

Witold WOCH\*, Michał FILIPIAK\*, Dariusz KURZ\*

## SYSTEM POZYCJONOWANIA POJAZDU WZGLĘDEM ŁADOWARKI BEZPRZEWODOWEJ

W artykule przedstawiono system pozycjonowania pojazdu względem ładowarki bezprzewodowej. We wstępie pracy omówiono rozwój bezprzewodowego ładowania oraz wskazano główny problem dotyczący strat energii przy tym procesie. W ramach badań poddano analizie metodę pozycjonowania pojazdu, z zamiarem zwiększenia efektywności procesu ładowania. W celu przedstawienia funkcjonalności tej metody, skonstruowano zdalnie sterowany pojazd posiadający innowacyjne koła typu Mecanum. W pracy zrealizowano pomiary prędkości pojazdu, aby zwiększyć dokładność algorytmu pozycjonowania.

SŁOWA KLUCZOWE: bezprzewodowe ładowanie, samochody elektryczne, system pozycjonowania.

### 1. WSTĘP

Na przestrzeni ostatnich lat, ładowanie indukcyjne umożliwiło ładowanie baterii urządzeń bez stosowania połączenia galwanicznego. W ten sposób zmieniło się postrzeganie sposobu przekazywania energii i poszerzył się zakres stosowania tego typu zasilania. Rozwój tej technologii i jej powszechne wdrożenie zapewniło użytkownikom wygodę oraz wyższą mobilność w codziennym życiu. Jednym z pierwszych urządzeń posiadających ładowanie indukcyjne były szczoteczki do zębów Oral-B. Następnie pojawił się na rynku multimedialny smartfon Palm Pre, który spopularyzował tę technologię w dziedzinie telefonii komórkowej. W ostatnim czasie, przemysł motoryzacyjny również nie pozostał obojętny na możliwość bezprzewodowego ładowania pojazdów elektrycznych. W kwestii tej problematyki prezentacje swoich rozwiązań przedstawili producenci wiodących marek takich jak BMW, Audi czy Volvo. W każdym z tych przypadków podstawowa idea pozostała niezmienna i dotyczyła możliwości ładowania samochodu bez użycia przewodów. Co różniło te systemy, to zarówno efektywność ładowania, jak i techniki dokładnego ustawienia pojazdu względem stacji ładowania [1].

Główną wadą układów ładowania indukcyjnego jest ich niska sprawność (<90%) w przypadku gdy odstęp między cewkami jest większy niż 0.2 metra.

---

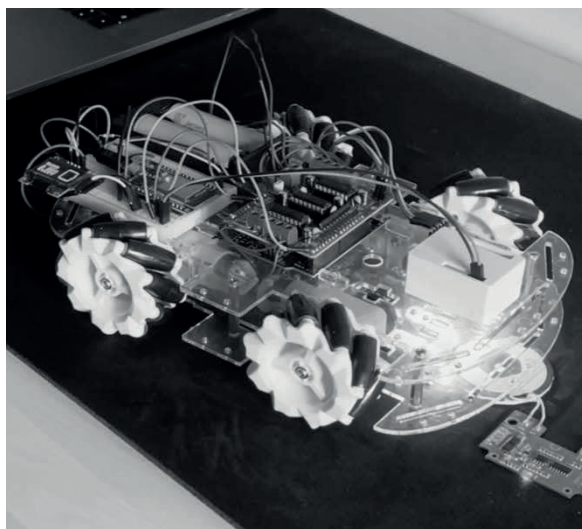
\* Politechnika Poznańska

Jednym ze sposobów na zapewnienie wysokiego sprzężenia magnetycznego jest precyzyjny system pozycjonowania pojazdu, pozwalający podłączyć samochód do ładowarki z akceptowalną tolerancją, w taki sposób, aby straty energii elektrycznej podczas ładowania były jak najmniejsze. Kolejnym wyzwaniem jest także standaryzacja tych systemów, zwłaszcza biorąc pod uwagę różnice w częstotliwości pracy sprzężonych cewek [2]. Obecnie obowiązuje norma SAE, która określa znamionową częstotliwość pracy na 85 kHz [3]. W obliczu rosnącej popularności ładowania indukcyjnego, standaryzacja jest niezbędna do zapewnienia szerokiej kompatybilności między urządzeniami różnych firm. Ostatecznie, sukces komercyjny takich systemów będzie zależał od tego, jak producenci zdołają zminimalizować te trudności dla użytkowników końcowych.

