| **TEMAT:** Programowanie robota – śledzenie linii |
| --- |

| **SCENARIUSZ LEKCJI** | | | |
| --- | --- | --- | --- |
| ***Szkoła:*** | | ***Czas (minuty)***: | 90 |
| ***Nauczyciel:*** |  | ***Wiek uczniów:*** | 13-14 |

| ***Zagadnienie główne:*** | Jak stworzyć robota do śledzenia linii |
| --- | --- |

| ***Tematyka:*** |
| --- |
| * Programowanie robota do śledzenia linii za pomocą kamery światła widzialnego, kamery termowizyjnej na podczerwień i innych instrumentów wykrywających |
| ***Cele:*** |
| * Nauka programowania robota do śledzenia linii |
| ***Oczekiwane efekty:*** |
| * Umiejętność kodowania robota do śledzenia linii |
| ***Formy pracy:***   * praca w parach, praca w grupach   ***Metody:*** |
| * prezentacja, dyskusja, ćwiczenia interaktywne |

| **TOK LEKCJI** |
| --- |
| **Przebieg zajęć** |
| **WPROWADZENIE**  Nauczyliśmy się na poprzedniej lekcji programować roboty, aby mogły śledzić i podążać za obiektami.  Upewnij się, że Twoi uczniowie zrozumieli wszystko z poprzedniej lekcji i są gotowi na kolejny krok.  Teraz nauczmy się śledzić i podążać za linią. |
|  |
| **MAIN PART**  **Śledzenie linii** odnosi się do procesu poruszania się obiektu wzdłuż określonej trasy. W pełni funkcjonalny robot do śledzenia linii wykorzystuje mobilnego robota jako nośnik, kamerę widzialnego światła, termowizor podczerwieni i inne instrumenty detekcyjne jako system obciążenia, wielopolową fuzję informacji z dziedziny widzenia maszynowego, pola elektromagnetycznego, GPS i GIS jako system nawigacji do autonomicznego poruszania się i śledzenia robota, oraz wbudowany komputer jako platformę do rozwoju oprogramowania i sprzętu dla systemu sterowania.  **Czujniki śledzenia linii**   | Porównanie | Sensor Śledzenia Linii Podczerwieni | Sensor Wizualny | | --- | --- | --- | | Koszt | Niski | Wysoki | | Zakres Widzenia | Sensor ma niewielki zakres widzenia; musi być blisko ziemi. | Zakres widzenia jest szeroki, a stan ruchu może być dostosowany z wyprzedzeniem zgodnie z zmianami linii. | | Dopasowanie do Środowiska | W przypadku zmiany środowiska użytkowania konieczne jest dostosowanie czułości sensora, a proces dostosowania jest skomplikowany. | W przypadku zmiany środowiska użytkowania konieczne jest jedynie ponowne nauczenie linii, a operacja jest prosta. | | Dopasowanie do Mapy | Zazwyczaj nadaje się tylko do prostych map z wyraźnymi liniami tła, czarno-białymi liniami lub liniami stałymi. | Nadaje się do map z wyraźnymi liniami tła, wielokolorowymi liniami, liniami stałymi, liniami przerywanymi i innymi złożonymi warunkami. |   **Algorytm śledzenia linii HuskyLens**  Funkcja śledzenia linii w HuskyLens opiera się na Pixy, projekcie open-source Uniwersytetu Carnegie Mellon. Algorytm Pixy jest w stanie rozpoznawać kolory na zdjęciach. Jego podstawową ideą jest wykorzystanie przestrzeni kolorów do usunięcia tła, które nie jest interesujące dla użytkowników, oraz wyodrębnienie pierwszego planu (takiego jak linie).  Chart, bar chart  Description automatically generated  Jak robot może śledzić czarną linię na mapie śledzenia (która ma czarne linie na białym tle)? W rzeczywistości, musimy jedynie znać względną pozycję ARTIEbota względem czarnej linii. W zasadzie mamy trzy następujące sytuacje:  1. Gdy robot jest po prawej stronie czarnej linii, powinien skręcić w lewo.  2. Gdy robot jest na środku, wyrównany z czarną linią, powinien jechać prosto.  3. Gdy robot jest po lewej stronie czarnej linii, powinien skręcić w prawo.  