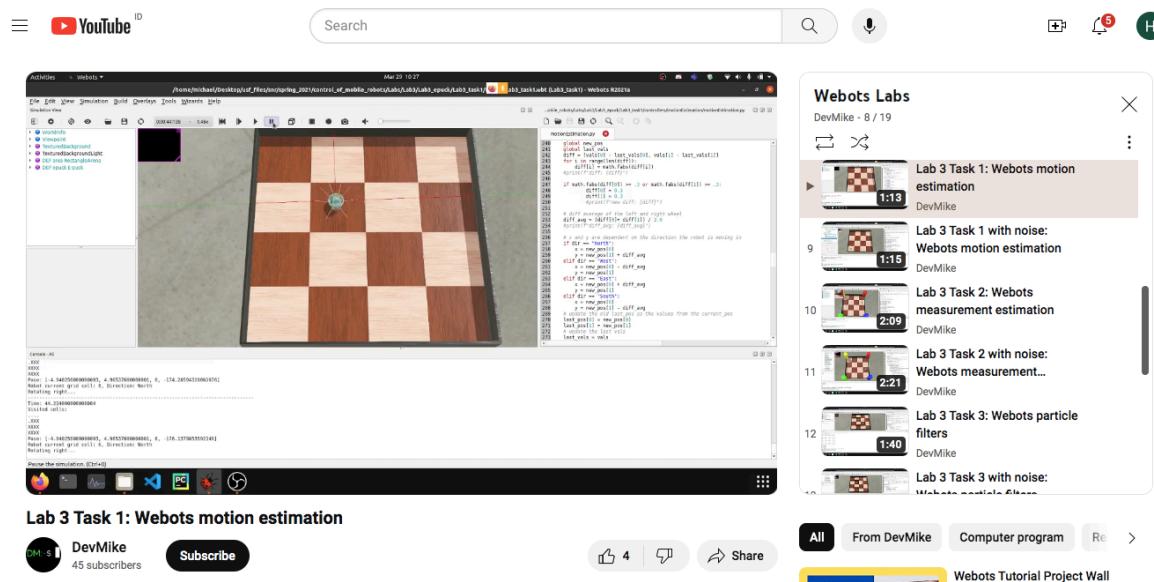


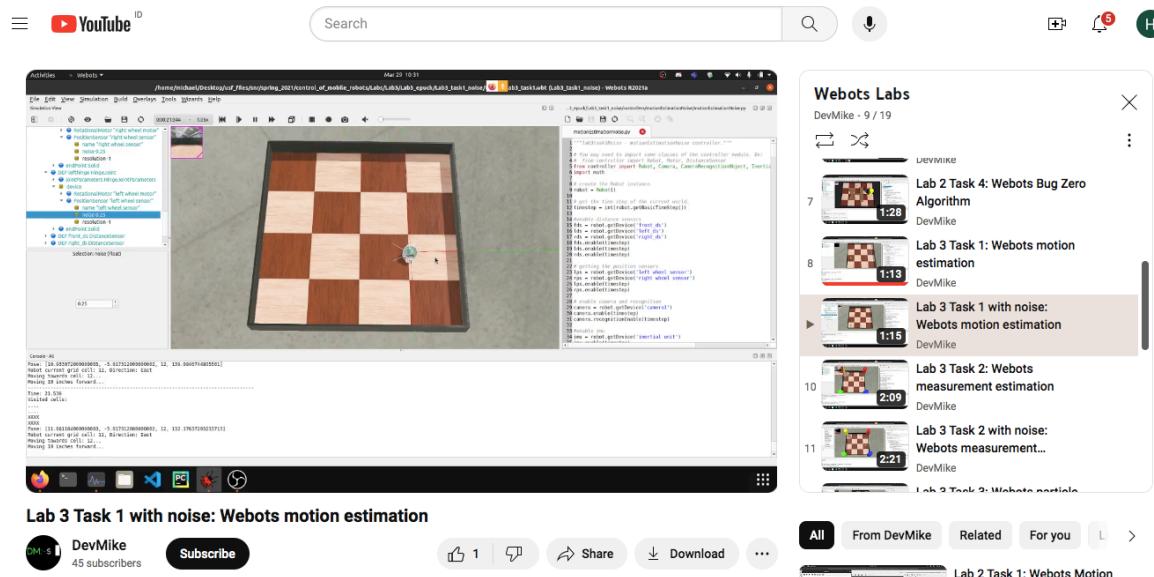
Nama : Hurin Salimah
NIM : 1103200021

Video 8



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

Video 9

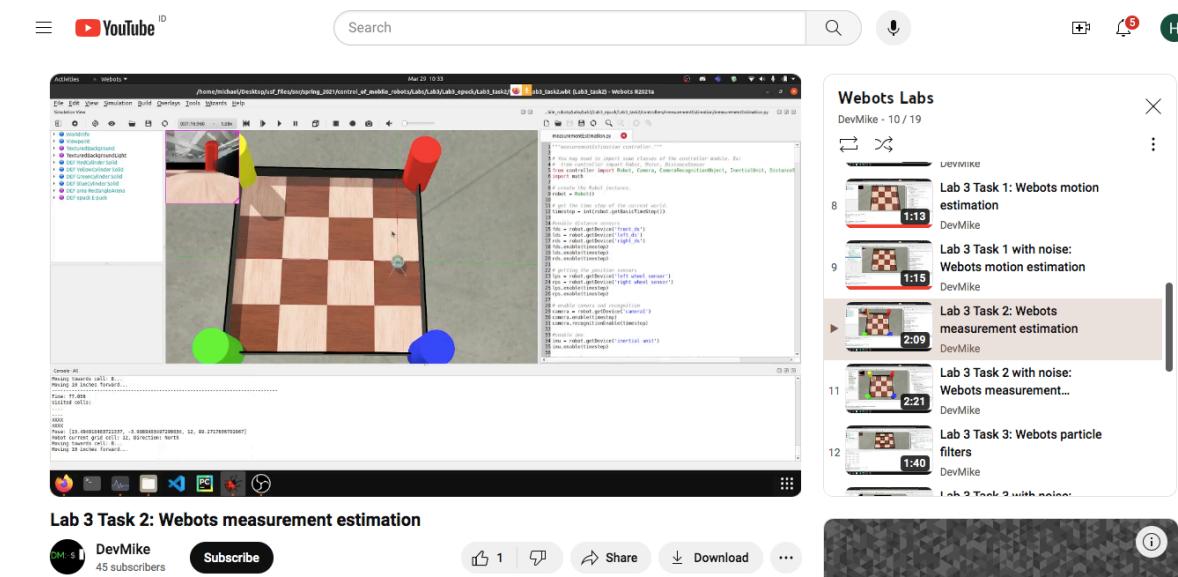


Robot ini menggunakan sensor posisi untuk mencari perbedaan dari pembacaan saat ini dan pembacaan sebelumnya untuk menentukan jarak linear yang telah ditempuhnya dari titik

awalnya di sel grid 16 yang dimulai dari koordinat 15.0, -15.0. Dari sini, robot dapat melacak seberapa jauh telah bergerak dari titik awal untuk menentukan koordinatnya dan mengetahui di kotak mana ia berada, serta terus bergerak hingga mengunjungi setiap kotak.

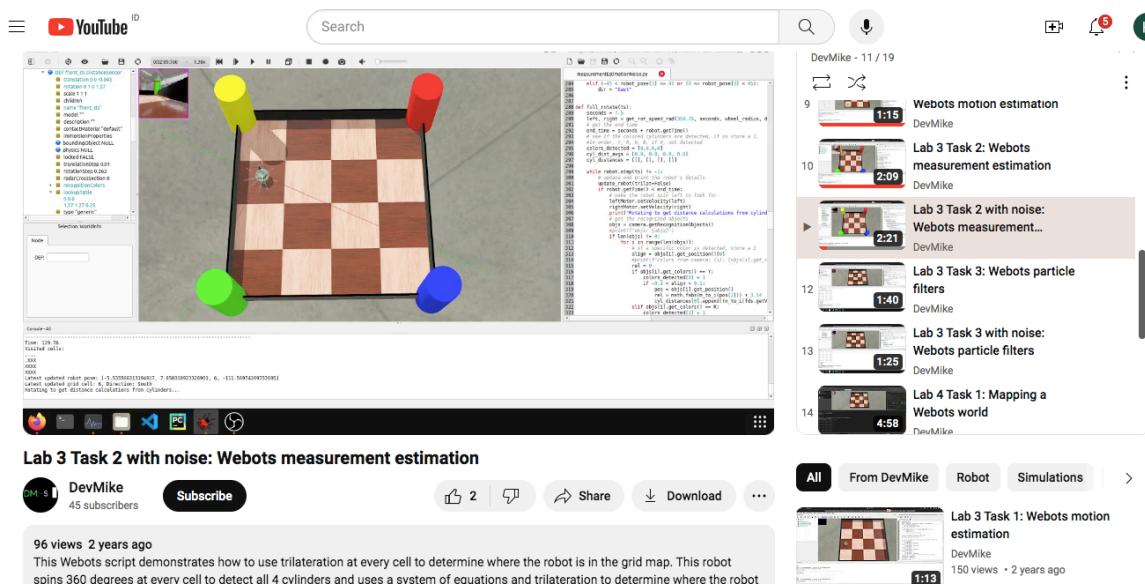
Robot ini juga mengalami noise pada unit pengukuran inersial dan sensor jaraknya, karena ia terus melacak bagaimana menambahkan perbedaan nilai ke koordinat x dan y-nya berdasarkan arah yang dituju, perlu adanya solusi untuk IMU untuk menentukan apakah sedang bergerak ke Utara, Timur, Selatan, atau Barat. Saya menyelesaikan masalah ini dengan membuat kamus Python dirs = {"Utara": 0, "Barat": 0, "Timur": 0, "Selatan": 0}, untuk setiap langkah pergerakan saya akan menambahkan arah sesuai dengan nilai IMU, saya kemudian mengasumsikan bahwa robot sedang bergerak ke arah yang memiliki hitungan tertinggi, melakukan hal ini menyelesaikan masalah saya dalam menentukan arah pergerakan dari satu sel ke sel lainnya dan dengan demikian memecahkan masalah koordinatnya juga! Mereset kamus untuk setiap arah baru yang ingin diambil juga sangat penting untuk mencegah estimasi arah yang miring.

Video 10



Webots ini menunjukkan cara menggunakan trilaterasi di setiap sel untuk menentukan posisi robot dalam peta grid. Robot ini berputar 360 derajat di setiap sel untuk mendekripsi keempat silinder dan menggunakan sistem persamaan serta trilaterasi untuk menentukan posisi robot dalam peta grid.

