AKADEMIA GÓRNICZO-HUTNICZA IM. STANISŁAWA STASZICA W KRAKOWIE

DZIEDZINA NAUK INŻYNIERYJNO-TECHNICZNYCH

DYSCYPLINA AUTOMATYKA, ELEKTRONIKA, ELEKTROTECHNIKA I TECHNOLOGIE KOSMICZNE

AUTOREFERAT ROZPRAWY DOKTORSKIEJ

Zastosowanie głębokiego uczenia ze wzmocnieniem w robotycznym montażu przemysłowym

Autor: Grzegorz Bartyzel

Promotor rozprawy: dr hab. inż. Paweł Rotter, prof. AGH
Promotor pomocniczy: dr inż. Marcin Węgrzynowski

Praca wykonana: Akademia Górniczo-Hutnicza im. Stanisława Staszica w Krakowie, Wydział Elektrotechniki, Automatyki, Informatyki i Inżynierii Biomedycznej

Kraków, 2025

Streszczenie

Współczesne linie produkcyjne cechują się skomplikowanymi procesami przemysłowymi, które są spowodowane produkcją coraz to bardziej złożonych produktów. Dodatkowo coraz częściej wymaga się od linii produkcyjnych elastyczności, co oznacza, że muszą one być zdolne szybko adaptować się do zmieniających się warunków produkcyjnych. Sytuacja ta jest najbardziej widoczna w przypadku procesów montażu, gdzie złożoność procesu jest związana z koniecznością precyzyjnego pozycjonowania i łączenia różnych elementów. Obecnie coraz częściej stosuje się stanowiska zautomatyzowane, bazujące na robotach przemysłowych, jednak tak wyposażone linie produkcyjne są często niezdolne do szybkiego dostosowania się do montażu często zmieniających się produktów. Dodatkowo tradycyjne metody sterowania robotami (m.in. ręczna definicja trajektorii) często są niewystarczające w przypadku elastycznych linii produkcyjnych. Natomiast uzbrojenie stanowisk montażowych w zaawansowane czujniki, czy też systemy do zmiany narzędzi znacząco podnosi koszty stworzenia i konserwacji linii produkcyjnej. W związku z tym w ostatnich latach coraz większą popularność zdobywają metody oparte na algorytmach wykorzystujących sztuczną inteligencję, a w szczególności uczenie ze wzmocnieniem (ang. *Reinforcement Learning* (RL)).

Celem rozprawy doktorskiej było opracowanie metody bazującej na uczeniu ze wzmocnieniem, która pozwoli na efektywne rozwiązanie problemu montażu różnorodnych elementów na jednej linii produkcyjnej. W celu realizacji tego problemu badawczego stworzono wydajną procedurę uczenia agenta RL na rzeczywistym robocie przemysłowym, a także opracowano algorytm *Multimodal Variational DeepMDP* (MVDeepMDP). Algorytm ten pozwolił na szybką adaptację wstępnie wytrenowanego agenta RL do nowych produktów na liniach produkcyjnych. MVDeepMDP łączy algorytm Soft Actor-Critic (SAC), z rodziny algorytmów *off-policy actor-critic*, z multimodalnym wariacyjnym autoenkoderem. Zaprezentowana metoda charakteryzuje się uczeniem wielomodalnej reprezentacji dynamiki środowiska, dzięki której agent RL uzyskuje istotne informacje o wykonywanym zadaniu. Dodatkowo wspomniana metodologia charakteryzuje się asynchronicznym procesem zbierania danych w stosunku do procesu uczenia, co znacząco usprawnia interakcję agenta RL ze środowiskiem. Co więcej, ramię robota sterowane jest admitancyjnie, dzięki czemu proces uczenia jest bezpieczny i płynny.

Proponowana metoda została przetestowana na problemie montażu elementów elektronicznych do odpowiednich miejsc na płycie drukowanej. Zadanie montażu elektronicznego stanowiło doskonałe środowisko do oceny ogólnej wydajności proponowanego rozwiązania oraz możliwości jego generalizacji, ze względu na różnorodność i złożoność elementów. Wstępne eksperymenty zakładały ewaluację proponowanej metodologii z wykorzystaniem algorytmu SAC. Analiza uzyskanych wyników sugerowała, że przedstawiona metodologia osiągnęła znacznie wyższe wskaźniki jakości niż powszechne techniki montażu. W następnej

kolejności wykonano serię eksperymentów, których celem było sprawdzenie możliwości szybkiej adaptacji MVDeepMDP do nowych montowanych produktów. Wykazano, że metoda ta w większości przypadków jest w stanie montować nowe elementy bez konieczności dodatkowego trenowania. W pozostałych przypadkach, w których błyskawiczne przeniesienie nie powiodło się, douczenie trwało maksymalnie 5 minut.

Podsumowując, zaprezentowane rozwiązanie pozwala na szybkie dostosowanie się zrobotyzowanego stanowiska montażowego do ciągle zmieniających się produktów w elastycznych środowisku produkcyjnym. Funkcjonalność ta jest pożądana z punktu widzenia współczesnych standardów produkcyjnych, co sprawia, że potencjalnie może zastąpić powszechnie stosowane rozwiązania.

Spis treści

Wstęp

1.1 Wprowadzenie

W nowoczesnych systemach produkcyjnych robotyzacja pełni istotną funkcję w automatyzacji powtarzalnych oraz pracochłonnych zadań. W przemyśle manipulatory są używane do realizacji takich działań jak montaż, pobieranie i transportowanie elementów oraz spawanie. Zrobotyzowane stanowiska pracy znajdują swoje zastosowanie głównie w produkcji masowej (ang. *low-mix, high-volume*, LMHV), gdzie jednorodne zadania są wykonywane cyklicznie. Tego rodzaju systemy produkcyjne są powszechnie wykorzystywane w branży motoryzacyjnej, gdzie specjalistyczne linie produkcyjne są zaprojektowane do montażu jednego modelu samochodu.

Natomiast w przypadku produkcji małoseryjnej, (ang. *high-mix, low-volume*, HMLV), gdzie produkty są bardziej zróżnicowane i zmieniają się częściej, coraz bardziej wykorzystywane są tzw. elastyczne systemy produkcyjne (ang. *Flexible Manufacturing Systems*, FMS). Systemy te charakteryzują się możliwością szybkiego dostosowania do zmieniających się warunków produkcyjnych, co pozwala na produkcję różnych produktów na jednej linii produkcyjnej. Systemy te są idealnym rozwiązaniem dla przemysłu elektronicznego, gdzie produkcja związana jest z różnorodnymi produktami, mającymi często krótki cykl życia. Od elastycznych systemów produkcyjnych oczekuje się możliwości szybkiego przezbrojenia zautomatyzowanych stanowisk pracy, takich jak roboty przemysłowe, co pozwala na zmianę procesu produkcyjnego w krótkim czasie. Osiągniecie tego celu w praktyce jest jednak skomplikowane, ze względu na złożoność procesów produkcyjnych, oraz zapewnienie bezpiecznej interakcji robota przemysłowego z otaczającym go środowiskiem. Co więcej, zadania wymagające wysokiej precyzji oraz zręczności, takie jak montaż elementów elektronicznych czy też podzespołów silników spalinowych, są wciąż wykonywane przez pracowników produkcji. Niestety, pomimo niezastąpionej sprawności motorycznej, ludzie są podatni na zmęczenie, co skutkuje obniżeniem produktywności, a także zwiększeniem szansy na popełnienie błędu. W związku z tym można zaobserwować coraz większe zainteresowanie rozwojem zrobotyzowanych systemów produkcyjnych, które odzwierciedlą ludzką motoryczność oraz umiejętność szybkiego dostosowania się do nowych zadań.

