerakan mana yang mungkin dilakukan, ke kiri, ke depan, atau ke kanan. Ia kemudian akan melakukan estimasi pengukuran pada semua partikel dengan menggunakan dinding yang baru saja didapatnya.

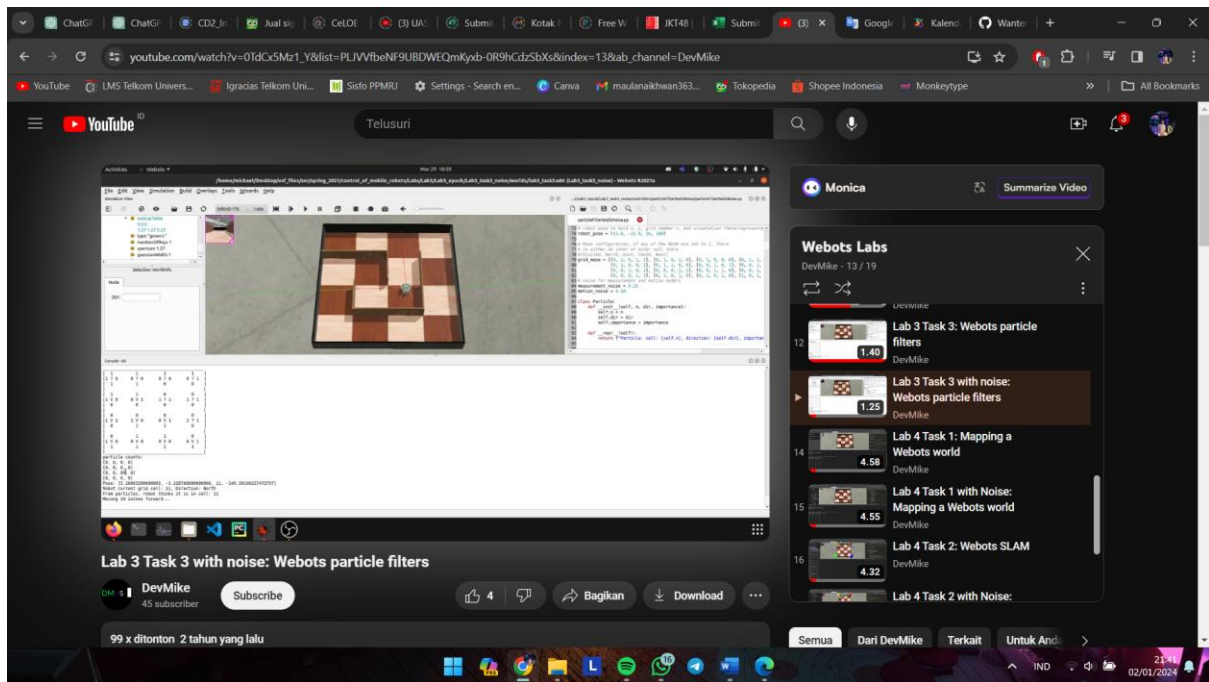
Selanjutnya, fungsi ini akan mendapatkan bobot setiap sel, menormalkannya, dan kemudian mengambil sampel ulang partikel berdasarkan bobot yang dinormalisasi tersebut. Robot kemudian akan bergerak terlebih dahulu, melihat apakah sel kiri tersedia, jika ada maka akan bergerak ke kiri. Jika tidak ada belok kiri maka akan maju jika tersedia. Jika jalan ke depan tidak tersedia, ia akan memeriksa apakah jalan ke kanan tersedia, jika tersedia maka ia akan berbelok ke kanan dan menuju ke sel itu. Jika tidak ada gerakan ke kiri, depan, atau kanan, ia akan menghadap ke selatan dan melihat apakah ke Selatan tersedia atau tidak, jika demikian, ia akan bergerak ke Selatan. Jika tidak, robot akan berbelok ke kiri, kini menghadap Timur, dan berpindah ke kotak kanan.

Dari titik ini, nilai kontrol disimpan dan digunakan sebagai parameter untuk fungsi motion\_update. Setelah setiap gerakan, robot akan menghadap Utara lagi dan mengulangnya hingga setiap sel dikunjungi.