## 2. BUDOWA POJAZDU

Celem wykonanych badań jest zaprogramowanie systemu pozycjonowania pojazdu względem ładowarki indukcyjnej i ocena jego przydatności. System ten bazuje na analizie obrazu z kamery OV2640 podłączonej do modułu ESP32-CAM. Do analizowania obrazu z kamery wykorzystano bibliotekę OpenCV w środowisku programistycznym Python. Na wyświetlanym obrazie ustawiono punkt, którego współrzędne wyznaczają środek cewki odbiorczej znajdującej się na pojeździe. Gdy samochód zbliży się do cewki nadawczej w taki sposób, że cała jej powierzchnia będzie widoczna w polu widzenia kamery, odpowiednia funkcja z biblioteki OpenCV wykryje jej środek. Następnie zostanie wyznaczona odległość pomiędzy stałym punktem środka cewki odbiorczej, a wykrytym punktem środka cewki nadawczej. Na podstawie podanej w milimetrach wartości odległości, użytkownik może wjechać sam na ładowarkę korzystając z obrazu z kamery lub też załączyć automatyczne pozycjonowanie.

W celu przeprowadzenia odpowiednich testów oceniających funkcjonalność systemu skonstruowano zdalnie sterowany pojazd. Budowę jego można sklasyfikować na dwie główne części: mechaniczną i elektryczną. Część mechaniczną podzielono na: podstawę składającą się z dwóch warstw szkła akrylowego, tulejki dystansowe wraz z śrubami, koła oraz osłonę na kamerę. Natomiast część elektryczna zawiera: cewkę odbiorczą i nadawczą, wyświetlacz OLED, płytkę stykową, moduł Bluetooth, mikrokontroler Arduino UNO oraz ESP32, silniki, moduł z kamerą i dwa ogniwa litowo-jonowe służące do zasilania pojazdu. Na rysunku 1 przedstawiono pojazd wykorzystywany w testach systemu.



Rys. 1. Zbudowany pojazd służący do testów systemu

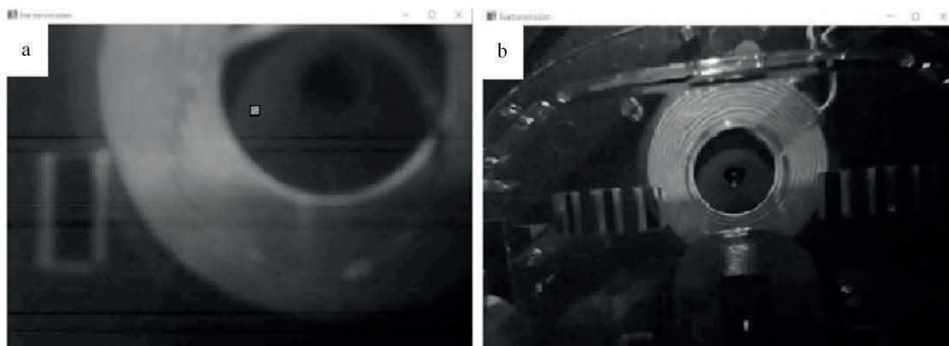
Ważnym, a zarazem innowacyjnym elementem konstrukcji pojazdu są koła typu Mecanum, które umożliwiają poruszanie się samochodu w dowolnym kierunku. Jest to możliwe przez specjalną strukturę budowy koła, które składa się z dwóch pierścieni oraz dziewięciu rolkowych elementów swobodnie obracających się pod kątem 45 stopni. Jednym z głównych powodów zastosowania kół Mecanum jest ich wysoka precyzja na małych odcinkach drogi. Do badań systemu pozycjonowania zastosowano pojazd o niewielkich rozmiarach, co wiąże się z pokonywaniem małych odległości. Na rysunku 2 przedstawiono koło Mecanum wykorzystane w projekcie.



