



# Nawigowanie podwodnym robotem autonomicznym przy użyciu technik głębokiego uczenia

Piotr Zieliński

Promotor: dr hab. inż. Urszula Markowska-Kaczmar

Katedra Inteligencji Obliczeniowej  
Politechnika Wrocławskiego



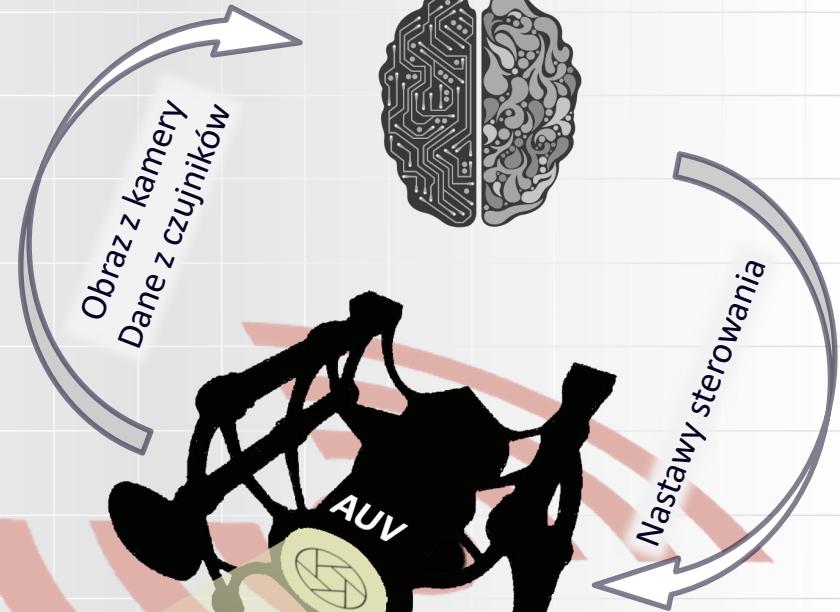
Uczenie cech wizualnych  
Głębokie sieci neuronowe  
Uczenie ze wzmocnieniem



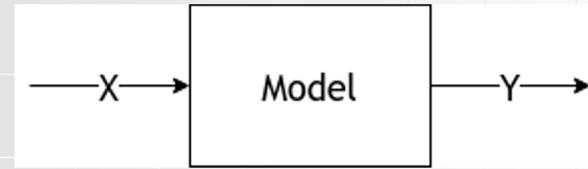
Punkt docelowy  $P^*$

NAWIGOWANIE

Punkt startowy  $P_0$

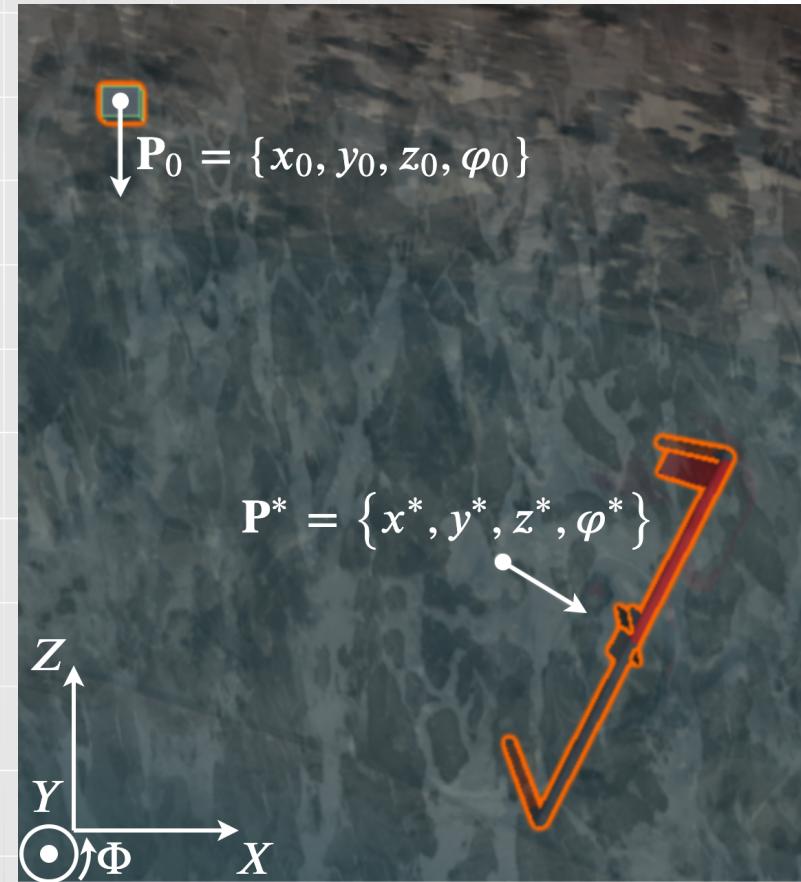


# Problem badawczy



$X$  – wektor danych wejściowych

$Y$  – wektor nastaw



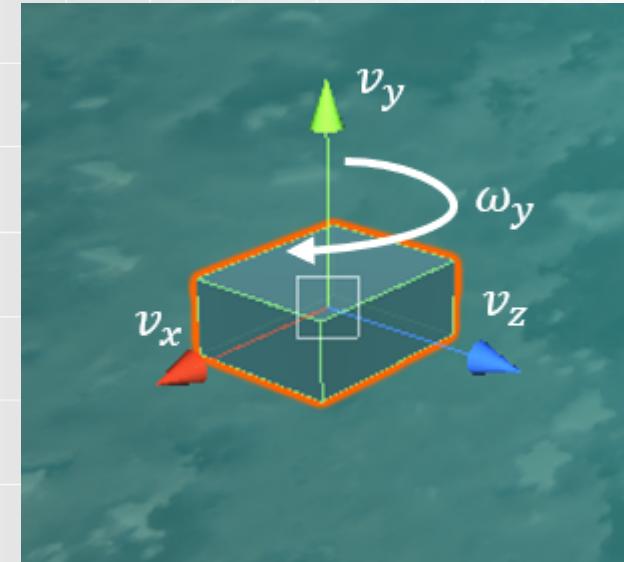
# Dane

Wejście:

- czujnik AHRS
  - przyspieszenie liniowe
  - prędkość kątowa
  - rotacja
- czujnik głębokości
- obraz z kamery

Wyjście:

- wektor nastaw sterowania
- $$Y = \{v_z, v_x, v_y, \omega_y\}$$
- $$v_z, v_x, v_y, \omega_y \in \mathbb{R},$$
- $$v_z, v_x, v_y, \omega_y \in [-1, 1]$$





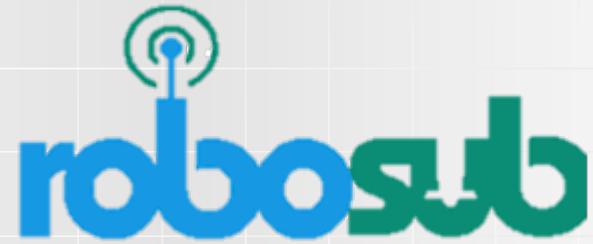
Politechnika  
Wrocławska

# Motywacja



**ROBOCIK**

Źródło: [FB KN Robocik](#)

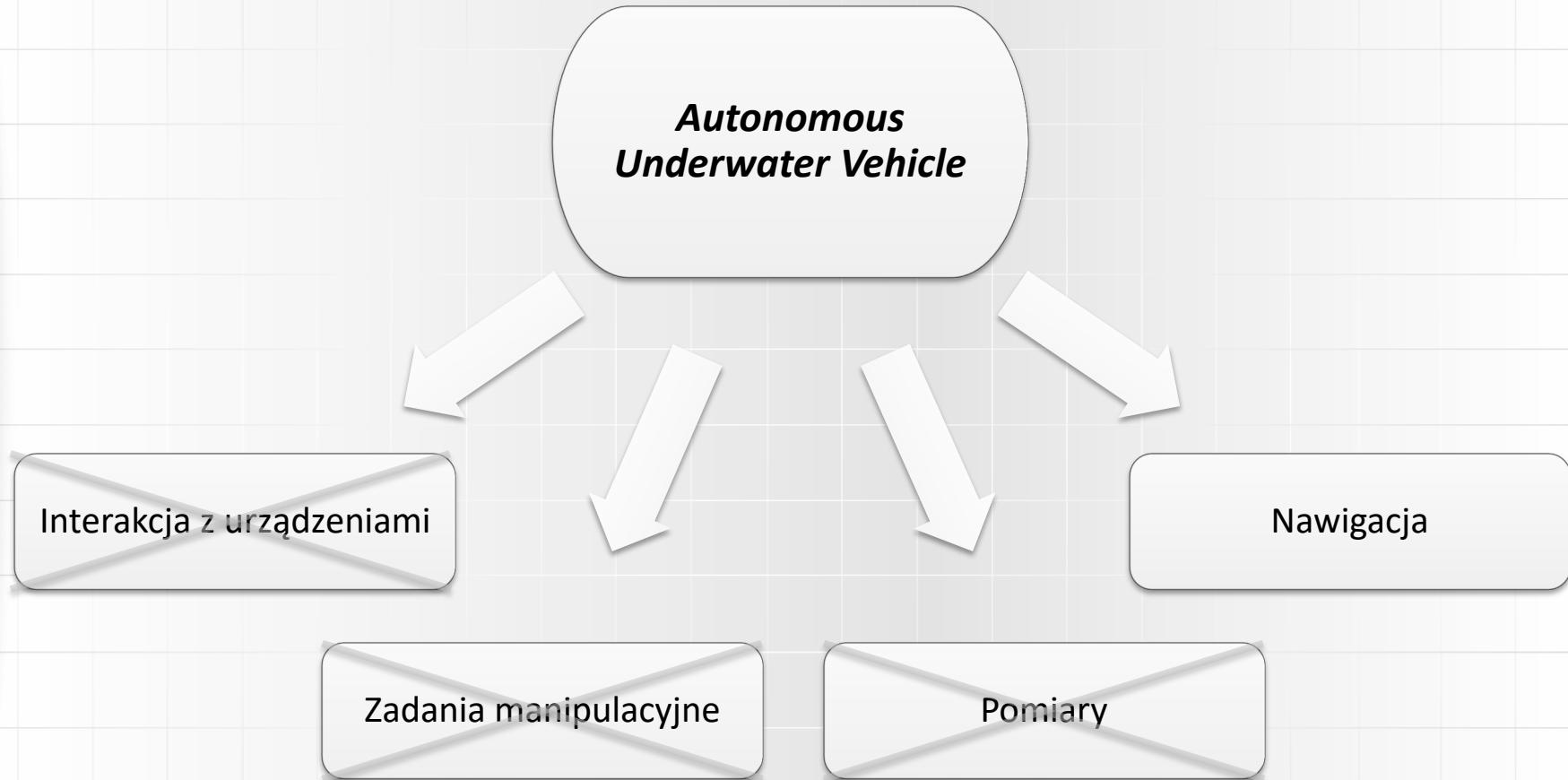


Źródło: [Robonation](#)



Źródło: [Robonation](#)

# Zadania AUV





# Rozwiązywane zadanie

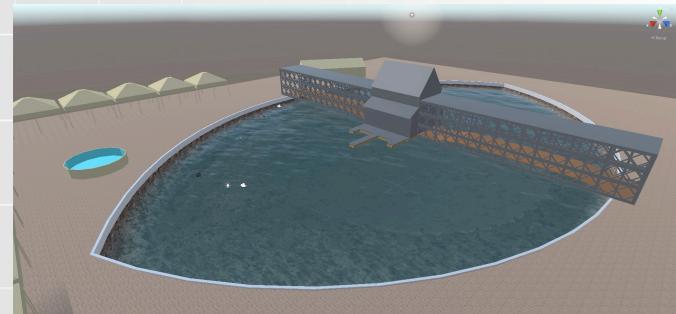
Metoda sterowania robotem podwodnym na podstawie danych z czujników pozwalająca na autonomiczne poruszanie się **pomiędzy** poszczególnymi zadaniami