Skrip ini merupakan model pengukuran 25% kebisingan pada sensor yang akan diterapkan saat estimasi pengukuran diterapkan pada setiap partikel di setiap sel saat menentukan kemungkinan bahwa partikel tersebut menunjukkan keadaan yang sama seperti robot.

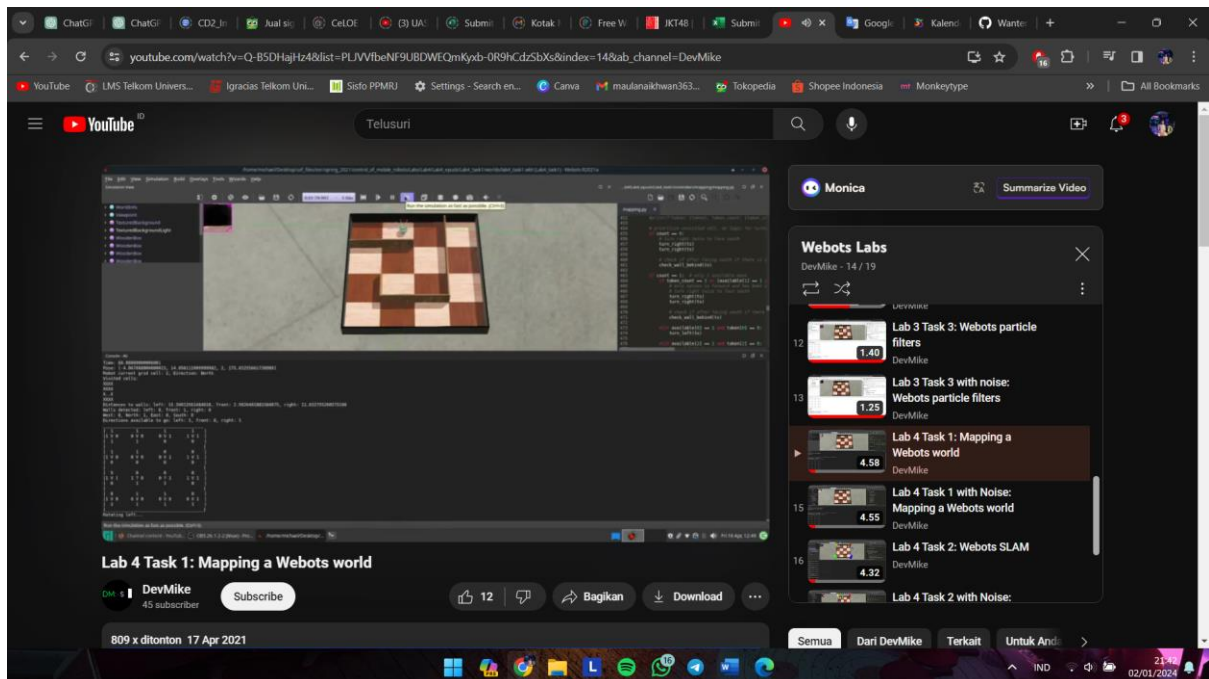
Skrip ini juga menggunakan model gerak partikel yang bergerak maju 90% dan diam 10% saat pergerakan partikel memungkinkan.