Rys. 2. Koła Mecanum zastosowane w projekcie

Niewielkie rozmiary pojazdu używanego do przeprowadzania testów systemu, powodują niskie usadowienie kamery, dlatego w projekcie zastosowano szerokokątny obiektyw, który zapewnia szerszy kąt obserwacji. Wykorzystanie tego obiektywu umożliwi objęcie w polu widzenia kamery zarówno cewkę

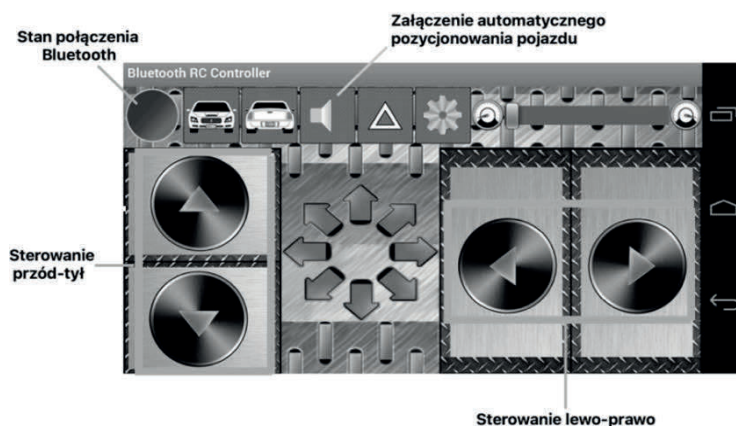
odbiorczą, jak i cewkę nadawczą, co jest warunkiem koniecznym dla prawidłowego działania algorytmu. Na rysunku 3 przedstawiono różnice pomiędzy wyświetlanym obrazem ze standardowej kamery oraz kamery wyposażonej w obiektyw szerokokątny. Dla obu wizualizacji, kamerę umieszczono w dokładnie takim samym punkcie.



Rys. 3. Porównanie wyświetlanego obrazu z obu kamer: a) obraz z kamery bez obiektywu szerokokątnego, b) obraz z kamery z obiektywem szerokokątnym