Graphical user interface  Description automatically generated with low confidence  **Implementacja**  Rozdzielczość ekranu HuskyLens wynosi 320×240. Punkt O w lewym górnym rogu ekranu jest początkiem układu współrzędnych ekranu (0, 0), prawo to kierunek dodatni osi X, a w dół to kierunek dodatni osi Y. Oznacza to, że współrzędne w prawym dolnym rogu ekranu to (320, 240). Przerywana pomarańczowa linia na obrazku to oś środkowa ekranu, a wartość x tej linii wynosi 160. Czarna linia na poniższym obrazku to linia mapy "widziana" przez kamerę HuskyLens. Niebieska strzałka to kierunek linii obliczony przez HuskyLens. Współrzędne punktu początkowego niebieskiej strzałki to (x1, y1), a współrzędne punktu końcowego to (x2, y2).    W uproszczeniu, potrzebujemy tylko wiedzieć, jaki jest punkt początkowy (x1) niebieskiej strzałki względem osi środkowej (x=160), aby zaimplementować śledzenie linii.  Ta funkcja może śledzić linie określonych kolorów i przewidywać ich ścieżkę. Domyślne ustawienie to śledzenie linii jednego koloru, a w tym projekcie będzie używane **śledzenie linii jednego koloru**.  **Ustawienia kamery**  **Krok 1:** Przekręć przycisk funkcji w lewo lub w prawo, aż na górze ekranu pojawi się napis "Śledzenie linii".  **Krok 2:** Przytrzymaj przycisk funkcji, aby wejść w ustawienia parametrów funkcji śledzenia linii.  **Krok 3:** Przekręć przycisk funkcji w prawo lub w lewo, aż zostanie wybrana opcja "Ucz się wielokrotnie". Następnie krótko naciśnij przycisk funkcji, a następnie przekręć go w lewo, aby wyłączyć przełącznik "Ucz się wielokrotnie", czyli kwadratowy symbol na pasku postępu zostanie przekręcony w lewo. Następnie krótko naciśnij przycisk funkcji, aby zatwierdzić ten parametr.    **Krok 4:** Możesz również włączyć diodę LED, ustawiając przełącznik "LED". Jest to bardzo przydatne w ciemnym otoczeniu. Korzystając z powyższej metody, włącz przełącznik "LED".  **Krok 5:** Przekręć przycisk funkcji w lewo, aż zostanie wybrana opcja "Zapisz i powróć", a następnie krótko naciśnij przycisk funkcji, aby zapisać parametry. Automatycznie zostanie wykonany powrót.  **Nauka i śledzenie**  Nauka linii: Wskaż symbol "+" na linii, a następnie skieruj pomarańczowe pole na tło. Zaleca się, aby na ekranie nie było innych linii. Staraj się utrzymać HuskyLens równolegle do linii docelowej; HuskyLens automatycznie wykryje linię, a na ekranie pojawi się biała strzałka. Następnie krótko naciśnij przycisk "nauki", a biała strzałka zmieni się w niebieską strzałkę.     * Podczas uczenia linii musimy dostosować pozycję HuskyLens, aby była równoległa do linii. * HuskyLens może uczyć się linii o dowolnym kolorze, który ma wyraźny kontrast kolorystyczny do tła, ale linia ta musi być jednokolorowa, aby proces śledzenia linii był stabilny. * HuskyLens może uczyć się i śledzić wiele linii o różnych kolorach, ale wszystkie te linie muszą być jednokolorowe i w widocznym kontraście do tła. W tym przykładzie użyjemy czarnej linii (czarny taśma izolacyjna na białym tle, takim jak papier lub biała płyta MDF). * Widoczność linii zależy w dużej mierze od oświetlenia otoczenia. Podczas śledzenia linii staraj się utrzymać oświetlenie otoczenia jak najstabilniejsze i użyj diody LED HuskyLens, jeśli to konieczne.   **Otwórz Mind+ i załaduj rozszerzenia**  Otwórz Mind+ i załaduj rozszerzenia. **Gdy korzystasz z robota micro:Maqueen Plus, upewnij się, że wybierasz odpowiednią wersję (V1 lub V2).**  Zmień nazwę **mojej zmiennej typu float na x**. Kliknij prawym przyciskiem myszy na zmienną -> Zmień nazwę zmiennej numerycznej.  **Algorytm:**  **•** Odczytaj wartość x1 (lub x początkowe) z funkcji śledzenia linii HuskyLens - jest to początkowy punkt niebieskiej strzałki.  • Jeśli czarna linia znajduje się po lewej stronie ekranu (x1<150), robot powinien skręcić w lewo.  • Jeśli czarna linia znajduje się po prawej stronie ekranu (x1>170), robot powinien skręcić w prawo.  • Jeśli czarna linia znajduje się w środku ekranu (150<=x1<=170), robot powinien jechać prosto.  **Opcja 1 - Śledzenie linii z Maqueen Plus**  Użyj tego kodu:  **Opcja 2 - Śledzenie linii z użyciem Arduino (ArtieBot)**  Najpierw zdefiniuj bloki Drive i Stop, jak opisano w lekcji Programowanie robota. Dobrym pomysłem jest zebranie często używanych skryptów i łatwe przenoszenie ich między projektami.      **Obie opcje — przetestuj swój algorytm**  Przygotuj białą powierzchnię (papier lub MDF) i za pomocą taśmy izolacyjnej stwórz linię  Prześlij program na swojego robota, umieść go gdzieś na linii i zobacz, jak się porusza. Czy śledzi linię? Czy gubi linię? Czy jest jakiś sposób na radzenie sobie z tym?  Oto kilka wskazówek, które mogą Ci pomóc:  - Spróbuj zmienić kąt kamery.  - Spróbuj uzyskać **początek X** strzałki dla ID 1 zamiast końca X.  Dodaj kod zabezpieczający, korzystając z następującej wskazówki:  - Jeśli linia jest obecna i wykryto strzałkę, przejdź do kodu obsługi strzałki i zapisz zawartość zmiennej x do nowej zmiennej o nazwie **last x**.  - Jeśli linia zostanie utracona i nie zostanie wykryta strzałka, użyj **last x** zamiast x, aby wrócić do linii.  I na koniec najważniejsze - wprowadź korekty, aby ruch był płynny, poprzez dostosowanie prędkości silników i czasu jazdy. Algorytm ten analizuje pozycję x; posiada 3 możliwości jak na zdjęciu poniżej.    Czy możesz zoptymalizować ten algorytm dla 5 przypadków lub nawet 7, aby ruch był płynniejszy?  Spójrz na obrazki poniżej, aby dowiedzieć się, jak to zrobić.  Slika na kojoj se prikazuje tekst  Opis je automatski generiran  Podsumowanie zdobytej wiedzy:  1. Zrozumienie głównych zasad śledzenia linii.  2. Umiejętność korzystania z funkcji śledzenia linii w HuskyLens.  3. Zastosowanie i optymalizacja algorytmu śledzenia linii. |
|  |
| **PODSUMOWANIE**  Jak sama nazwa wskazuje, robot śledzący linię to zautomatyzowany pojazd, który podąża za widoczną linią umieszczoną na powierzchni. Ta linia stanowi ścieżkę, po której porusza się robot śledzący linię. Zazwyczaj wykorzystuje się czarną linię na białej powierzchni, choć można ją również dostosować jako białą linię na czarnej powierzchni.  Zazwyczaj początkujący i studenci mają pierwsze doświadczenia z robotami właśnie tego typu.  W przemyśle wykorzystuje się olbrzymie roboty śledzące linie do wspomagania zautomatyzowanego procesu produkcji. Są one również wykorzystywane w celach wojskowych, pomocy ludziom, usług dostarczania, itp.  Teraz rozumiemy główne zasady śledzenia linii i umiemy stosować funkcję śledzenia linii w HuskyLens.  Wiemy także, jak stosować i optymalizować algorytm śledzenia linii.  Omów z uczniami różnice i podobieństwa między śledzeniem i podążaniem za obiektami a śledzeniem i podążaniem za linią.  Opracuj tabelę K.W.L. (Know, Want, Learned) ze swoimi uczniami.   | Co wiem | Co chcę wiedzieć | Czego się nauczyłem/nauczyłam | | --- | --- | --- | |  |  |  | |
|  |

| ***Metody*** | ***Formy pracy*** |
| --- | --- |
| ***prezentacja***  ***ćwiczenie interaktywne/symulacja komputerowa*** | ***praca w parach***  ***praca grupowa*** |

| ***Materiały:*** |
| --- |
| * <http://mindplus.cc/download-en.html> |

| ***Literatura*** |
| --- |

| **OBSERWACJE, UWAGI, NOTATKI** |
| --- |
|  |