W ostatnich latach poczyniono znaczący postęp w dziedzinie robotyki, a w szczególności w obszarze manipulacji o częstym kontakcie. Obszar ten określa takie zadania jak: chwytanie, montaż obiektów, a także

ruch za zachowaniem odpowiednich sił. Zadanie te cechują się tym, że w trakcie manipulacji dochodzi do częstej interakcji pomiędzy robotem a środowiskiem, w którym pracuje. Niestety, osiągnięcie tego typu manipulacji jest skomplikowanym zadaniem, ze względu na to, że system sterowania robota przemysłowego musi uwzględniać w swoich obliczeniach własności fizyczne przestrzeni roboczej takie jak tarcie, czy też siły kontaktowe.

Konwencjonalne metody, które są aplikowane do tego typu problemów, najczęściej wykorzystują hybrydowe lub impedancyjne systemy sterowania w celu zapewnienia bezpiecznej interakcji pomiędzy robotem a obiektami znajdującymi się w jego przestrzeni roboczej. Rozwiązania te charakteryzują się tym, że robot wykonuje z góry zaprogramowaną trajektorię przy jednoczesnym zachowaniu zadanych sił lub założonym profilu impedancyjnym. Pomimo tego, że metody bazujące na sterowaniu siłą zostały pomyślnie wdrożone w wielu przemysłowych aplikacjach (i nie tylko), to jednak wymóg ręcznego dostrajania parametrów znacząco ogranicza ich zdolności szybkiego dostosowania się do nowych zadań, czy też manipulowanych obiektów.

We współczesnych systemach produkcyjnych, możliwości zrobotyzowanych stanowisk opartych na sterowaniu siłą powszechnie rozszerza się poprzez zastosowanie tzw. modułów wymiany narzędzi. Moduły te pozwalają wykonywać różnego rodzaju zadania na pojedynczym stanowisku poprzez zastosowanie dedykowanego narzędzia. Niemniej jednak, rozwiązanie to nie jest bez wad, mianowicie, maszyny wyposażone w moduł wymiany wymagają użycia dodatkowego sprzętu, a także, podczas przezbrojenia linia produkcyjna musi zostać zatrzymana. Co więcej, narzędzia umieszczone w module są dedykowane tylko do pewnych obiektów, jakie mogą znaleźć się na linii produkcyjnej. Jak widać, pomimo swojej prostoty oraz efektywności, rozwiązania oparte na modułach wymiany narzędzi są drogie oraz wymagają znaczącej przestrzeni, która pozwoli umieścić potrzebne narzędzia.

Alternatywnym podejściem zwiększającym elastyczność zrobotyzowanych systemów produkcyjnych jest zastosowanie systemów wizyjnych, które dostarczają informacji o środowisku, w jakim operuje robot. Wizyjna informacja zwrotna często wykorzystywana jest do określenia docelowej pozycji narzędzia lub modyfikacji wykonywanej trajektorii w zależności od obecnego zadania. Jednakże tego typu rozwiązania bazują na złożonych algorytmach do przetwarzania obrazów oraz wymagają stabilnego warunków oświetleniowych, aby móc funkcjonować powtarzalnie. Co więcej, osiągnięcie jednostajnej wydajność oraz możliwości ponownego użycia do manipulacji innych obiektów jest procesem czasochłonnym, oraz kosztownym (koszt pracy inżyniera). W związku tym, rozwiązania bazujące na systemach wizyjnych najczęściej są projektowane pod dedykowany zestaw zadań lub obiektów. Co więcej, w przypadku wysoko precyzyjnych zadań, koszt tego typu systemów znacząco wzrasta, ponieważ wymagane jest użycie kamer o wysokiej rozdzielczości, oraz większej mocy obliczeniowej do przetwarzania zaawansowanych algorytmów.

Ostatnie postępy głębokiego uczenia ze wzmocnieniem (ang. *deep reinforcement learning*, DRL) otworzyły nowe możliwości rozwoju zrobotyzowanych systemów zdolnych do wykonywania precyzyjnych zadań przy jednoczesnej możliwości adaptacji. Uczenie ze wzmocnieniem jest jednym z paradygmatów uczenia maszynowego, w którym to agent uczy się rozwiązać dane mu zadanie, poprzez interakcję ze środowiskiem oraz otrzymanie informacji zwrotnej w postaci nagrody określającej jakość podjętej decyzji. Jak widać, zastosowanie uczenia ze wzmocnieniem w zrobotyzowanych systemach produkcyjnych może do-

starczyć rozwiązania dla zadań, które dotychczasowy były zbyt skomplikowane dla tradycyjnych metod. Co więcej, algorytmy DRL aproksymują funkcję polityki poprzez głębokie sieci neuronowe, w związku z czym danewejściowe nie muszą być wstępnie procesowane a parametry systemów sterowania mogą być dostrajane na bieżąco. Tak więc, w teorii, zastosowanie DRL powinno obniżyć całościowe koszty oraz usprawnić możliwość wykorzystania modelu w innych zadaniach. Jednakże należy wspomnieć, że bazowe algorytm RL często mają problemy z szybką adaptacją do nowych zadań, czy też obiektów, a uczenie modelu od początku, podobnie jak w przypadku systemów wizyjnych, jest procesem czasochłonnym. W związku z tym, osiągnięcie szybkiej adaptacji bądź też natychmiastowego transferu do nowych zadań jest wciąż otwartym problemem naukowym.