# Założenia i ograniczenia

- środowisko trójwymiarowe
- konieczność przygotowania symulacji
- wykorzystanie zadań z zawodów RoboSub 2018
- rozwiązanie generyczne



Źródło: [Robonation](#)

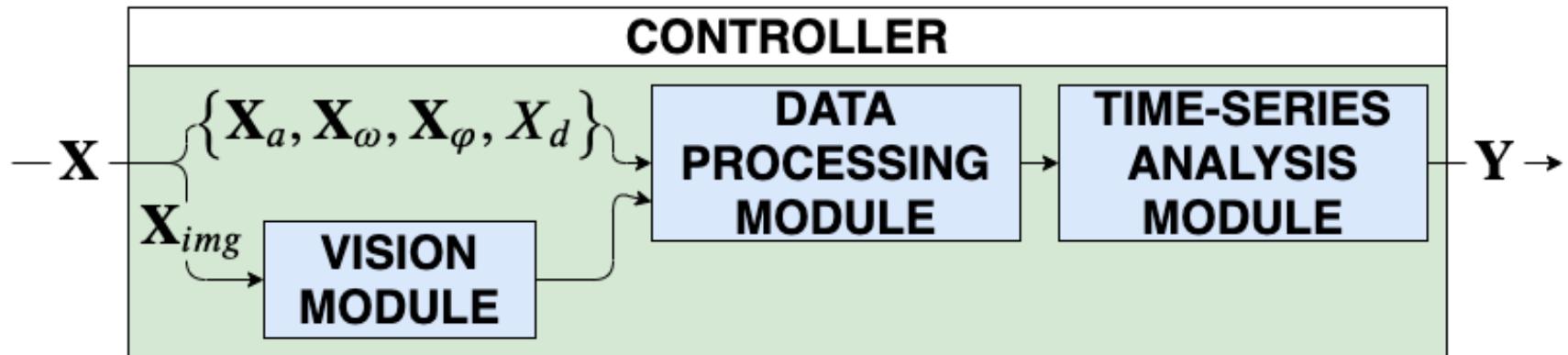




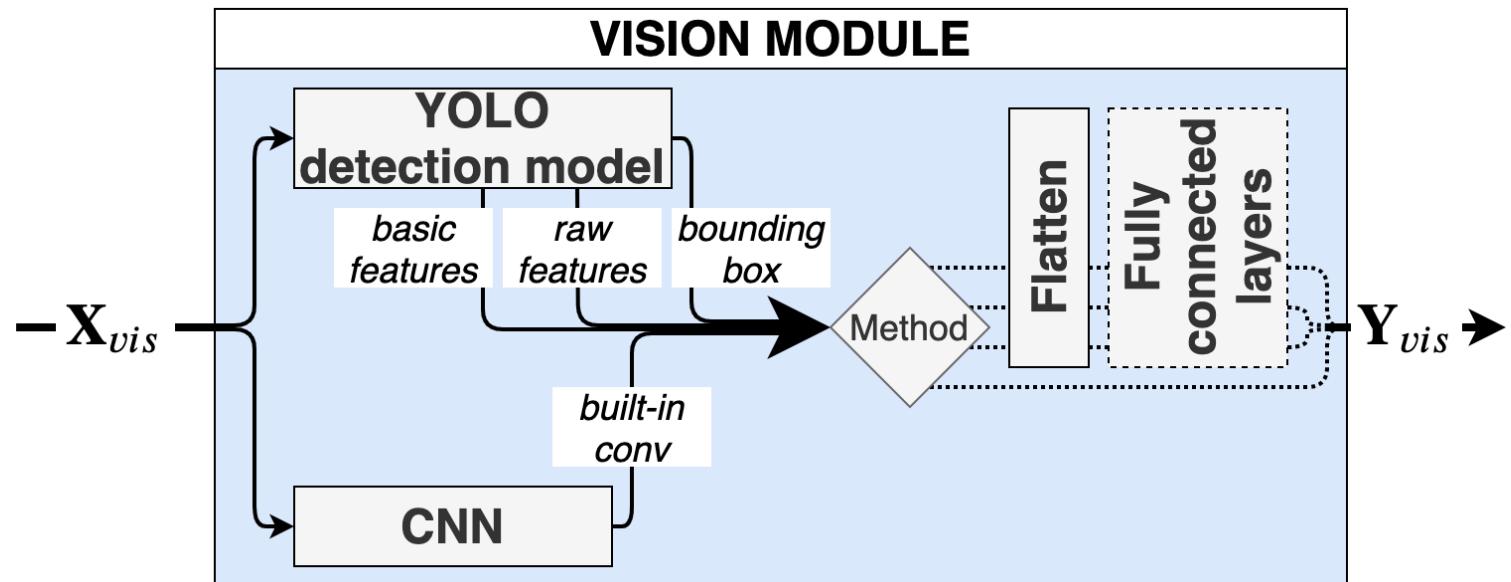
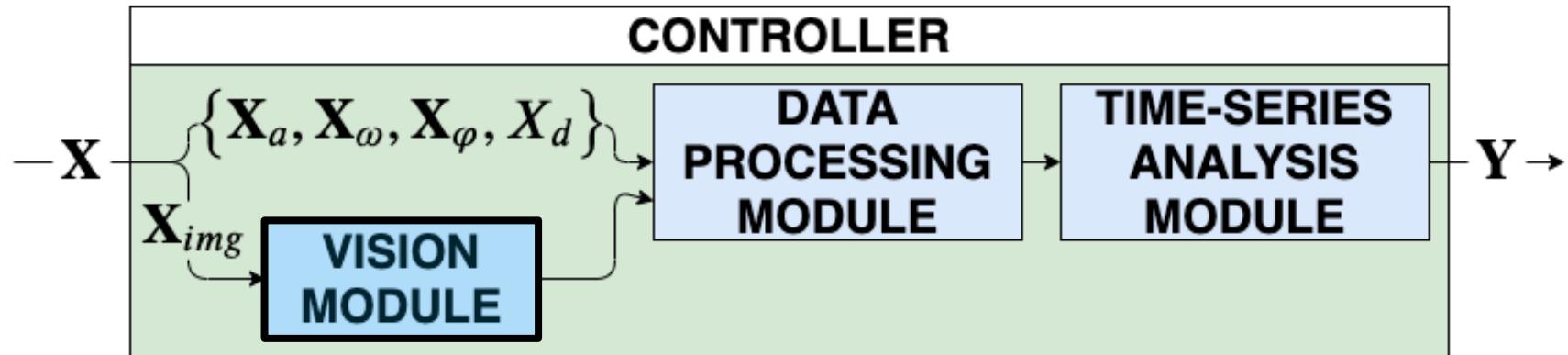
# Proponowane rozwiązanie