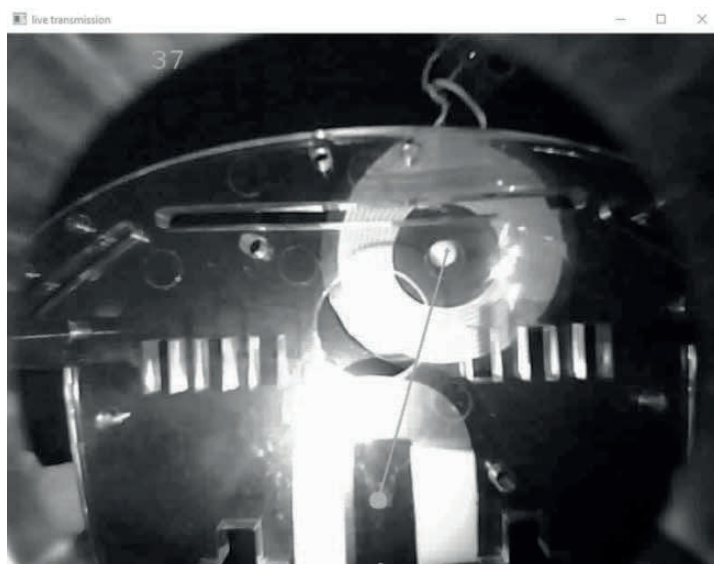
### 3. APLIKACJA I STEROWANIE POJAZDEM

Sterowanie pojazdem odbywa się poprzez aplikację działającą na smartfonie z systemem Android. Telefon łączy się z modułem Bluetooth znajdującym się na pojeździe. Każdy przycisk w aplikacji ma przypisaną odpowiednią zmienną, która jest przesyłana do Arduino UNO podłączonego do sterownika silników. Dostępne zmienne, które są wysyłane do mikrokontrolera to: F (jazda do przodu), B (jazda do tyłu), L (skręt w lewo), R (skręt w prawo), V (załączenie automatycznego pozycjonowania pojazdu). Na rysunku 4 przedstawiono wygląd aplikacji wraz z opisem wykorzystywanych przycisków.



Rys. 4. Funkcje wykorzystywane w aplikacji do sterowania pojazdem

Gdy pojazd znajduje się w pobliżu ładowarki, należy uruchomić procedurę pozycjonowania pojazdu, którą napisano w Pythonie. W pierwszej kolejności skrypt identyfikuje adres IP kamery, która jest połączona z modułem ESP32-CAM. Następnie, aplikacja tworzy okno wyświetlające obraz w zdefiniowanej rozdzielczości  $800 \times 600$  pikseli. Po aktywacji biblioteki OpenCV, na ekranie utworzony zostaje punkt o stałych współrzędnych (415, 520). Ten punkt symbolizuje środek cewki odbiorczej, która jest zamontowana na pojeździe. Program wykorzystuje filtr medianowy, aby zwiększyć jakość obrazu, poprzez redukcję szumów. Filtr ten wpływa korzystnie na efektywność rozpoznawania kształtów. Przy użyciu funkcji HoughCircles dostępnej w bibliotece OpenCV, wykrywany jest środek cewki odbiorczej. Następnie stosując zależność odległości euklidesowej, wyznaczana jest odległość między nieruchomym punktem środka cewki zamontowanej na pojeździe, a wykrytym punktem oznaczającym środek cewki ładowarki. W kolejnym etapie przeliczana jest wartość wyświetlanej odległości z pikseli na milimetry. Aby zidentyfikować konkretną wartość dzielnika wykonano szereg pomiarów. Przy wykorzystaniu linijki zmierzono rzeczywistą odległość pomiędzy dwoma punktami, która została przyrównana do wartości wyświetlanej w programie. Na tej podstawie oszacowano, że odległość, która jest podawana w pikselach, należy podzielić przez wartość 7,7, aby otrzymać rzeczywistą odległość mierzoną w milimetrach. Zastosowanie przekształcenia jednostki wyświetlanej z pikseli na milimetry, pomaga użytkownikowi lepiej zrozumieć faktyczny dystans między pojazdem, a ładowarką. Na rysunku 5 przedstawiono obraz z kamery podczas wyznaczania odległości pomiędzy cewkami.



Rys. 5. Wyznaczona odległość między cewką nadawczą a odbiorczą

Zastosowany układ kół Mecanum umożliwia poruszanie się pojazdu pod kątem 90 stopni zgodnie z osią  $x$  i  $y$ . Zastosowany algorytm automatycznego systemu pozycjonowania został tak zaimplementowany, że pozwala na ruch tymi współrzędnymi:  $x$ , charakteryzującą przesunięcie w prawo lub w lewo oraz  $y$ , która oznacza przesunięcie do przodu lub do tyłu. W danym projekcie założono, że pojazd nie wyjedzie poza obrys cewki odbiorczej, stąd przyjęto, że wartość współrzędnej  $y$  będzie zawsze dodatnia. Wyznaczone w programie parametry  $x$  i  $y$ , określające odległość między dwoma punktami, są przesyłane za pomocą protokołu TCP do modułu ESP32 znajdującego się na pojeździe. Aplikacja korzysta z modułu socket, służącego do inicjowania punktu dostępu sieciowego oraz łączenia się z mikrokontrolerem ESP32. Realizacja tego połączenia odbywa się przez określenie właściwego adresu IP oraz portu serwera. W celu uniknięcia komplikacji związanych z przesyłaniem danych przez wykorzystywany protokół TCP, wartości współrzędnych  $x$  i  $y$  są konwertowane do formatu string. Taki sposób przedstawiania danych ułatwia transmisję i umożliwia mikrokontrolerowi ESP32 łatwe rozpoznawanie oraz rekonwersję tych wartości na ich numeryczne odpowiedniki, dzieląc je według wybranego separatora (przecinek).