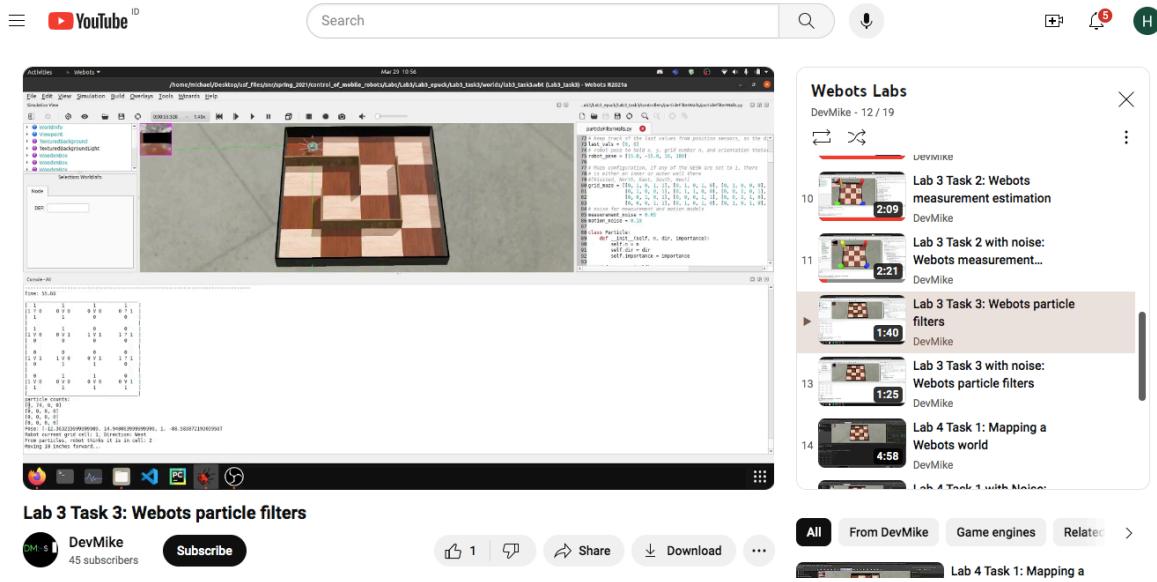
Video 11



Webots ini menunjukkan bagaimana menggunakan trilaterasi di setiap sel untuk menentukan di mana posisi robot berada dalam peta grid. Robot ini berputar 360 derajat di setiap sel untuk mendeteksi keempat silinder dan menggunakan sistem persamaan dan trilaterasi untuk menentukan posisi robot dalam peta grid.

Dunia Webots ini memperkenalkan noise atau kebisingan pada kedua sensor jarak yang membuatnya lebih sulit untuk mendekati seberapa jauh robot dari setiap silinder. Untuk mengatasinya, saya menggunakan jarak relatif kamera, dirata-ratakan dengan nilai sensor jarak depan, dan kemudian melakukan perataan pada banyak nilai tersebut untuk setiap silinder guna mendapatkan jari-jari dari robot ke setiap silinder. 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 agar mendapatkan pembacaan yang akurat.

Video 12



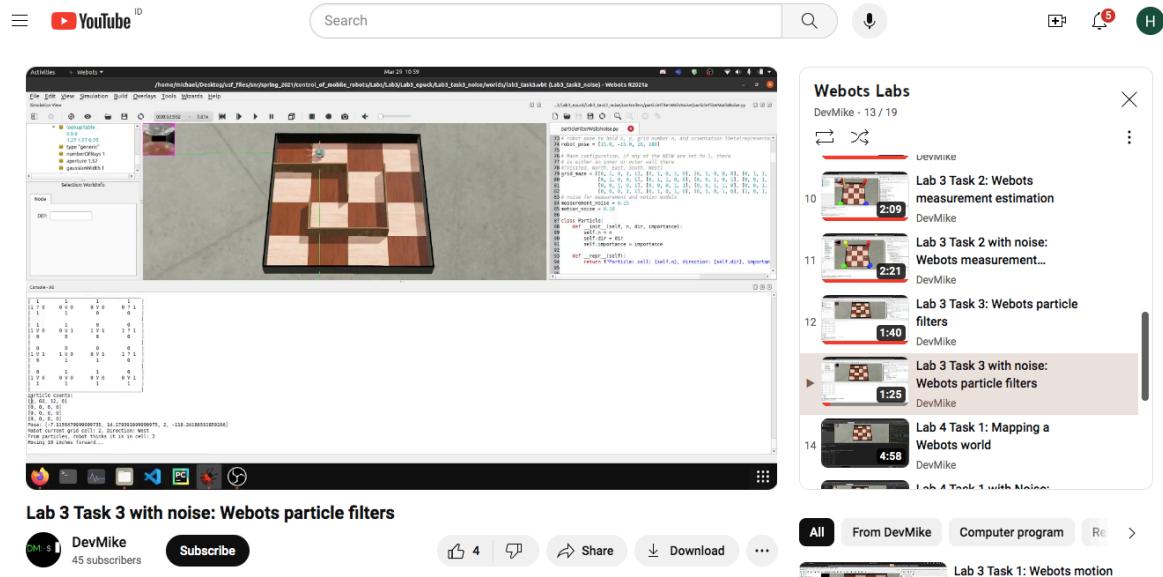
Video ini akan menginstruksikan robot untuk melintasi labirin menggunakan algoritma yang saya buat untuk menyelesaikan labirin ini dan labirin-labirin di masa depan. Ini juga menerapkan filter partikel yang terdiri dari 80 partikel. Robot akan mendapatkan dinding-dinding di sekitarnya, dan melihat kemungkinan langkah, ke kiri, ke depan, atau ke kanan. Kemudian, akan dilakukan estimasi pengukuran pada semua partikel menggunakan dinding-dinding yang baru didapatkan.