# Sterownik ruchu robota



# Moduł wizyjny

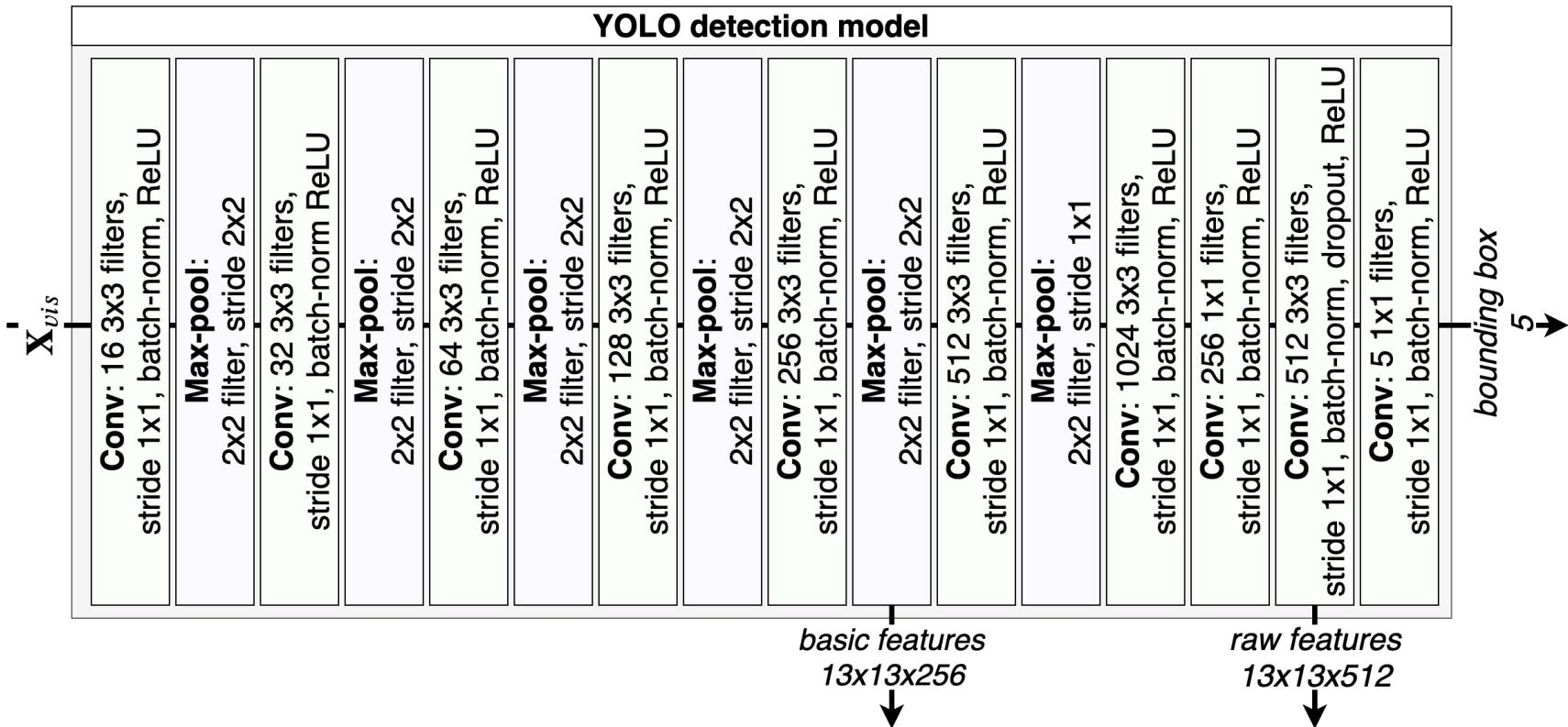


# Moduł wizyjny – model detekcji

- architektura wzorowana na TinyYOLOv3
- dodatkowe wyjścia map cech

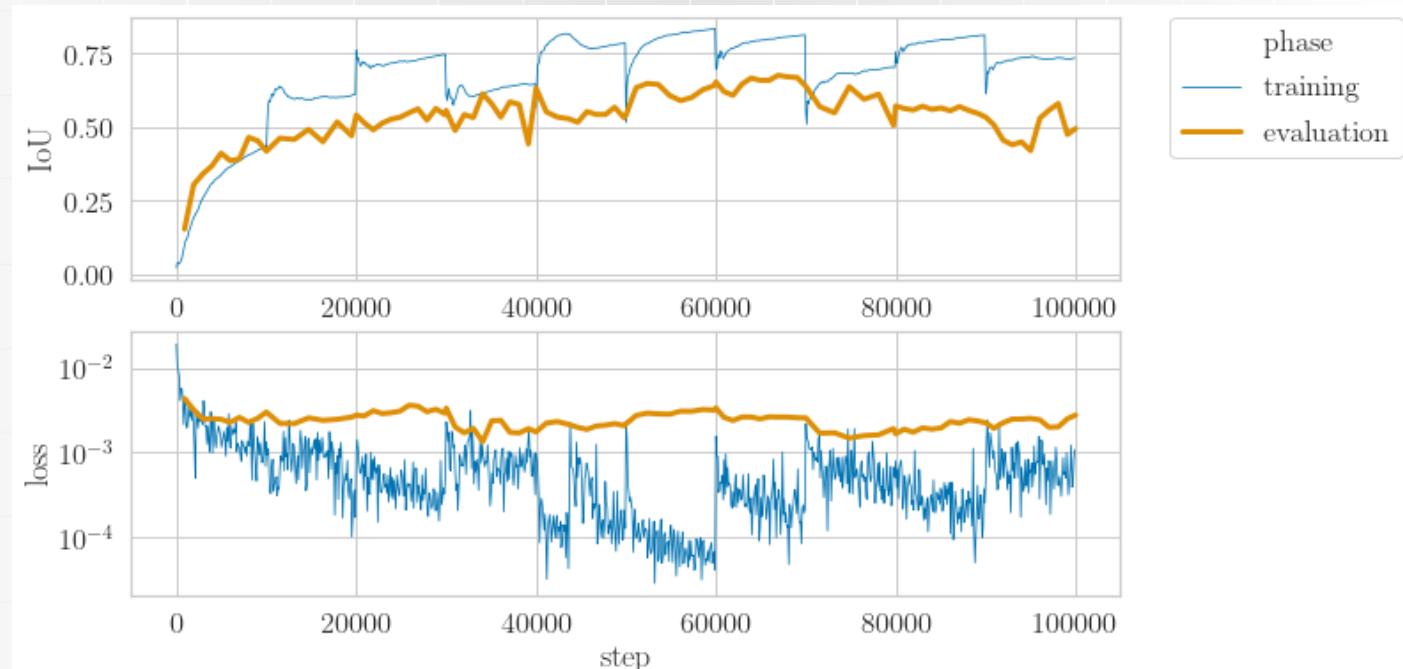


Źródło: Redmon et al. [5.]

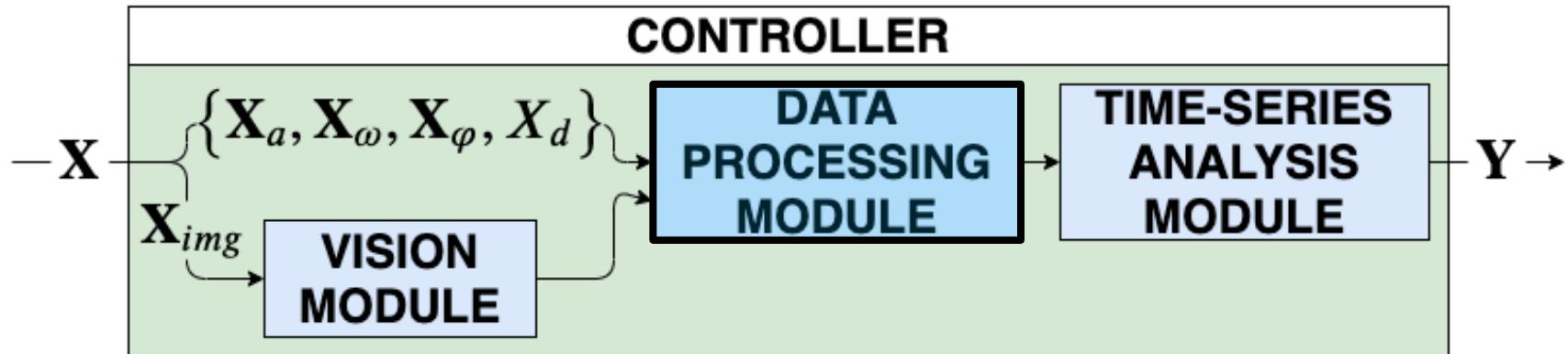