W całym procesie przesyłu informacji pomiędzy komputerem a pojazdem, mikrokontroler ESP32 pełni rolę pośrednika, który odbiera informacje wysyłane po sieci z komputera i dostarcza je do Arduino UNO, do którego podłączone są silniki. Moduł ESP32 komunikuje się z siecią bezprzewodową poprzez ruter mobilny, używając informacji takich jak identyfikator sieci i klucz dostępowy. Po udanym połączeniu powstaje serwer http na porcie 23. Następnie mikrokon-

troler odbiera ciągi znaków wysyłane przez komputer i przetwarza je na rzeczywiste wartości liczbowe. Przez łącze szeregowo, współrzędne  $x$  i  $y$  są transmitowane z ESP32 do Arduino UNO. Na podstawie tych danych, sterownik Motor Shield L293D aktywuje konkretne silniki. Gdy moduł Bluetooth podłączony do płytki Arduino otrzyma wiadomość wysłaną z aplikacji sterującej w postaci zmiennej „V”, oznaczającej aktywowanie automatycznego systemu pozycjonowania, program sprawdza czy na porcie szeregowym znajdują się wartości współrzędnych, które są dostarczane przez moduł ESP32. Jeśli znajdują się dane parametry, odpowiednia funkcja sprawdza najpierw współrzędną  $x$ . Jeśli jest dodatnia, oznacza to, że pojazd znajdują się po lewej stronie względem ładowarki, w związku z tym należy go przesunąć o wartość współrzędnej  $x$  w prawo. Natomiast gdy współrzędna  $x$  jest ujemna, wtedy pojazd przesuwa się o kąt 90 stopni w lewo, ponieważ znajduje się po prawej stronie względem cewki odbiorczej. Następnie sprawdzana jest wartość  $y$ , która z założenia jest zawsze dodatnia, dlatego też pojazd zawsze przesuwa się do przodu o odczytaną wartość  $y$ .

W celu precyzyjnego określenia czasu pracy silników do przejechania wskazanej odległości, wykonano serie pomiarów służących do określenia prędkości pojazdu. W związku z tym wyznaczono odcinek jednego metra, a następnie zmierzono czas potrzebny do pokonania tej drogi przez pojazd. Czas pomiaru uwzględnia także opóźnienie załączenia silników, co umożliwi precyzyjniejsze dostosowanie czasu działania silników w odniesieniu do automatycznego pozycjonowania pojazdu. W tabeli 1 przedstawiono wyniki przeprowadzonych pomiarów czasu w jakim pojazd przebył drogę jednego metra.

Tabela 1. Wyniki pomiarów czasu przejazdu pojazdu dla odległości 1 metra.

Pomiar	Czas
[-]	[s]
1	4,76
2	4,78
3	4,80
4	4,78
5	4,82
6	4,78
7	4,80
8	4,76

Na bazie uzyskanych danych z pomiarów obliczono średnią prędkość, z jaką pojazd przemierzył jednometrowy odcinek, korzystając z danej formuły (1).

$$\begin{aligned} \bar{t} &= \frac{t_1 + t_2 + \dots + t_8}{n} = \\ &= \frac{4,76 + 4,78 + 4,80 + 4,78 + 4,82 + 4,78 + 4,80 + 4,76}{8} = 4,785 \text{ s} \end{aligned} \quad (1)$$

Biorąc pod uwagę, że dane przychodzące z ESP32 przeliczane są na milimetry, prędkość pojazdu określono w milimetrach na sekundę, bazując na następującym wzorze (2).