Selanjutnya, fungsi ini akan mendapatkan bobot setiap sel, menormalakkannya, dan kemudian memilih kembali partikel berdasarkan bobot yang telah dinormalisasi ini. Robot kemudian akan bergerak dengan pertama-tama, melihat apakah sel ke kiri tersedia, jika iya maka akan bergerak ke kiri. Jika tidak ada belokan ke kiri, maka akan bergerak ke depan jika itu memungkinkan. Jika bergerak ke depan tidak memungkinkan, akan dicek apakah bergerak ke kanan memungkinkan, jika iya maka akan berbelok ke kanan dan menuju ke sel tersebut. Jika tidak ada langkah ke kiri, ke depan, atau ke kanan yang tersedia, robot akan berputar menghadap ke selatan dan melihat apakah bergerak ke Selatan memungkinkan atau tidak, jika iya, maka akan bergerak ke Selatan. Jika tidak, robot akan berbelok ke kiri, sekarang menghadap ke Timur, dan bergerak ke selatan.

Dari titik ini, nilai kontrol disimpan dan digunakan sebagai parameter untuk fungsi pembaruan gerakan. Setelah setiap pergerakan, robot akan kembali menghadap ke Utara dan mengulang prosesnya sampai setiap sel dikunjungi.

video ini menggunakan model pengukuran dengan 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 dengan robot. Video ini juga menggunakan model gerakan di mana partikel bergerak ke depan 90% dari waktu dan tetap 10% dari waktu ketika pergerakan partikel memungkinkan.

Video 13

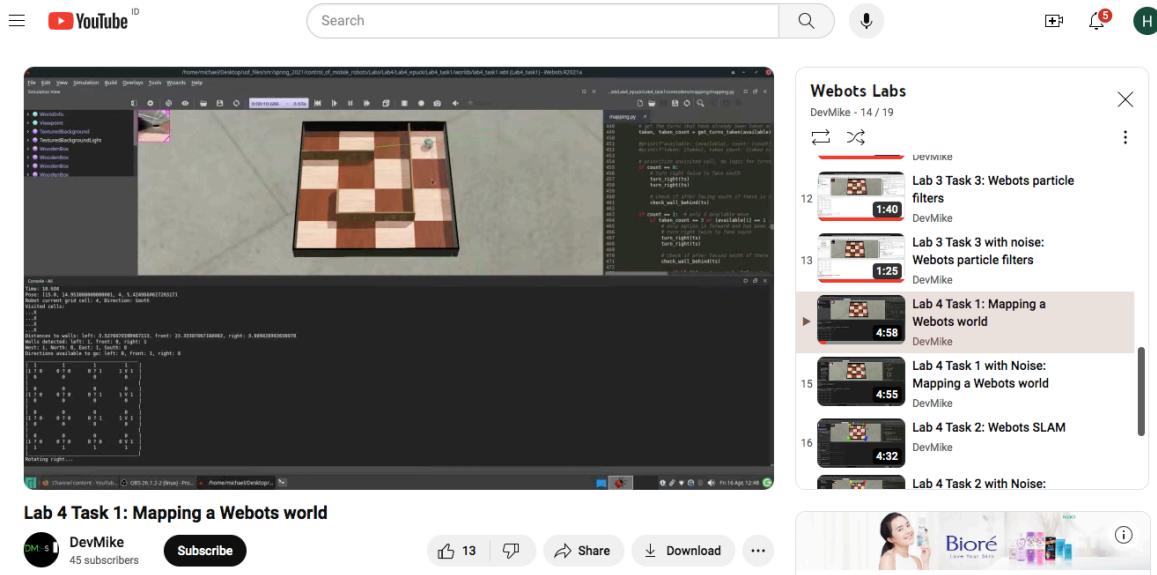


Video ini akan menginstruksikan robot untuk menjelajahi labirin menggunakan algoritma yang saya buat untuk menyelesaikan labirin ini dan labirin di masa depan. Ini juga menerapkan filter partikel yang terdiri dari 80 partikel. Robot akan mendapatkan dinding di sekitarnya dan melihat kemungkinan gerakan, ke kiri, ke depan, atau ke kanan. Kemudian, akan dilakukan estimasi pengukuran pada semua partikel menggunakan dinding yang baru saja didapatkan.