# Moduł wizyjny – model detekcji

- generowanie zbioru danych w środowisku testowym
- automatyczne anotowanie obrazów

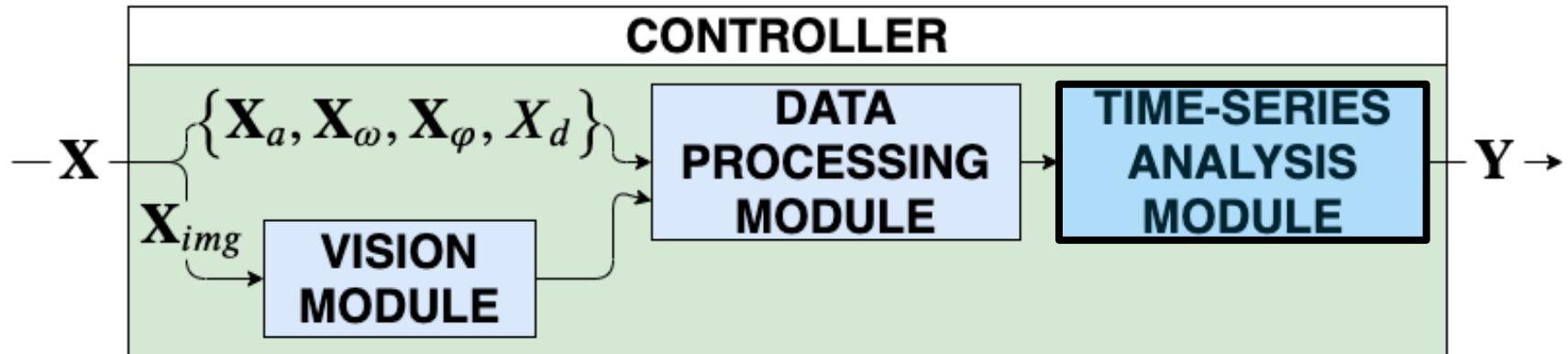


# Moduł przetwarzania danych



- konkatenacja danych z czujników oraz wyjścia modułu wizyjnego
- sekwencja trenowalnych warstw w pełni połączonych