$$v = \frac{s}{t} = \frac{1000}{4,785} = 209 \frac{mm}{s} \quad (2)$$

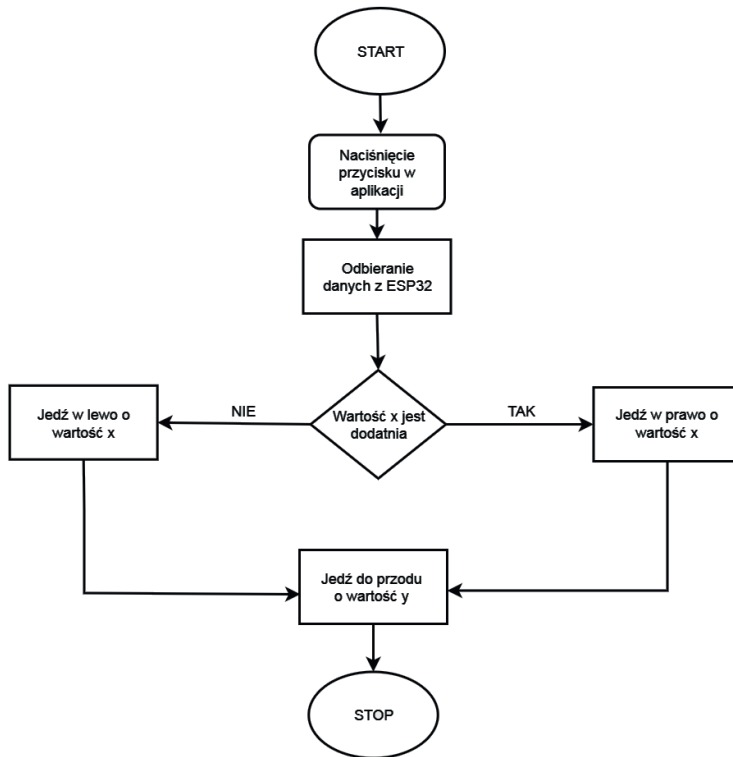
Podczas określania czasu działania silników, podzielono wartości  $x$  i  $y$  przez wcześniej wspomnianą wartość 7,7, co pozwoliło przeliczyć piksele na milimetry. Dodatkowo, dokonano konwersji czasu z sekund na milisekundy, dzieląc wartości przez ustaloną prędkość i mnożąc przez 1000. Za pomocą tych kroków w programie, mikrokontroler Arduino UNO jest w stanie dynamicznie dostosowywać czas działania silników w zależności od odbieranych wartości współrzędnych  $x$  i  $y$ . Na rysunku 6 przedstawiono kod funkcji, którą przetwarza mikrokontroler Arduino UNO.



```
void odbieranieESP() {  
  
  Serial.println("start");  
  if (Serial.available() > 0) {  
    float ESPx = Serial.parseFloat();  
    float ESPy = Serial.parseFloat();  
    Serial.print("ESPx: ");  
  
    Serial.println(ESPx);  
    delay(10);  
  
    Serial.print("ESPy: ");  
  
    Serial.println(ESPy);  
  
    float dx=(((ESPx)/7.7)/209)*1000.0;  
    float dy=(((ESPy)/7.7)/209)*1000.0;  
  
    if (ESPx > 0) {  
      Serial.print("wchodzi");  
      Serial.println(dx);  
      left();  
      delay(dx);  
    }  
  
    if (ESPx < 0) {  
      Serial.print("wchodzi");  
      Serial.println(dx);  
      right();  
      delay(dx);  
    }  
  
    if (ESPy > 0) {  
      Serial.print("wchodzi");  
      Serial.println(dy);  
      back();  
      delay(dy);  
    }  
  
  }  
}
```

Rys. 6. Fragment kodu dotyczący odbioru wartości współrzędnych x i y, a także wyznaczania czasu działania silników

Na rysunku 7 przedstawiono algorytm przetwarzania danych przez mikrokontroler Arduino UNO. Schemat blokowy pokazuje cały proces od odebrania wartości współrzędnych z ESP32 do prawidłowego ustawienia pojazdu względem ładowarki.