1.2 Motywacja oraz cel pracy

Prace badawcze przeprowadzone w ramach doktoratu wykonano przy współpracy z firmą Fitech Sp. z o.o. w ramach projektu pt. "Intelligent robot for autonomous handling of assembly of electronic components based on artificial intelligence and neural networks"nr. POIR.01.01.01-00-0123/19, finansowanego przez Narodowe Centrum Badań i Rozwoju (NCBiR). Celem tego projektu było opracowanie bazującego na sztucznej inteligencji rozwiązania problemu montażu nieznormalizowanych elementów elektronicznych na płytach drukowanych (ang. *Printed Circuit Board*, PCB). Jak wiadomo, geometria tego typu elementów jest bardzo zróżnicowana, przez co zastosowanie powszechnie stosowanych maszyn do montażu przewlekanego (ang. *Through-Hole Technology*, THT) jest niemożliwe lub zbyt kosztowne. Co więcej, rozwiązania oparte na klasycznych systemach sterowania takich jak sterowanie siłą, które wykonują z góry zaprogramowaną trajektorię, są niewystarczające i wymagają dodatkowego systemu wizyjnego, aby zapewnić wysoką precyzję montażu. W związku z tym podjęto decyzję o przeprowadzeniu badań nad opracowaniem systemu zdolnego do montażu różnorodnych elementów elektronicznych na płytach drukowanych z wykorzystaniem algorytmu uczenia ze wzmocnieniem.

Ze względu na specyfikę systemów produkcyjnych o dużej zmienności produktów, nie jest możliwe zastosowanie podstawowych algorytmów uczenia ze wzmocnieniem takich jak Soft Actor-Critic [5], Proximal Policy Optimization (PPO) [7], czy też Twina Delayed Deep Deterministic Policy Gradient (TD3) [3], ponieważ napotyka się na problem związany z generalizacją nauczonych agentów. Algorytmy te najlepiej działają na wariancie zadania, który był użyty podczas uczenia. Natomiast, gdy agent RL jest wystawiony do rozwiązania zadania, które odbiega od bazowego (np. zmiana montowanego elementu), to w większości przypadków nie jest on w stanie zakończyć go pomyślnie. Kwestia generalizacji jest wciąż otwartym problemem, a jej częściowe rozwiązanie, chociażby przez szybką adaptację nauczonego agenta RL do nowych zadań znacząco poprawiłoby możliwość zastosowania algorytmów uczenia ze wzmocnieniem do rzeczywistych problemów. Co więcej, dla aplikacji bazujących na uczeniu ze wzmocnieniem, które sterują rzeczywistym obiektem takim jak np. manipulator, krytycznym czynnikiem jest zapewnienie płynnej pętli sterowania podczas uczenia. Tradycyjne podejście do uczenia, czyli sekwencyjny proces zbierania danych oraz aktualizacji sieci neuronowej niestety znacząco zaburza płynne sterowanie. W związku z tym, dla tego typu aplikacji, istotne jest opracowanie odpowiedniej architektury programistycznej, która zapewni proces zbierania danych, w którym to aktualizacja parametrów agenta RL nie wpływa na wydajność pętli sterowa-

nia. Na podstawie przedstawionego powyżej problemu zastosowania algorytmów uczenia ze wzmocnieniem do rzeczywistych aplikacji, sformułowano następujące cele badawcze:

- Opracowanie systemu, który pozwoli na przeprowadzenie efektywnego procesu uczenia agenta RL w rzeczywistym środowisku produkcyjnym. System ten powinien zapewnić płynne sterowanie robotem przemysłowym podczas uczenia oraz zbierania danych.
- Opracowanie metody zwiększającej możliwości generalizacji nauczonego agenta RL do nowych zadań. Metoda ta, w kontekście zrobotyzowanego procesu montażu, powinna pozwolić na natychmiastowy transfer wiedzy lub szybką adaptację agenta RL do nowych produktów montowanych na linii produkcyjnej.

Najważniejsze wyniki badań

2.1 Zastosowanie głębokiego uczenia ze wzmocnieniem w zrobotyzowanym montażu przemysłowym

Pierwszym istotnym osiągnięciem zaprezentowanym w rozprawie doktorskiej jest opracowanie rozwiązania umożliwiającego skuteczne zastosowanie głębokiego uczenia ze wzmocnieniem w kontekście zrobotyzowanego montażu przemysłowego. Proponowane podejście zostało zweryfikowane eksperymentalnie w rzeczywistym środowisku produkcyjnym, w którym agent RL uczył się montażu elementów elektronicznych na płytce PCB. Wyniki eksperymentów oraz ogólna architektura systemu umożliwiającego uczenie agenta w środowisku rzeczywistym zostały szczegółowo przedstawione w publikacji [4]

2.1.1 Środowisko do zrobotyzowanego montażu przemysłowego

Na potrzeby badań nad zastosowaniem DRL w zrobotyzowanym montażu przemysłowym zaprojektowano i zrealizowano dedykowane środowisko testowe, zgodne z formalizmem procesu decyzyjnego Markowa. Stworzony system umożliwia prowadzenie eksperymentów w warunkach odpowiadających rzeczywistym operacjom montażowym, z uwzględnieniem fizycznych interakcji robota z otoczeniem.

Agent RL podejmuje decyzje na podstawie multimodalnych obserwacji obejmujących: obrazy RGB z kamer zamontowanych na narzędziu robota, relatywne położenie narzędzia względem pozycji montażowej, 6-osiową prędkość oraz pomiar sił i momentów. Na tej podstawie agent podejmuje decyzję o wykonaniu akcji, która jest zdefiniowana jako czterowymiarowy wektor przemieszczenia i rotacji narzędzia robota w przestrzeni kartezjańskiej $a_t = [\Delta x, \Delta y, \Delta z, \Delta\theta]$, gdzie $\Delta\theta$ jest rotacją narzędzia wokół osi Z. Tak zdefiniowana akcja jest transformowana do układu współrzędnych robota i przekazywana do systemu sterowania typu admitancyjnego. Sterownik ten umożliwia adaptację trajektorii ruchu narzędzia do aktualnie działających sił i momentów, co znacząco zwiększa bezpieczeństwo operacji montażowych i ogranicza ryzyko uszkodzenia manipulowanych komponentów.

Podczas każdej interakcji z otoczeniem agent otrzymuje nagrodę zdefiniowaną jako:

$$r = -\tanh\left(\alpha \cdot d\right), \tag{2.1}$$

gdzie d oznacza odległość narzędzia od punktu montażowego, a α jest współczynnikiem skalującym. Funkcja ta ma na celu motywowanie agenta do możliwie szybkiego i precyzyjnego zbliżenia się do celu. W przypadku osiągnięcia pozycji montażowej agent otrzymuje nagrodę końcową $r = 10$, natomiast w przypadku przekroczenia dopuszczalnej orientacji narzędzia względem osi Z przyznawana jest kara $r = -2$. Kara ta ma na celu ograniczenie ryzyka uszkodzenia okablowania narzędzia robota.