Selanjutnya, fungsi ini akan mendapatkan bobot setiap sel, menormalkannya, dan kemudian melakukan pengambilan sampel ulang partikel berdasarkan bobot yang dinormalisasi tersebut. Robot akan bergerak dengan melihat apakah sel di sebelah kiri tersedia; jika ya, maka akan bergerak ke kiri. Jika tidak ada jalan ke kiri, robot akan maju jika memungkinkan. Jika tidak memungkinkan untuk maju, robot akan memeriksa apakah bisa berbelok ke kanan; jika bisa, maka akan berbelok ke kanan dan menuju ke sel tersebut. Jika tidak ada kemungkinan gerakan ke kiri, depan, atau kanan, robot akan berputar menghadap ke selatan dan melihat apakah ada kemungkinan untuk bergerak ke arah selatan; jika ada, maka akan bergerak ke selatan. Jika tidak, robot akan berbelok ke kiri, sekarang menghadap ke timur, dan bergerak ke selatan.

Mulai dari titik ini, nilai kontrol disimpan dan digunakan sebagai parameter untuk fungsi pembaruan gerakan. Setelah setiap gerakan, robot akan menghadap ke utara lagi dan mengulangi proses tersebut hingga setiap sel dikunjungi. Video ini memiliki model pengukuran dengan tingkat kebisingan sebesar 25% pada sensor yang diterapkan saat estimasi pengukuran diaplikasikan pada setiap partikel di setiap sel ketika menentukan kemungkinan partikel menunjukkan keadaan yang sama dengan robot. Video ini juga menggunakan model gerakan dimana partikel bergerak ke depan 90% dari waktu dan tetap 10% dari waktu ketika pergerakan partikel memungkinkan.

Video 14

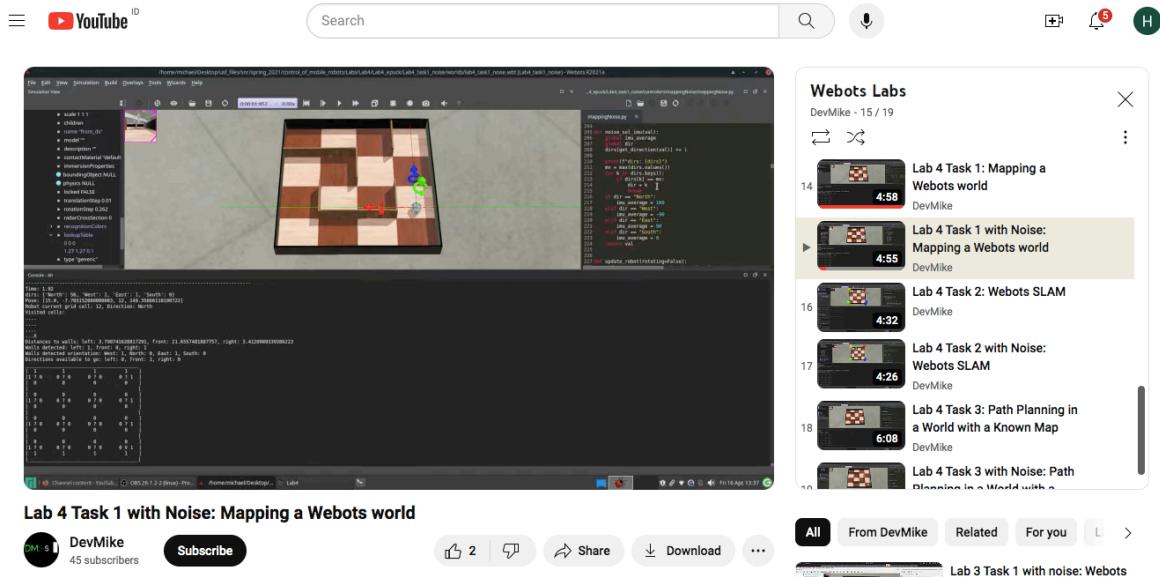


Anda memiliki pemetaan 2D dari suatu area dengan sel 10×10 yang direpresentasikan sebagai grid. Robot diwakili oleh 'x' dan sudah mengunjungi beberapa sel. Sel yang belum dikunjungi ditandai dengan 'a', 'tanda tanya', atau 'titik'. Pemetaan ini memiliki dinding eksternal di bagian barat laut, timur, dan selatan setiap sel. Dinding-dinding ini sudah ditandai, tetapi dinding bagian dalam belum.

Tujuan robot adalah melewati setiap sel yang belum dikunjungi, menandainya, dan memetakan dinding internal setelah mengunjungi setiap sel. Robot menggunakan logika untuk menentukan arahnya berdasarkan dinding yang terdeteksi di sekitarnya. Misalnya, jika tidak ada dinding di depannya, robot akan maju. Jika ada dinding di depannya, ia akan mencoba berbelok ke arah lain yang tersedia. Robot berjalan melalui grid ini dengan memperhatikan dinding-dinding internal. Ketika tidak ada jalan yang bisa ditempuh, ia berbalik dan mencari arah lain yang tersedia.

Dengan mengikuti logika ini, robot akhirnya berhasil melintasi grid, menandai sel yang telah dikunjungi, dan memetakan dinding-dinding internal hingga semua sel terpetakan dengan baik. Inti dari solusi ini adalah bahwa robot menggunakan informasi sekitarnya, mengutamakan penjelajahan sel yang belum dikunjungi, dan mencatat dinding-dinding yang terdeteksi untuk membuat pemetaan akurat.