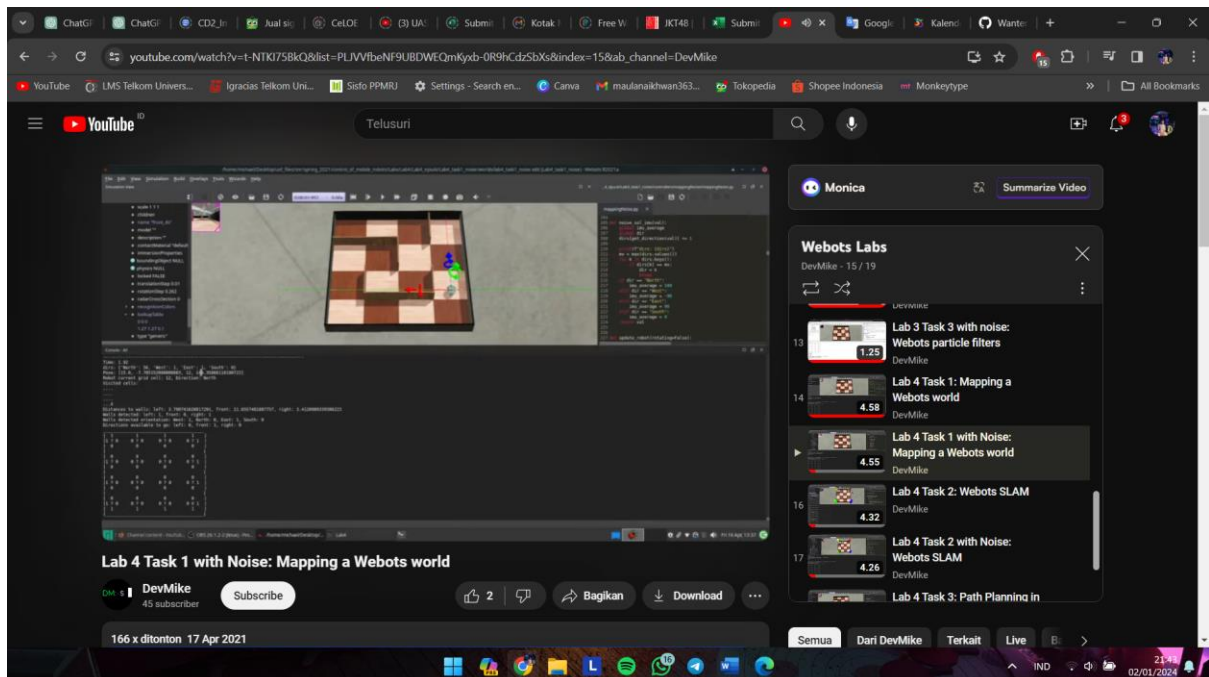
- Lab 4 Task 1: Mapping a Webots world

Tugas lab 4 ini menunjukkan bagaimana robot dapat melacak lokasinya di dunia hanya dengan mengetahui lokasi awalnya di dunia dan dapat mengumpulkan potongan-potongan dari apa yang diamati menjadi representasi berguna dari dunia yang dialaminya. Hal penting yang tidak saya sebutkan adalah bagaimana setiap kali robot mengunjungi sel, ia mendapatkan dinding dan menambahkannya ke susunan 2D sebelum membuat keputusan berikutnya, ia memerlukan informasi tersebut untuk mengetahui bahwa dinding berada di lokasi tertentu dan gerakan tertentu tidak dapat dilakukan.



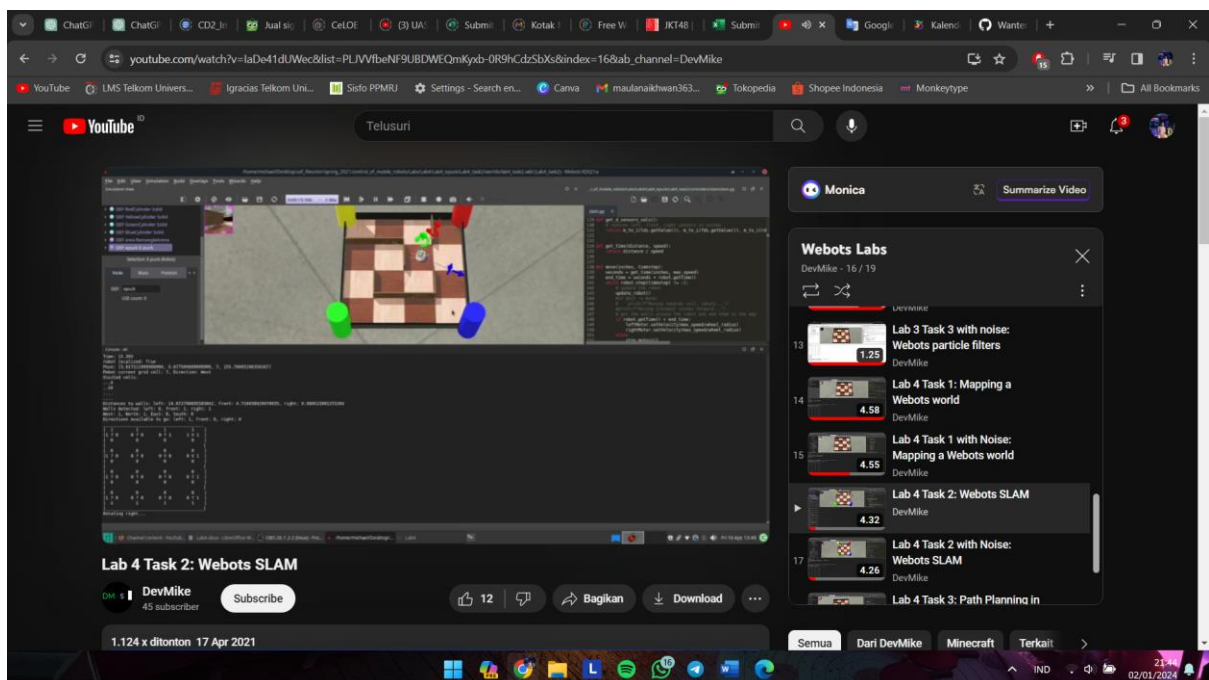
- Lab 4 Task 1 with Noise: Mapping a Webots world