2.1.2 Architektura systemu uczenia agenta

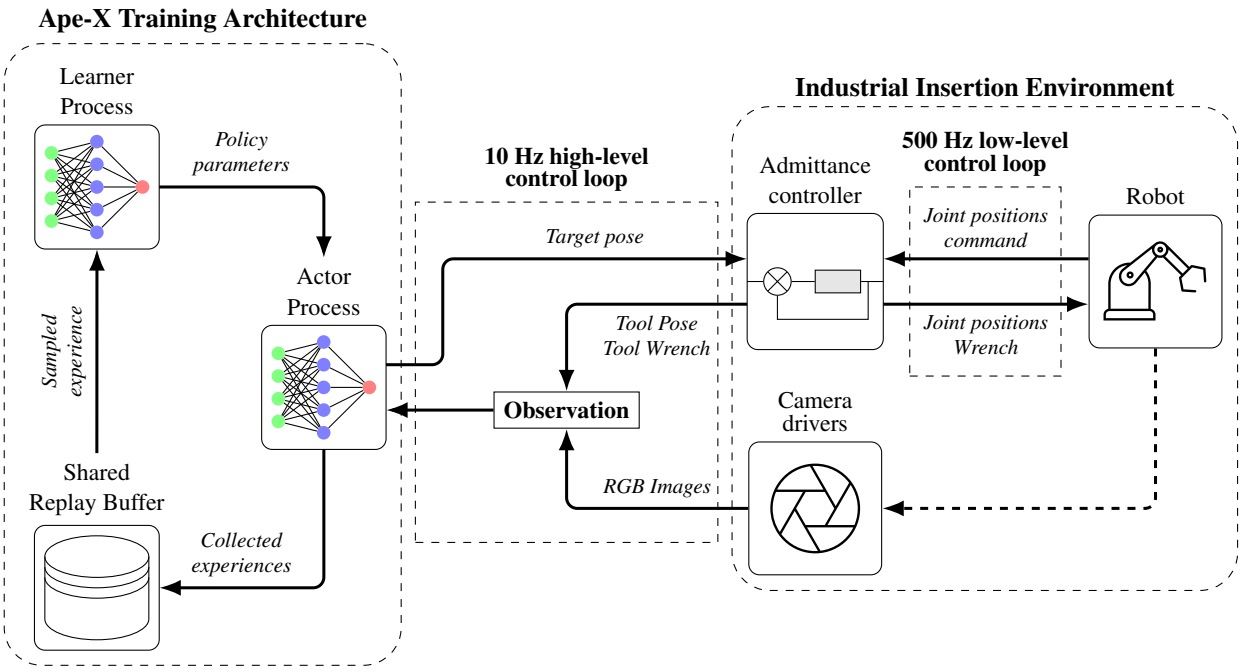
Jednym z kluczowych osiągnięć przedstawionych w rozprawie doktorskiej jest opracowanie architektury systemu umożliwiającej efektywne uczenie agenta RL w środowisku rzeczywistym. Zaprojektowany system składa się z dwóch głównych komponentów: wysokopoziomowej pętli sterowania, w której członem sterującym jest agent RL, oraz niskopoziomowej pętli sterowania, w której system sterowania typu admitancyjnego realizuje zadane przez agenta akcje. Wysokopoziomowa pętla działa z częstotliwością $f_o = 10$ Hz, generując sygnały sterujące, które przekazywane są do niskopoziomowej pętli. Jednocześnie agent otrzymuje aktualne informacje o stanie środowiska. Z kolei niskopoziomowa pętla steruje bezpośrednio robotem przemysłowym z wyższą częstotliwością $f_i = 500$ Hz, zapewniając precyzyjne i płynne wykonanie działań. W roli członu decyzyjnego w pętli wysokopoziomowej wykorzystano algorytm SAC [5], który ze względu na swoje właściwości – stabilność i efektywność – należy do najczęściej stosowanych metod RL w robotyce.

Dodatkowo, aby zwiększyć wydajność procesu uczenia w warunkach rzeczywistych, system został rozszerzony o asynchroniczną architekturę uczenia Ape-X [6]. Rozwiązanie to wykorzystuje jeden proces główny odpowiedzialny za aktualizację parametrów sieci neuronowych oraz wiele procesów roboczych, które asynchronicznie zbierają dane z otoczenia i przesyłają je do procesu głównego. Dzięki temu rozwiązaniu proces aktualizacji parametrów sieci neuronowej nie zakłóca pętli sterowania robota, a agent RL jest w stanie uczyć się w czasie rzeczywistym. Schemat przedstawiający architekturę systemu został przedstawiony na Rysunku 2.1.

Implementacja systemu

Wysokopoziomowa pętla sterowania oraz komponent odpowiedzialny za proces uczenia agenta zostały zaimplementowane w języku Python z wykorzystaniem bibliotek RLlib oraz Gymnasium. Komunikacja z robotem oraz urządzeniami peryferyjnymi (kamera, 6-osiowy czujnik sił i momentów) zrealizowana została w języku C++ z użyciem middleware ROS 2.

Ze względu na asynchroniczny charakter komunikacji w ROS 2 konieczne było wprowadzenie dodatkowej warstwy pośredniczącej, zapewniającej spójną synchronizację danych pomiędzy agentem a systemem sterowania. W tym celu zaprojektowano i zaimplementowano komponent nazwany *ROS 2–Gym Bridge* – oddzielny węzeł ROS, napisany w C++, który wykorzystuje protokół XML-RPC do wymiany danych pomiędzy środowiskiem a ROS 2. Rozwiązanie to gwarantuje, że agent RL otrzymuje poprawnie zsynchronizowane i aktualne dane o stanie otoczenia, niezbędne do podejmowania decyzji.