Video 15

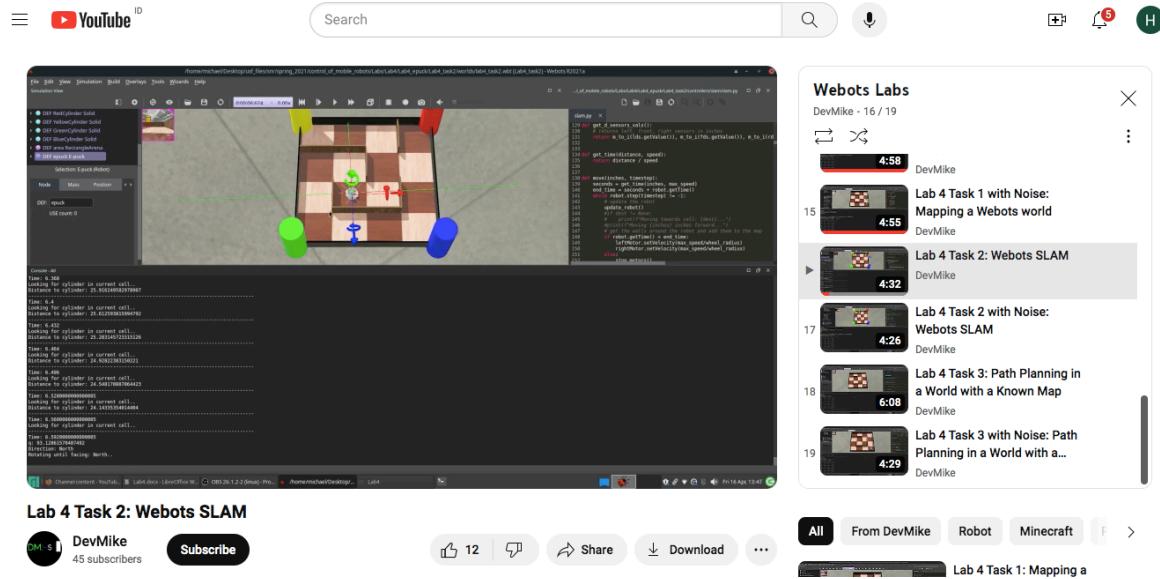


Video memiliki solusi untuk pemetaan dalam situasi di mana ada kebisingan di dalam pusaran web. Solusi ini menggunakan data dari sensor inersia (IMU) yang mengukur arah pergerakan robot. Solusi mengatasi kebisingan dengan mencatat nilai arah yang diukur oleh IMU, kemudian memasukkannya ke dalam kamus dengan kunci berdasarkan arah (barat, utara, timur, selatan). Nilai-nilai ini diakumulasikan sehingga robot bisa mengetahui arah terdominan berdasarkan pengukuran tersebut.

Dalam skenario ini, solusi serupa dengan pemetaan dinding pada dunia sebelumnya, namun kali ini mempertimbangkan kebisingan yang terjadi. Penghitungan arah yang terakumulasi dari IMU menentukan arah yang dianggap utama, meskipun terkadang terdapat ketidaksesuaian antara arah yang sebenarnya dengan yang diukur karena kebisingan.

Pada akhirnya, robot berhasil menyelesaikan pemetaan, menghindari sel-sel yang sudah dikunjungi sebelumnya, dan membuat pemetaan dinding yang cukup akurat berdasarkan nilai-nilai arah yang diperoleh dari IMU. Solusi ini berhasil mengatasi kebisingan di lingkungan pusaran web dan tetap memberikan pembacaan yang cukup akurat untuk membuat pemetaan dinding yang akurat pula.