Sangat mirip dengan video Lab 4 tugas 1 sebelumnya, dunia ini memiliki 10% noise pada sensor jarak, unit pengukuran inersia, dan sensor posisi. Saya menjelaskan bagaimana metode kamus saya digunakan untuk mengetahui arah utama robot.



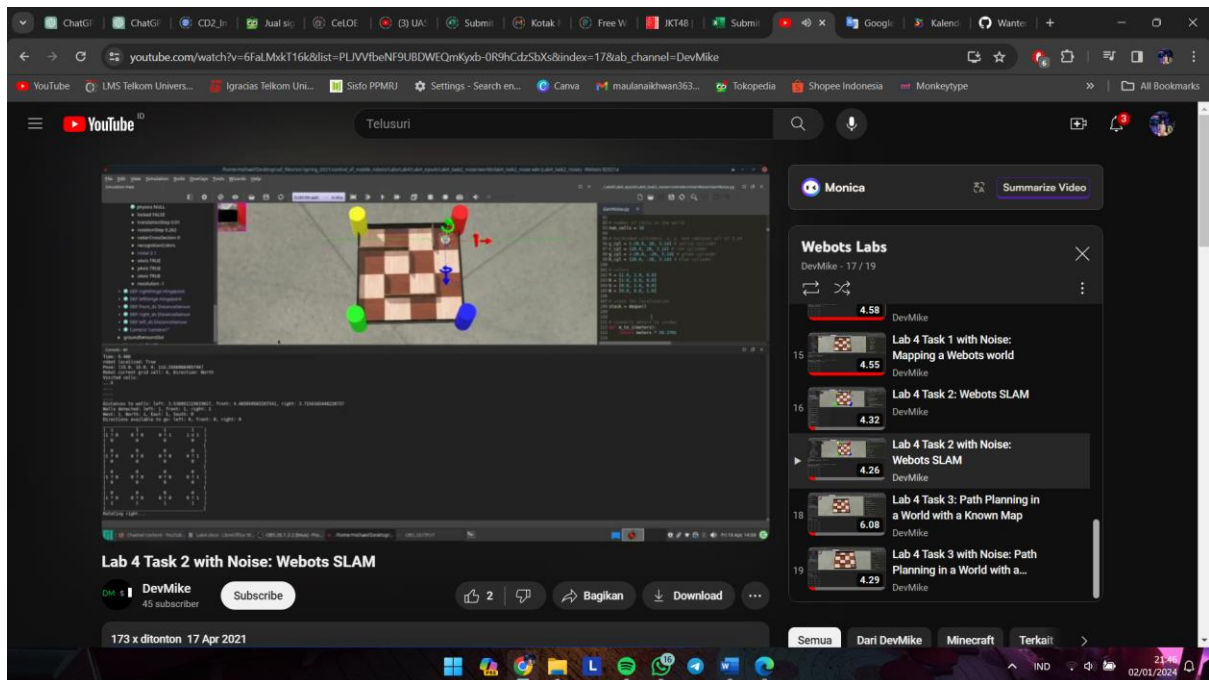
- Lab 4 Task 2: Webots SLAM

Video ini menjelaskan solusi SLAM (lokalisasi dan pemetaan simultan) saya. SLAM adalah masalah yang sangat sulit dalam robotika karena robot perlu mencari tahu di mana letaknya tanpa harus membuat peta. Cara kami menyiasatinya dalam video ini pada dasarnya adalah dengan menggunakan landmark untuk memperkirakan lokasi dunia pertama kali (lokalisasi) dan kemudian kami mulai menyatukan bagian-bagian dunia setelah itu (pemetaan).