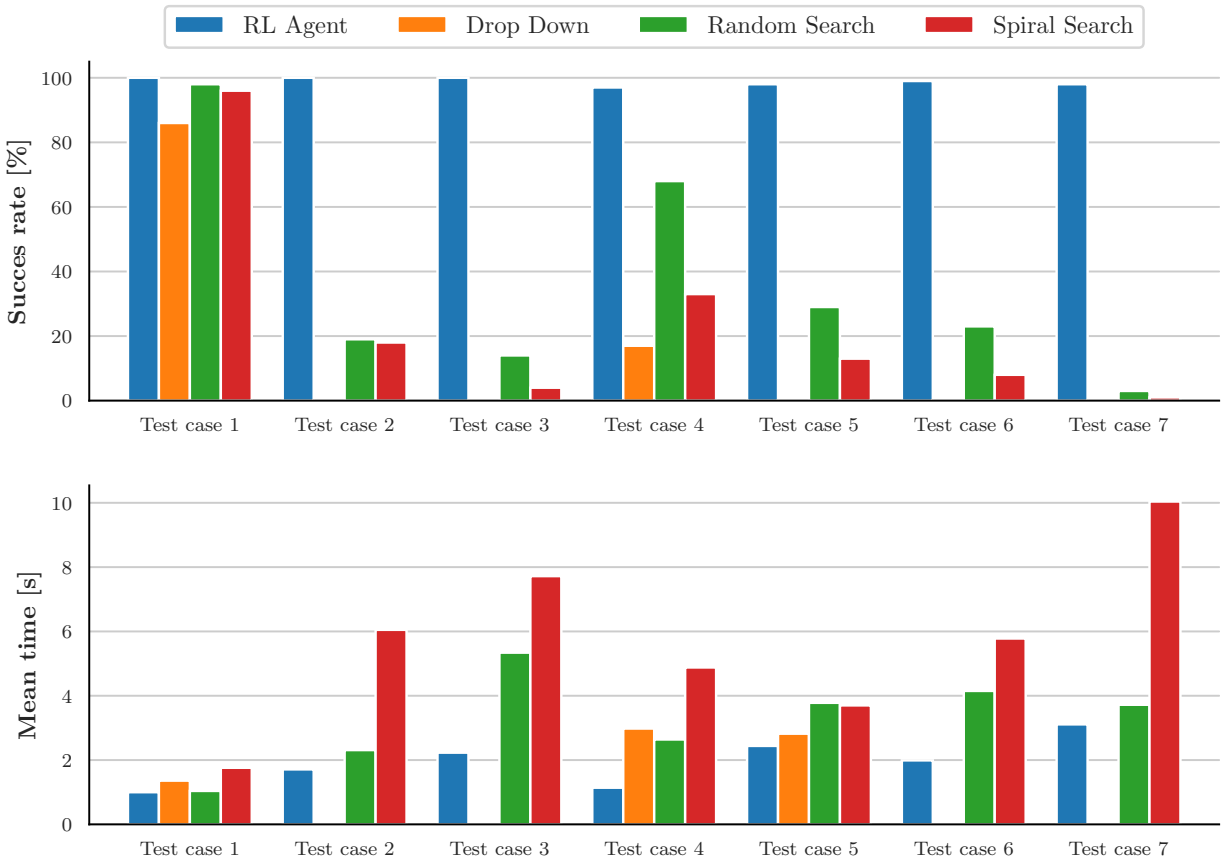
Rysunek 2.1: Schemat blokowy systemu sterowania dla zrobotyzowanego montażu przemysłowego. **Po lewej:** architektura Ape-X dla asynchronicznego uczenia ze wzmocnieniem. **Po prawej:** środowisko montażu przemysłowego z kontrolerem admitancyjnym oraz oprogramowaniem do obsługi urządzeń peryferyjnych. W wysokopoziomowej pętli sterowania członem sterującym jest agent RL, który wysyła docelową pozycję $\boldsymbol{x}_d$ do kontrolera admitancyjnego z częstotliwością $10\,\mathrm{Hz}$. Kontroler ten następnie steruje pozycjami przegubów $\boldsymbol{q}_r$ robota z częstotliwością $500\,\mathrm{Hz}$.

2.1.3 Wydajność uczenia ze wzmocnieniem w zrobotyzowanym montażu przemysłowym

Kolejnym istotnym elementem prezentowanej rozprawy doktorskiej jest eksperymentalna weryfikacja zastosowania głębokiego uczenia ze wzmocnieniem w zadaniach zrobotyzowanego montażu przemysłowego. W szczególności porównano wydajność agenta RL do wydajności tradycyjnych algorytmów planowania ruchu robotów przemysłowych.

W tym celu opracowano scenariusz testowy obejmujący siedem różnych pozycji początkowych narzędzia robota – od położenia najbliższego miejscu montażu do pozycji oddalonej o $10\,\mathrm{mm}$. Dla każdej pozycji przeprowadzono 100 niezależnych prób montażu, rejestrując zarówno czas zakończenia epizodu, jak i odsetek prób zakończonych sukcesem. Wyniki eksperymentów wykazały, że agent RL osiągał skuteczność przekraczającą 95% niezależnie od położenia początkowego, a czas wykonania zadania mieścił się w przedziale od $1.5\,\mathrm{s}$ do $2.5\,\mathrm{s}$. Dla porównania skuteczność klasycznych algorytmów planowania znacząco malała wraz ze wzrostem odległości od punktu montażowego — w niektórych przypadkach spadając nawet do 0%. Omówione wyniki zostały przedstawione na Rysunku 2.2.

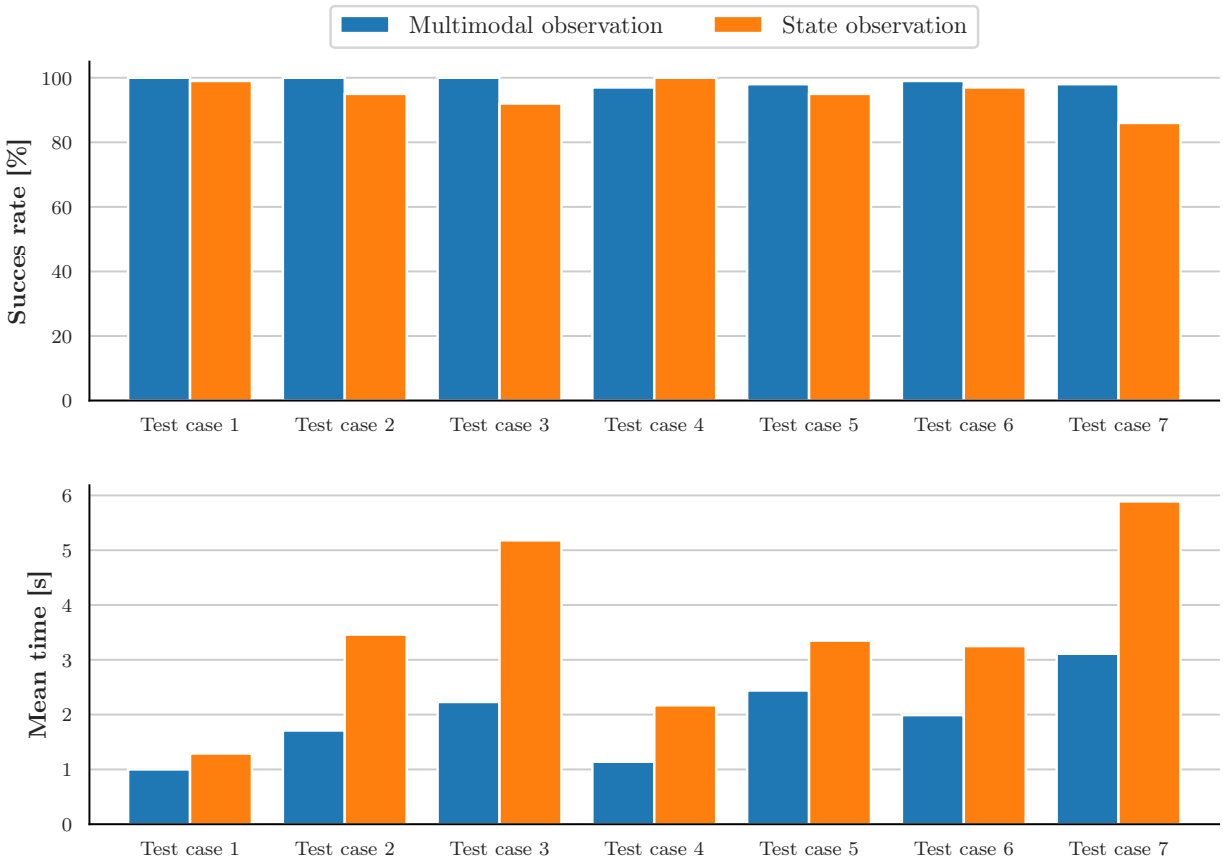
Dodatkowo przeprowadzono analizę wpływu rodzaju obserwacji na efektywność procesu uczenia agenta. Szczególną uwagę poświęcono obecności informacji wizyjnej w multimodalnym wektorze obserwacji. Eksperymenty wykazały, że obecność obrazów RGB znacząco skraca czas potrzebny do osiągnięcia wysokiej skuteczności. Bez informacji wizyjnej agent przekroczył próg 90% skutecznych montażów do-