Video 16

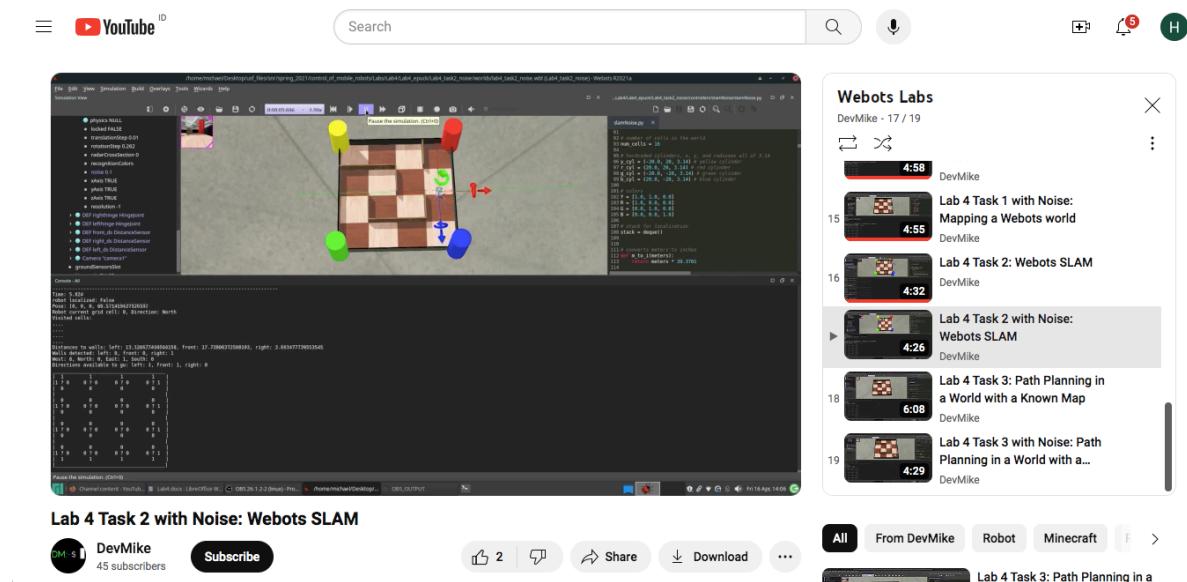


Solusimu ini menghadapi masalah pemetaan dan lokalisasi dalam sebuah lingkungan yang tidak diketahui, memanfaatkan konsep SLAM (Simultaneous Localization and Mapping) di mana robot memetakan sekitarnya sambil memperkirakan posisi dan orientasi. Robot, tanpa pengetahuan awal tentang posisi atau orientasi sebenarnya, memulai dengan berputar dalam lingkaran untuk menemukan silinder yang terletak di salah satu sudut. Dengan mengidentifikasi keberadaan silinder, robot bisa mengonfirmasi lokasinya dan memulai proses pemetaan.

Meskipun pada awalnya robot tidak dapat menemukan silinder di salah satu sudut, ia tetap melanjutkan perjalanannya untuk memetakan seluruh lingkungan. Robot secara otomatis melakukan eksplorasi dan pemetaan, tanpa pengetahuan awal tentang lokasi awalnya. Setelah menemukan silinder, robot bisa menyesuaikan langkah-langkahnya untuk menemukan dengan lebih cepat di masa depan. Dalam contoh ini, robot berputar cepat dalam lingkaran untuk menemukan silinder dan kemudian melanjutkan pemetaannya.

Dalam hasil akhirnya, robot berhasil melakukan lokalisasi dan pemetaan dengan baik, mengunjungi setiap sel dalam lingkungan dan memetakan dinding dengan akurat. Hal ini menunjukkan efektivitas dari pendekatan SLAM dalam situasi di mana posisi awal tidak diketahui.

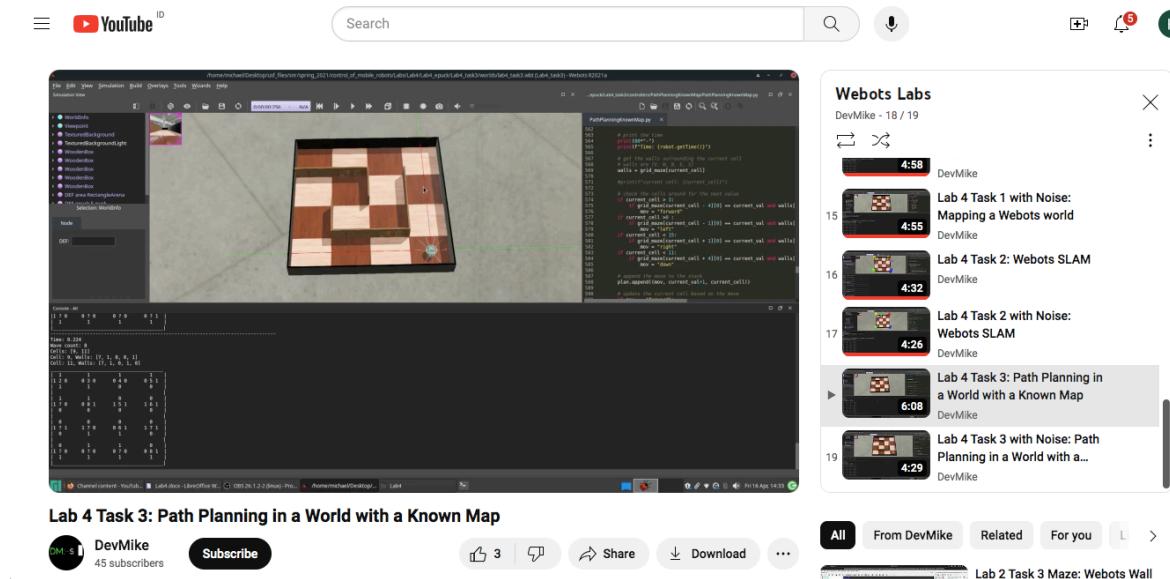
Video 17



Dalam solusi ini, robot menghadapi tantangan melokalisasi dan pemetaan di dalam dunia yang memiliki kebisingan. Dalam skenario ini, robot tidak mengetahui posisinya di awal, namun menggunakan landmark yang berbeda di sekitarnya sebagai referensi. Robot menggunakan informasi dari silinder-silinder unik di sekitarnya, diidentifikasi oleh warna dan koordinatnya. Dengan kamera di bagian depannya, robot memperkirakan posisi relatifnya terhadap setiap silinder dengan memanfaatkan nilai RGB dari landmark tersebut. Saat melewatinya, robot mendapatkan informasi posisi relatifnya.

Robot kemudian menggunakan informasi tersebut untuk melokalisasi dirinya, memulai dengan asumsi bahwa setiap titik tengah sel adalah titik nol untuk memetakan seluruh dunia. Dengan bergerak ke sekitarnya, robot berhasil memetakan setiap dinding yang dikunjunginya. Pada akhirnya, robot berhasil memperkirakan posisi dompet terakhir, yang berada sekitar 4 inci dari sumbu X dan 16 inci dari sumbu Y. Solusi kebisingan yang diusulkan tampak berhasil dalam melokalisasi robot dan memetakan seluruh lingkungan, memungkinkannya untuk bergerak dengan akurat meskipun dalam lingkungan dengan kebisingan.