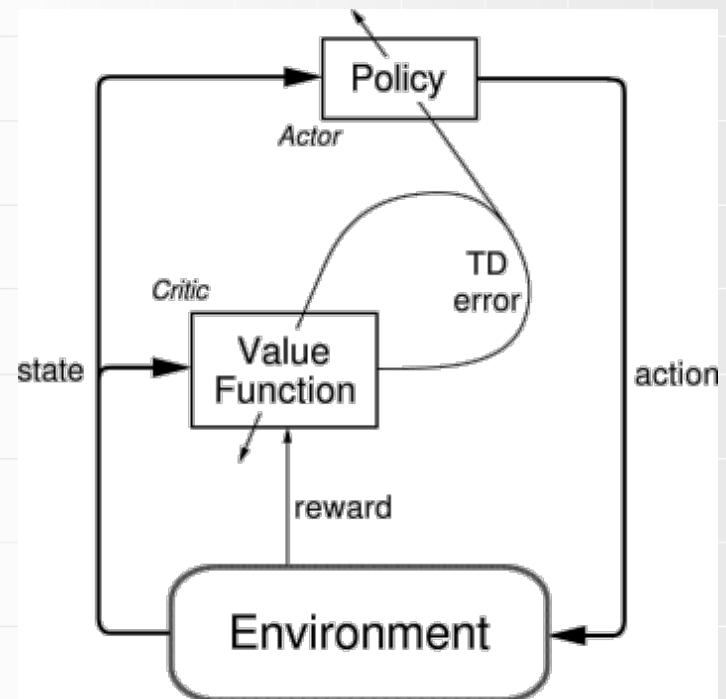
# Moduł analizy zależności czasowych



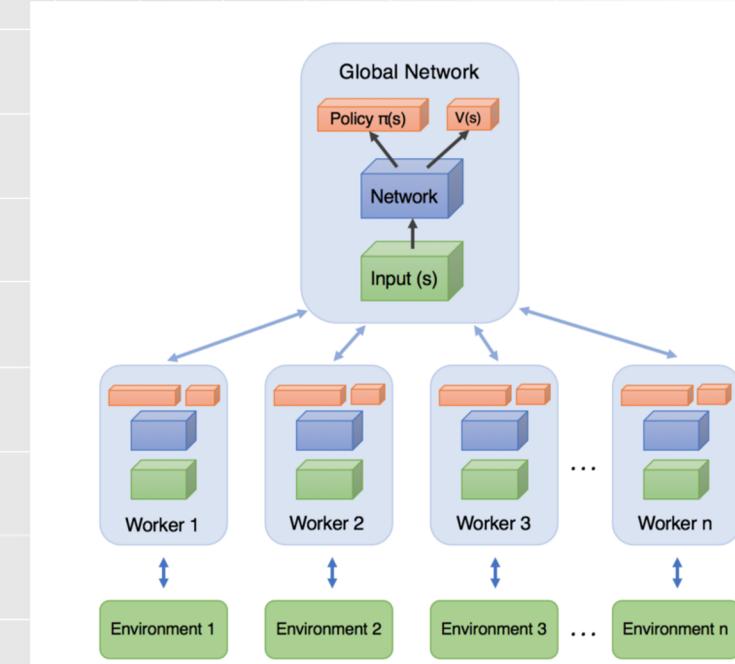
- zastosowanie warstwy rekurencyjnej LSTM
- uwzględnienie silnej i długoterminowej zależności czasowej

# Głębokie uczenie ze wzmacnieniem

- architektura aktor-krytyk
- metoda A2C
- *Proximal Policy Optimization* do aproksymacji strategii