Rysunek 2.2: Porównanie wydajności agenta RL i konwencjonalnych metod montażu takich jak: pionowy ruch w dół (ang. *Straight Down*), strategia losowego wyszukiwania (ang. *Random Search*), oraz strategia wyszukiwania po trajektorii spiralnej (ang. *Spiral Search*) w przypadku zaburzeń pozycji początkowej podczas montażu. **Górny wykres:** odsetek udanych prób montażu dla 100 prób. **Dolny wykres:** średni czas montażu dla 100 prób w sekundach.

piero po około 2 godzinach treningu. W wariancie z informacją wizyjną taki sam poziom skuteczności uzyskano już po około 25 minutach. Co więcej, choć ostateczna skuteczność była porównywalna w obu wariantach, czas realizacji zadania istotnie się różnił. Agent dysponujący informacją wizyjną wykonywał montaż w czasie od 1.0 s do 3.0 s, podczas gdy agent operujący bez obrazu potrzebował od 2.0 s do 6.0 s. Wyniki te jednoznacznie wskazują, że zastosowanie multimodalnej obserwacji — szczególnie z uwzględnieniem sygnału wizyjnego — istotnie poprawia zarówno czas uczenia, jak i wydajność działania agenta RL w środowisku rzeczywistym. Wyniki z przeprowadzonych eksperymentów zostały przedstawione na Rysunku 2.3.

Powyższe obserwacje potwierdzają potencjał metod głębokiego uczenia ze wzmocnieniem, wspieranych przez multimodalną percepcję, w zastosowaniach związanych ze zrobotyzowanym montażem przemysłowym.



Rysunek 2.3: Porównanie wydajności algorytmu SAC z obserwacją zawierającą informację wizyjną oraz bez niej. Można zauważyć, że największa różnica występuje w przypadku średniego czasu montażu, gdzie algorytm z obserwacją wizyjną osiąga lepsze wyniki. W przypadku wskaźnika sukcesu różnica jest mniejsza, ale również zauważalna. **Górny wykres:** Odsetek udanych prób montażu dla 100 prób. **Dolny wykres:** Średni czas montażu dla 100 prób w sekundach.

2.2 Multimodal Variational DeepMDP

Najważniejszym zadaniem w prezentowanej rozprawie doktorskiej było opracowanie i opisanie nowatorskiej metody o nazwie Multimodal Variational DeepMDP [1]. Metoda ta łączy probabilistyczne wnioskowanie z multimodalnych obserwacji z metodologią zaproponowaną w algorytmie DeepMDP [2]. Dzięki czemu zaproponowany algorytm cechuje się znacząco wyższą zdolnością generalizacji agentów RL względem innych znanych algorytmów.

Głównym celem MVDeepMDP jest nauka wspólnej reprezentacji multimodalnych obserwacji poprzez minimalizację wariacyjnego dolnego ograniczenia (ang. *variational lower bound*, ELBO). W szczególności funkcja celu modelu składa się z dwóch części - jednej odpowiedzialnej za predykcję przyszłych ukrytych stanów oraz drugiej odpowiedzialnej za przewidywanie nagrody.

Funkcja celu związana z dynamiką środowiska (predykcją kolejnego ukrytego stanu) ma postać:

$$\begin{aligned}\mathcal{J}_{\text{DYN}}(\mathbf{\Phi}) = \frac{1}{M}\sum_{i=0}^{M} D_{\text{KL}}\big(q_{\phi_i}\left(\mathbf{z}_{t+1}^{i}\middle|\mathbf{x}_{t}^{i}\right) \parallel p_{\phi_i}\left(\widetilde{\mathbf{z}}_{t+1}^{i}\middle|\mathbf{z}_{t}^{i},\mathbf{a}_t\right)\big) \\ + D_{\text{KL}}\Big(q_{\mathbf{\Phi}}\left(\mathbf{z}_{t+1}^{joint}\middle|\mathbf{X}_t\right) \parallel p_{\mathbf{\Phi}}\left(\widetilde{\mathbf{z}}_{t+1}^{joint}\middle|\mathbf{Z}_t,\mathbf{a}_t\right)\Big)\end{aligned} \tag{2.2}$$

Funkcja celu związana z predykcją nagrody zdefiniowana jest jako:

$$\begin{aligned}\mathcal{J}_{REW}(\mathbf{\Phi}) = \frac{1}{M}\sum_{i=0}^{M} \mathbb{E}_{\mathbf{z}_t^i\sim q_{\phi_i}(\cdot|\mathbf{x}_t^i)}\left[\log p_{\phi_i}\left(r_t\middle|\mathbf{z}_t^i,\mathbf{a}_t\right)\right] \\ + \mathbb{E}_{\mathbf{z}_t^{joint}\sim q_{\mathbf{\Phi}}(\cdot|\mathbf{X}_t)}\left[\log p_{\mathbf{\Phi}}\left(r_t\middle|\mathbf{z}_t^{joint},\mathbf{a}_t\right)\right]\end{aligned} \tag{2.3}$$

Całkowita funkcja celu modelu MVDeepMDP przyjmuje postać:

$$\mathcal{J}_M(\mathbf{\Phi}) = \mathcal{J}_{REW}(\mathbf{\Phi}) - \beta\mathcal{J}_{DYN}(\mathbf{\Phi}) \tag{2.4}$$

gdzie M jest liczbą modalności, $\mathbf{X}_t$ jest multimodalną obserwacją, $\mathbf{z}_t^i$ jest ukrytym stanem pojedynczej modalności, $\mathbf{z}_t^{joint}$ jest multimodalnym ukrytym stanem, $\mathbf{a}_t$ jest akcją, r_t jest nagrodą, a β jest współczynnikiem regularyzującym.

W zaproponowanej metodzie uwspólniona reprezentacja multimodalnych stanów ukrytych $\mathbf{z}_t^{joint}$ stanowi wejście do polityki oraz funkcji wartości w algorytmie SAC. Funkcje straty SAC pozostają bez zmian i odpowiadają za minimalizację błędu Bellmana oraz maksymalizację oczekiwanej skumulowanej nagrody. Pełna procedura aktualizacji parametrów sieci neuronowych w ramach MVDeepMDP została przedstawiona na Algorytmie 1.