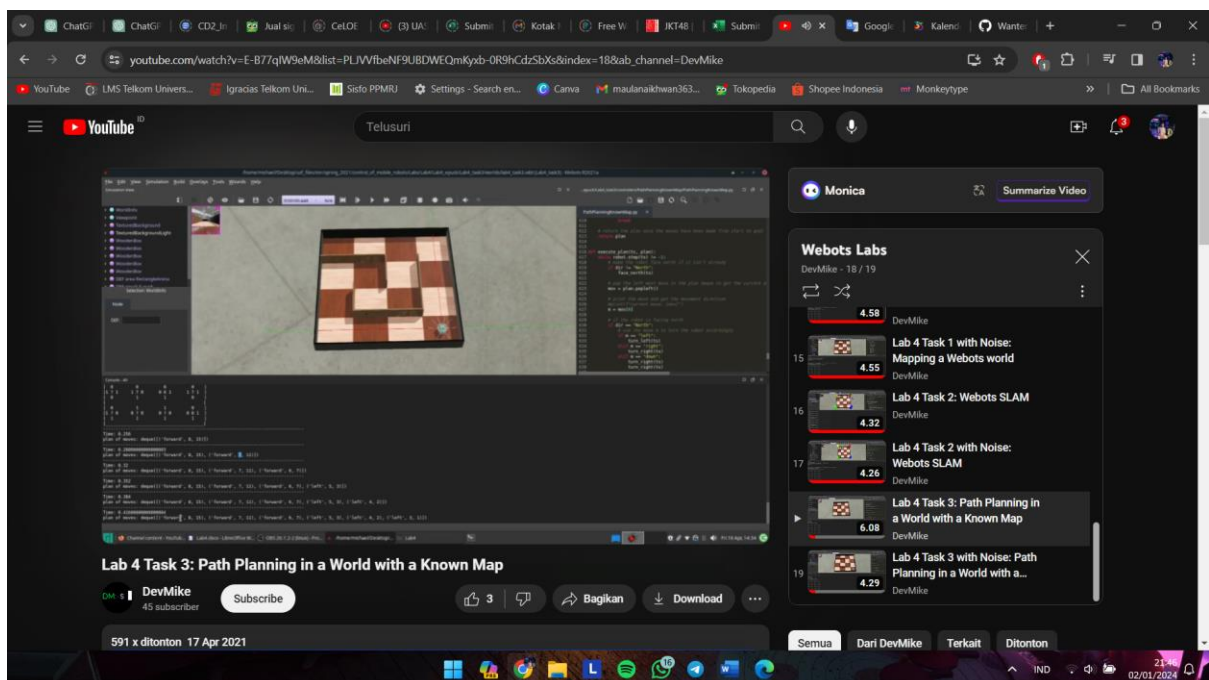
- Lab 4 Task 2 with Noise: Webots SLAM

Dalam video ini Anda dapat mendemonstrasikan SLAM di dunia Webots yang memiliki noise di dalamnya. Anda dapat melihat bagaimana robot dapat memperkirakan dirinya di suatu dunia menggunakan landmark.



- Lab 4 Task 3: Path Planning in a World with a Known Map

Video ini menunjukkan algoritma perencanaan jalur muka gelombang sedang beraksi. Hal pertama yang saya tunjukkan adalah bagaimana algoritma muka gelombang digunakan untuk menyebarkan informasi berguna dalam representasi peta dunia. Hal selanjutnya yang harus dilakukan adalah membuat rencana tindakan yang harus dilakukan mulai dari posisi awal hingga posisi tujuan. Terakhir, satu-satunya hal yang perlu dilakukan adalah melaksanakan rencana kita dengan menggunakan serangkaian tindakan yang sesuai untuk menggerakkan robot dari pose awal ke pose tujuan.



- Lab 4 Task 3 with Noise: Path Planning in a World with a Known Map

Saya mendemonstrasikan lebih lanjut algoritma perencanaan jalur muka gelombang dan membuat robot bergerak dari sel 16 menuju sel 6 dan menghadap ke Barat hingga ujung.