Rys. 7. Algorytm systemu pozycjonowania pojazdu względem ładowarki bezprzewodowej

## 4. WNIOSKI

Przedstawiony w artykule zaprojektowany system automatycznego pozycjonowania pozwala na dokładne pozycjonowanie pojazdu względem ładowarki bezprzewodowej, co przyczynia się do minimalizacji strat energii w trakcie procesu ładowania. Precyzja systemu jest zależna od przepustowości sieci. Wykorzystane w projekcie moduły ESP32 bazują na sieci WiFi, w przypadku, gdy jest niska przepustowość, obraz z kamery był zakłócony, przez co prawidłowe działanie systemu jest utrudnione. W przeprowadzonych testach zastosowano ruter mobilny posiadający wbudowaną antenę wzmacniającą przesyłany sygnał. Kolejnym ważnym czynnikiem wpływającym na prawidłowość działania systemu są innowacyjne koła typu Mecanum, które umożliwiły poruszanie się pojazdu pod kątem 90 stopni. Pozwoliło to na optymalne i sprawne podłączenie cewki pojazdu z cewką ładowarki za pomocą dwóch płynnych ruchów.

W celu uniezależnienia projektu od komputera przenośnego, warto rozważyć zastosowanie jednopłytkowego komputera Raspberry Pi, który wspiera programowanie w Pythonie. Chociaż to podejście wymagałoby większego budżetu,

efektywność pozycjonowania znacznie by wzrosła. Jeżeli kamera byłaby bezpośrednio podłączona do wyświetlacza w pojeździe, a wyświetlacz miałby połączenie z Raspberry Pi, na którym wykonywana by była analiza obrazu, opóźnienia w transmisji były by znacząco zredukowane. Jest to potencjalna ścieżka rozwoju dla tego systemu, która mogłaby być użyteczna w rzeczywistych pojazdach elektrycznych. Rozwiązanie z kamera mogłoby stanowić dodatkową opcję kontroli pozycjonowania przez użytkownika.

## LITERATURA

- [1] Kopka J., Pietruszewski R. Wykorzystanie zjawiska indukcji elektromagnetycznej do zasilania samochodu osobowego z napędem elektrycznym. Instytut Naukowo-Wydawniczy "SPATIUM", 2016.
- [2] Czyż P., Cichowski A. Przegląd systemów ładowania elektrycznych osobowych pojazdów i koncepcja dwukierunkowej ładowarki pokładowej. XXVII Seminarium. Zastosowanie komputerów w nauce i technice 2017. Oddział Gdański PTETiS.
- [3] <https://www.emobility-engineering.com/wireless-charging/> (dostęp 18.09.2023).
- [4] Ahmad A., Alam M., Chabaan R. A Comprehensive Review of Wireless Charging Technologies for Electric Vehicles, marzec 2018. IEE.
- [5] Babiuch M., Folýnek P., Smutný P. Using the ESP32 Microcontroller for Data Processing. VSB - Technical University of Ostrava, 2019.
- [6] <https://botland.com.pl/blog/budujemy-robota-mecanum-z-m5stack/> (dostęp 28.09.2023).

## VEHICLE POSITIONING SYSTEM RELATIVE TO A WIRELESS CHARGER

The article presents a vehicle positioning system relative to a wireless charger. The introduction of the paper discusses the development of wireless charging and points out the main problem related to energy loss during this process. As part of the research, the vehicle positioning method was analyzed with the intention of increasing the efficiency of the charging process. To demonstrate the functionality of this method, a remotely controlled vehicle with innovative Mecanum-type wheels was constructed. In the study, vehicle speed measurements were carried out to improve the accuracy of the positioning algorithm.

*(Received: 29.09.2023, revised: 20.11.2023)*