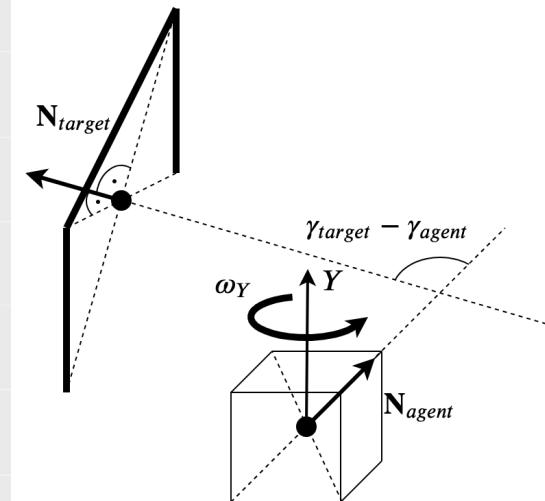
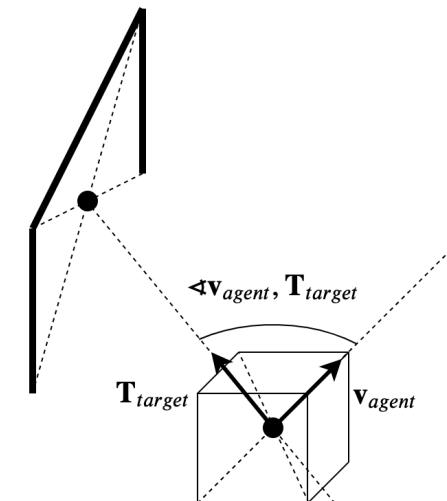
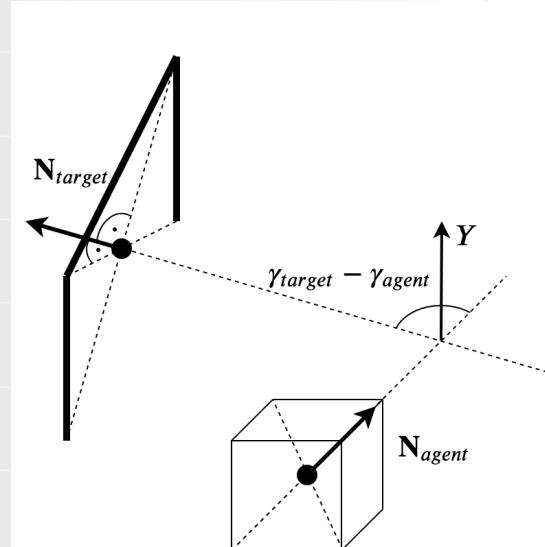
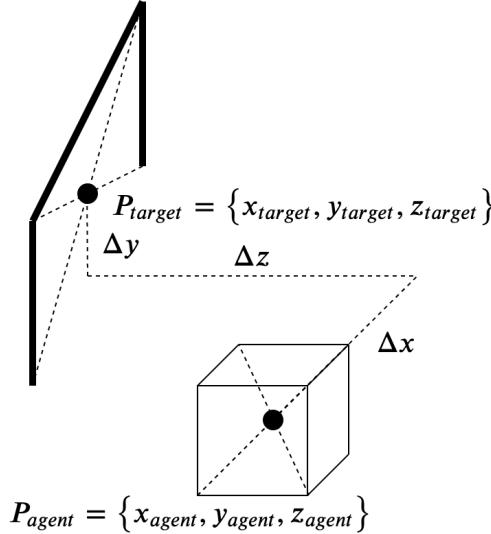
Źródło: Sutton, R. S., Barto, A. G. [7.]



Źródło: Juliani A.

# Funkcja nagrody

- 4 składowe:
  - pozycja
  - rotacja
  - prędkość liniowa
  - prędkość kątowa
- dyskontowanie w czasie
- ograniczona długość epizodu
- nagroda za osiągnięcie celu
- kary za błędy





# Badania



# Środowisko testowe

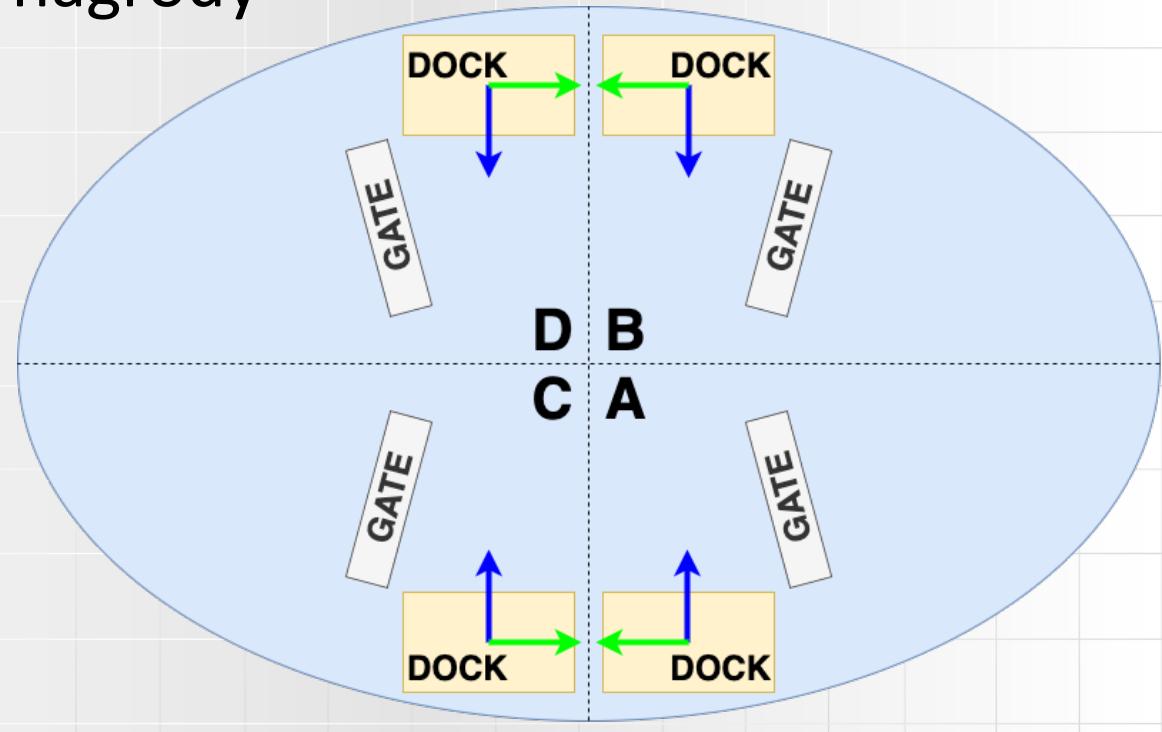
- RoboSub 2018 – zadanie 1

## Wymagania:

- komunikacja z kontrolerem
- symulowanie czujników
- obliczanie funkcji nagrody



Źródło: Robonation

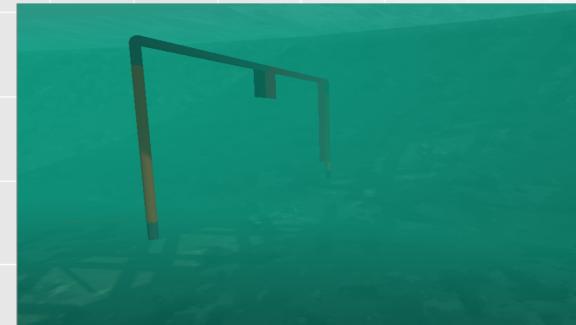


# Środowisko testowe

- możliwość losowania części basenu
- zmienne warunki (kolor i przezroczystość wody)
- wykorzystanie Unity-ML