2.2.1 Weryfikacja możliwości generalizacji

Możliwości generalizacji zaproponowanego algorytmu MVDeepMDP zostały zweryfikowane w eksperymentach odzwierciedlających rzeczywiste wyzwania, jakie mogą wystąpić w zrobotyzowanym montażu przemysłowym w kontekście elastycznej produkcji. W tym celu opracowano dwa scenariusze testowe:

Scenariusz 1: Montaż tego samego elementu elektronicznego na różnych płytkach PCB.

Scenariusz 2: Montaż różnych elementów elektronicznych na różnych płytkach PCB.

W obu przypadkach wyniki uzyskane przez MVDeepMDP zostały porównane z wynikami innych metod, uznawanych za reprezentatywne w kontekście generalizacji w uczeniu ze wzmocnieniem.

Dla większości przypadków testowych MVDeepMDP osiągnął skuteczność powyżej 95% udanych prób montażu. Dodatkowo wykazał się przewagą czasową względem konkurencyjnych metod, co podkreśla jego praktyczną efektywność oraz zdolność do generalizacji. Co więcej,w przypadku scenariusza drugiego, dla dwóch spośród sześciu elementów testowych, które nie były obecne w zbiorze treningowym, skuteczność agenta spadła poniżej 90%. Wyniki te przedstawiono w Tabeli 2.1. W odpowiedzi na te obserwacje przeprowadzono dalsze eksperymenty mające na celu ocenę możliwości szybkiej adaptacji modelu do nowych zadań montażowych. Eksperymenty adaptacyjne wykazały, że agent MVDeepMDP był w stanie dostosować się do nowych komponentów i osiągnąć niemal 100% skuteczności już po około 5 minutach ponownego

Algorithm 1 Multimodal Variational DeepMDP

Parameters: learning rates η_Q, η_π, η_M, target update coefficient υ

Initialize function approximators parameters $\theta_Q, \theta_\pi, \mathbf{\Phi}$, temperature coefficient α

Replay buffer $\mathcal{D}$

for each iteration **do**
  Initialize state $\mathbf{X}_t$
  for each environment step **do**
    $\mathbf{z}_t^{joint} \sim p_{\mathbf{\Phi}}^{gPoE}\left(\mathbf{z}_t|\mathbf{X}_t\right)$ ▷ Sample latent state at time t
    $\mathbf{a}_t \sim \pi_\theta\left(\mathbf{a}_t\middle|\mathbf{z}_t^{joint}\right)$ ▷ Sample action from policy
    $\mathbf{X}_{t+1} \sim Pr\left(\mathbf{X}_{t+1}|\mathbf{X}_t, \mathbf{a}_t\right), r_t = R\left(\mathbf{X}_t, \mathbf{a}_t\right)$ ▷ Sample next state and reward
    $\mathcal{D} \leftarrow \mathcal{D} \cup \left(\mathbf{X}_t, \mathbf{a}_t, r_t, \mathbf{X}_{t+1}\right)$ ▷ Store transition in replay buffer
    $\mathbf{X}_t \leftarrow \mathbf{X}_{t+1}$
  end for
  for each update step **do**
    $\{(\mathbf{X}_t, \mathbf{a}_t, r_t, \mathbf{X}_{t+1})\} \sim \mathcal{D}$ ▷ Sample batch of transitions from replay buffer
    $\theta_Q \leftarrow \theta_Q + \eta_Q \nabla_{Q_\theta} \mathcal{J}_Q(\theta)$ ▷ Update Q-function parameters with SAC soft Q-function loss
    $\mathbf{\Phi} \leftarrow \mathbf{\Phi} + \eta_M \nabla_M \mathcal{J}_M(\mathbf{\Phi})$ ▷ Update model parameters with (2.4)
    $\theta_\pi \leftarrow \theta_\pi + \eta_\pi \nabla_{\pi_\theta} \mathcal{J}_\pi(\theta)$ ▷ Update policy parameters with SAC policy loss
    $\alpha \leftarrow \alpha + \eta_\alpha \nabla_\alpha \mathcal{J}_\alpha(\alpha)$ ▷ Update temperature coefficient with SAC temperature loss
    $\bar{\theta} \leftarrow (1-\upsilon)\bar{\theta} + \upsilon\theta$ ▷ Update $\bar{\theta}$ using EMA of θ, where υ defines Polyak coefficient
  end for
end for

uczenia. Wynik ten świadczy o dużej plastyczności modelu i jego zdolności do efektywnego transferu wiedzy.

Co więcej, w celu dalszej oceny zdolności generalizacji, przeprowadzono testy z wykorzystaniem innych typów obiektów – w szczególności dużych wydrukowanych klocków o różnych geometriach. Agent poradził sobie z tym zadaniem bez problemów osiągając 100% skuteczność dla wszystkich testowych klocków. Jednak, że warto zaznaczyć, że zadanie to było prostsze niż montaż precyzyjnych komponentów elektronicznych (ze względu na większe rozmiary i tolerancje otworów montażowych).

Podsumowując, wyniki uzyskane w ramach szeroko zakrojonych eksperymentów potwierdzają, że MVDeepMDP znacząco zwiększa możliwości generalizacji w uczeniu ze wzmocnieniem i może być z powodzeniem stosowany w zrobotyzowanym montażu przemysłowym, również w kontekście zmiennych warunków i zróżnicowanych zadań.

2.2.2 Analiza wpływu komponentów MVDeepMDP na generalizację

W celu lepszego zrozumienia wpływu poszczególnych komponentów architektury MVDeepMDP na zdolność generalizacji agenta RL przeprowadzono szereg eksperymentów porównujących różne warianty zaproponowanego algorytmu.

Tabela 2.1: Porównanie możliwości transferu w obrębie tej samej domeny dla metod referencyjnych oraz MVDeepMDP. Dla każdego obiektu i metody przeprowadzono 100 prób montażu. Wyniki pokazują, że MVDeepMDP przewyższa metody referencyjne pod względem wskaźnika sukcesu oraz średniego czasu realizacji zadania.