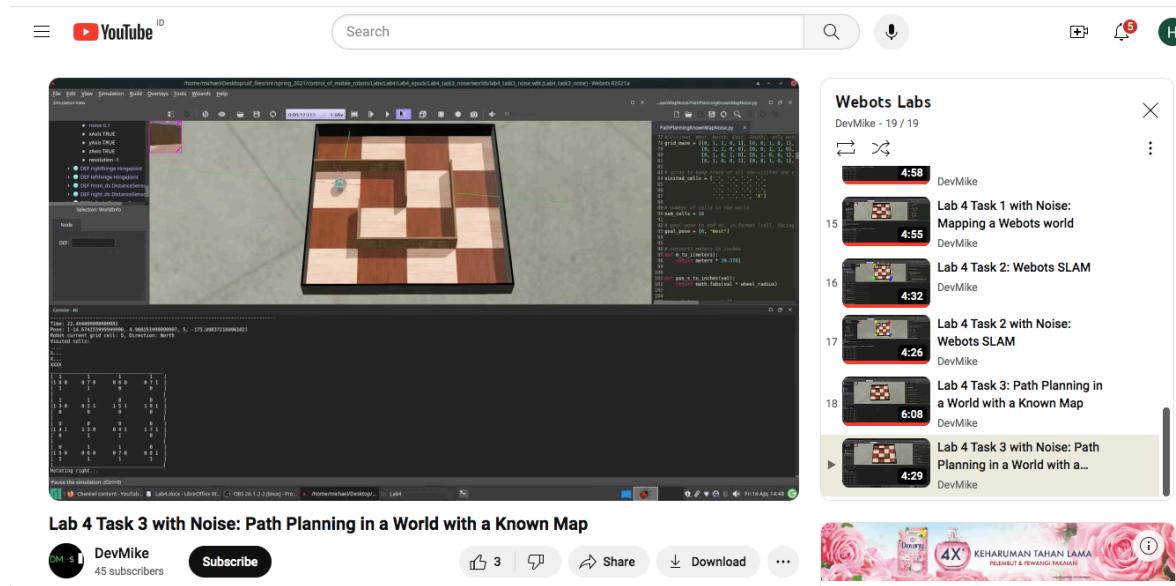
Video 18



Langkah pertama adalah menetapkan angka-nilai pada sel-sel berdasarkan kemungkinan pergerakan yang dapat dilakukan dari setiap sel. Melalui pemetaan nilai-nilai dari pose tujuan ke pose awal, algoritma memetakan nilai-nilai arah dari nilai tinggi (pose akhir) ke nilai rendah (pose awal). Setelah memetakan arah-nilai, algoritma perencanaan jalur dilakukan dari sel awal. Dengan melihat nilai-nilai sekitar, algoritma memutuskan pergerakan berikutnya hingga mencapai pose tujuan. Dalam kasus ini, perencanaan jalur memerlukan serangkaian gerakan maju, belok kiri, dan gerakan maju lagi, tergantung pada nilai-nilai yang telah dipetakan sebelumnya.

Setelah semua gerakan direncanakan, langkah terakhir adalah menjalankan rencana tersebut dengan melakukan gerakan berdasarkan perintah yang disimpan dalam dek. Robot mengeksekusi gerakan sesuai rencana, mengunjungi sel-sel yang memiliki tanda "x" karena sudah dikunjungi, hingga mencapai sel tujuan dengan arah yang diinginkan, lalu mencetak tujuan dan jalur perjalanan. Dengan demikian, demonstrasi ini menunjukkan bagaimana algoritma perencanaan jalur gelombang berhasil mengerakkan robot dari pose awal menuju pose tujuan dalam waktu yang ditentukan dengan arah yang diinginkan.

Video 19



Anda telah menunjukkan bagaimana robot dalam lingkungan simulasi Webots merencanakan jalur untuk mencapai tujuan di dunia yang memiliki kebisingan di dalamnya. Melalui peta yang diketahui, Anda menetapkan tujuan di sel tertentu dengan arah tertentu. Dalam demo ini, Anda menunjukkan langkah-langkah perencanaan jalur yang diperlukan untuk mencapai sel tujuan di tengah kebisingan tersebut. Dengan memetakan nilai-nilai pada sel-sel berdasarkan kemungkinan pergerakan, Anda membuat jalur dari sel awal ke sel tujuan dengan menggunakan angka-angka yang menunjukkan urutan pergerakan terpendek.

Anda menetapkan nilai-nilai pada sel-sel berdasarkan jalur terpendek yang tersedia. Pada kasus ini, terdapat jalur yang jelas dari sel awal ke sel tujuan yang lebih pendek daripada jalur lainnya. Setelah memetakan jalur, Anda mengeksekusi pergerakan sesuai dengan rencana yang telah disusun. Robot mengikuti jalur yang telah direncanakan, mengunjungi sel-sel yang ditandai dan akhirnya mencapai tujuan dengan arah yang diinginkan. Anda menunjukkan bahwa robot berhasil menavigasi melalui kebisingan di dunia simulasi, mengikuti jalur yang telah direncanakan, dan akhirnya mencapai tujuan dengan arah yang ditentukan. Demonstrasi ini berhasil menunjukkan perencanaan jalur yang efektif dalam menghadapi lingkungan yang kompleks.