	Success rate [%]					
	Type 2	Type 3	Type 4	Type 5	Type 6	Type 7
SAC	6	7	1	15	10	5
DrQ	2	0	28	18	13	27
SVEA	43	95	84	12	68	64
PAD	1	32	81	18	6	52
DeepMDP	4	93	91	81	29	31
DBC	56	97	80	79	90	21
MVDeepMDP	**87**	**100**	**100**	**100**	**99**	**85**
	Mean time [s]					
SAC	8.07	8.71	9.10	8.23	6.34	8.93
DrQ	8.57	9.20	7.38	7.96	8.42	7.56
SVEA	6.43	2.76	3.41	8.41	4.99	5.07
PAD	13.85	10.94	5.09	11.91	13.27	8.17
DeepMDP	9.02	2.54	2.92	4.02	7.43	7.37
DBC	6.04	2.91	3.82	4.31	3.72	8.11
MVDeepMDP	**3.59**	**2.15**	**1.83**	**2.08**	**2.13**	**3.55**

Wpływ mechanizmu fuzji modalności

Pierwszy eksperyment dotyczył oceny wpływu mechanizmu probabilistycznego wnioskowania wielomodalnego na wydajność agenta. Porównano wydajność agenta MVDeepMDP wykorzystującego mechanizm gPoE (generalizowany iloczyn ekspertów) z następującymi wariantami:

MVDeepMDP (PoE): wariant z klasycznym iloczynem ekspertów (ang. *Product-of-Experts* PoE),

MVDeepMDP (MoE): wariant z mieszanką ekspertów (ang. *Mixture-of-Experts* MoE),

MVDeepMDP (Naive): wariant wykorzystujący proste uśrednianie modalności z wykorzystaniem średniej arytmetycznej.

Wyniki jednoznacznie wskazały na przewagę wariantu opartego o gPoE, który zapewniał wyraźnie wyższą skuteczność montażu w testach transferu. Co więcej, analiza jakościowa przy użyciu metody t-SNE wykazała, że wspólna reprezentacja ukryta generowana przez mechanizm gPoE była bardziej zwarta. Oznacza to, że mechanizm ten lepiej izoluje cechy istotne dla danego zadania

Wpływ architektury enkoderów dla danych fizycznych

Drugą analizą objęto sposób przetwarzania obserwacji opisujących wartości fizyczne, takie jak siły i momenty. W literaturze dominującym podejściem jest ich łączenie w jeden wektor cech, a następnie przetwarzanie przez wspólny enkoder. W przeciwieństwie do tego MVDeepMDP zakłada przetwarzanie każdej modalności — także tych reprezentujących dane fizyczne — przez odrębny dedykowany enkoder.

Aby ocenić skutki tej decyzji, przeprowadzono eksperyment sprawdzający zdolność transferu agenta RL do nieznanych elementów elektronicznych. Uzyskane wyniki wykazały, że klasyczne podejście znacząco obniża skuteczność montażu, redukując odsetek udanych prób do poziomu poniżej 50%. Z kolei wariant MVDeepMDP z dedykowanymi enkoderami zachował wysoką skuteczność, co potwierdza zasadność projektowej separacji modalności.

Podsumowanie

W rozprawie zaprezentowano efektywną metodę dla robotycznego montażu przemysłowego, opartą na głębokim uczeniu ze wzmocnieniem. Głównym elementem tej metody jest autorski algorytm Multimodal Variational DeepMDP, który znacząco zwiększa możliwość generalizacji agenta RL do nowych zadań. Istotną cechą opisanych eksperymentów jest fakt, że przeprowadzone je w środowisku laboratoryjnym, które odzwierciedla rzeczywiste warunki przemysłowe. W przygotowanym środowisku zadaniem agenta RL było zamontowanie elementów elektronicznych w otwory na płytce drukowanej.

Do najważniejszych osiągnięć autora należą:

1. Opracowanie nowatorskiego algorytmu *Multimodal Variational DeepMDP*, który znacząco poprawia możliwości generalizacji agenta RL do nieznanych zadań montażowych oraz umożliwia jego szybką adaptację w przypadku niepowodzenia transferu. Efekt ten uzyskano dzięki zastosowaniu uogólnionego iloczynu funkcji gęstości prawdopodobieństwa (gPoE) do fuzji modalności oraz funkcji celu integrującej predykcję nagrody i dynamiki multimodalnego środowiska.

2. Przeprowadzenie systematycznej analizy wpływu komponentów MVDeepMDP na jego wydajność, w tym wpływu mechanizmu fuzji modalności oraz strategii przetwarzania danych fizycznych. Wyniki wykazały, że gPoE oraz osobne enkodery dla każdej modalności znacząco poprawiają jakość reprezentacji i skuteczność agenta.

3. Zaprojektowanie i implementacja platformy programistycznej umożliwiającej efektywne trenowanie agenta RL sterującego rzeczywistym robotem przemysłowym. Architektura platformy wspiera asynchroniczne zbieranie danych względem procesu aktualizacji parametrów modelu, co zwiększa jej skalowalność i elastyczność.

4. Empiryczne wykazanie, że agent RL może z powodzeniem operować w zastosowaniach przemysłowych wymagających wysokiej precyzji i niezawodności, takich jak montaż elementów elektronicznych o zróżnicowanej geometrii i przeznaczeniu.

Podsumowując, przedstawiona praca potwierdza, że głębokie uczenie ze wzmocnieniem ma realny potencjał do zastosowania w robotyce przemysłowej. Opracowana metoda MVDeepMDP stanowi krok

w stronę autonomicznych systemów zdolnych do działania w dynamicznych i niejednorodnych środowiskach produkcyjnych. Mimo uzyskanych obiecujących wyników dalsze badania w rzeczywistych warunkach przemysłowych są niezbędne, aby w pełni ocenić praktyczne możliwości wdrożenia oraz wyznaczyć kierunki dalszego rozwoju i adaptacji metody do innych gałęzi przemysłu.

Bibliografia

[1] Grzegorz Bartyzel. Multimodal Variational DeepMDP: An Efficient Approach for Industrial Assembly in High-Mix, Low-Volume Production. *IEEE Robotics and Automation Letters*, 9(12):11297–11304, 2024.

[2] Carles Gelada and Saurabh Kumar and Jacob Buckman and Ofir Nachum and Marc G. Bellemare. DeepMDP: Learning Continuous Latent Space Models for Representation Learning. In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pages 2170–2179. PMLR, 09–15 Jun 2019.

[3] Scott Fujimoto, Herke van Hoof, and David Meger. Addressing Function Approximation Error in Actor-Critic Methods. In Jennifer G. Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pages 1582–1591. PMLR, 2018.

[4] Grzegorz Bartyzel and Wojciech Półchłopek and Dominik Rzepka. Reinforcement Learning With Stereo-View Observation for Robust Electronic Component Robotic Insertion. *Journal of Intelligent & Robotic Systems*, 109(3):57, Oct 2023.

[5] Tuomas Haarnoja, Aurick Zhou, Kristian Hartikainen, George Tucker, Sehoon Ha, Jie Tan, Vikash Kumar, Henry Zhu, Abhishek Gupta, Pieter Abbeel, and Sergey Levine. Soft Actor-Critic Algorithms and Applications, 2018.

[6] Dan Horgan, John Quan, David Budden, Gabriel Barth-Maron, Matteo Hessel, Hado van Hasselt, and David Silver. Distributed Prioritized Experience Replay. In *6th International Conference on Learning Representations*, 2018.

[7] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal Policy Optimization Algorithms. *CoRR*, abs/1707.06347, 2017.