Źródło: [Unity.com](https://www.unity.com)



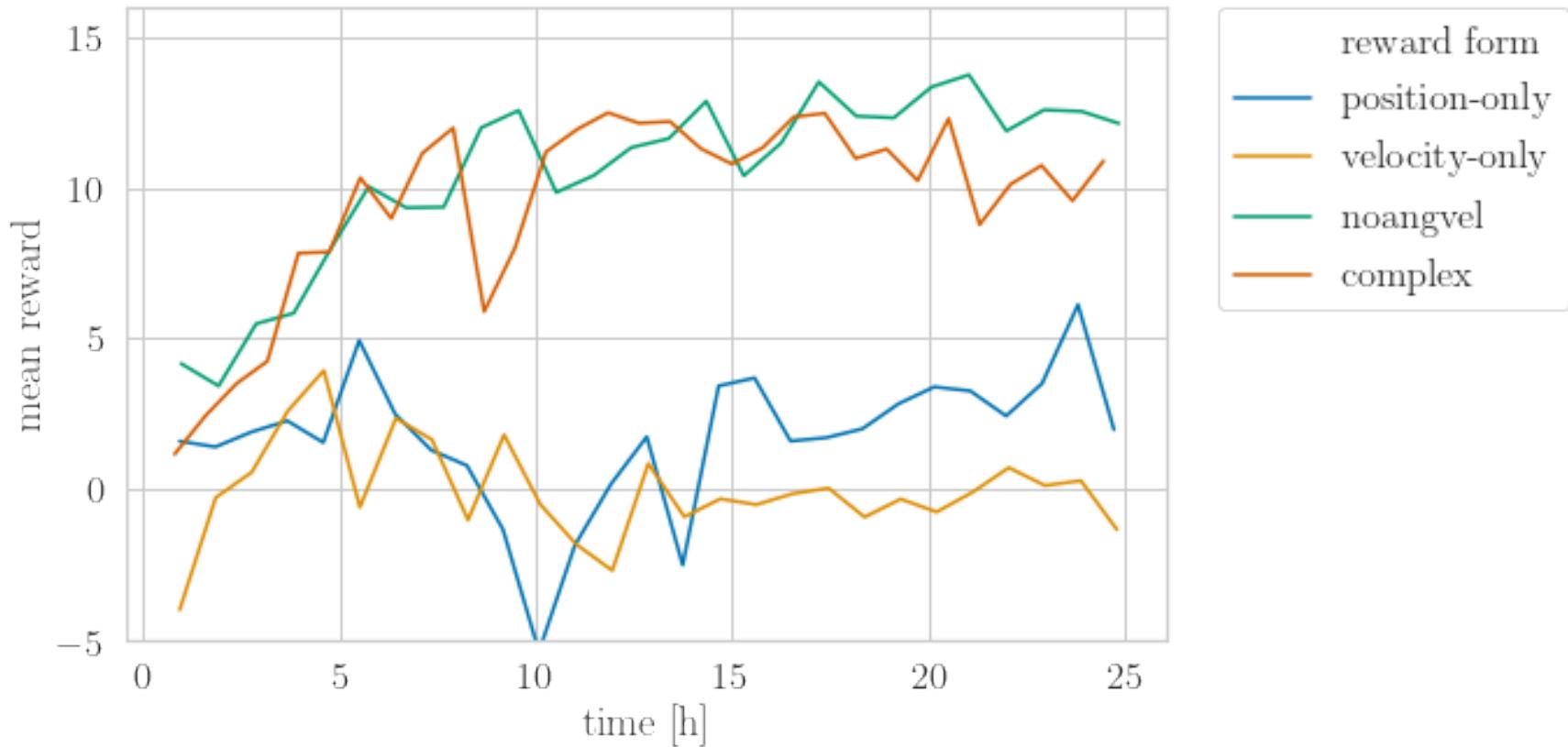
Źródło: [Robonation](https://www.robonation.com)



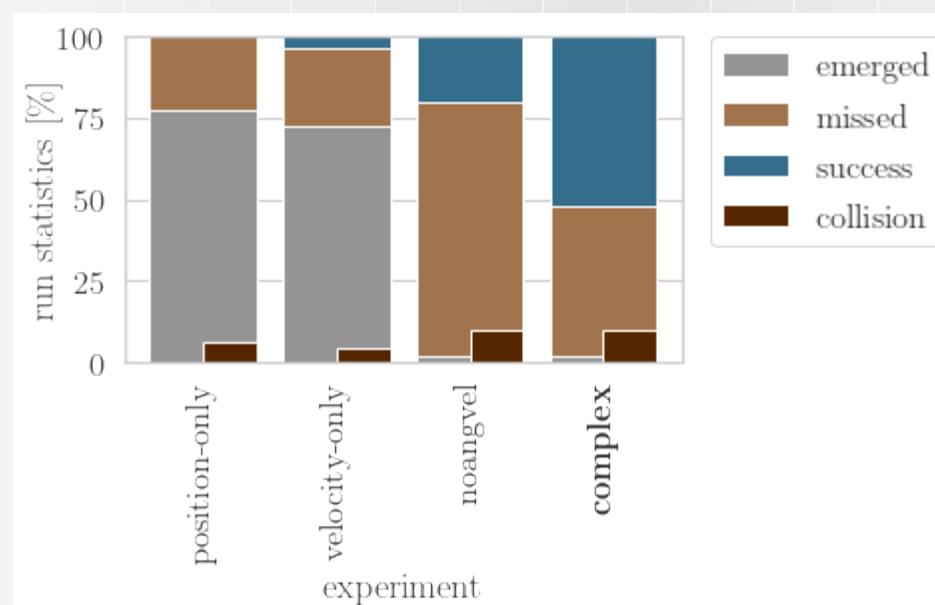
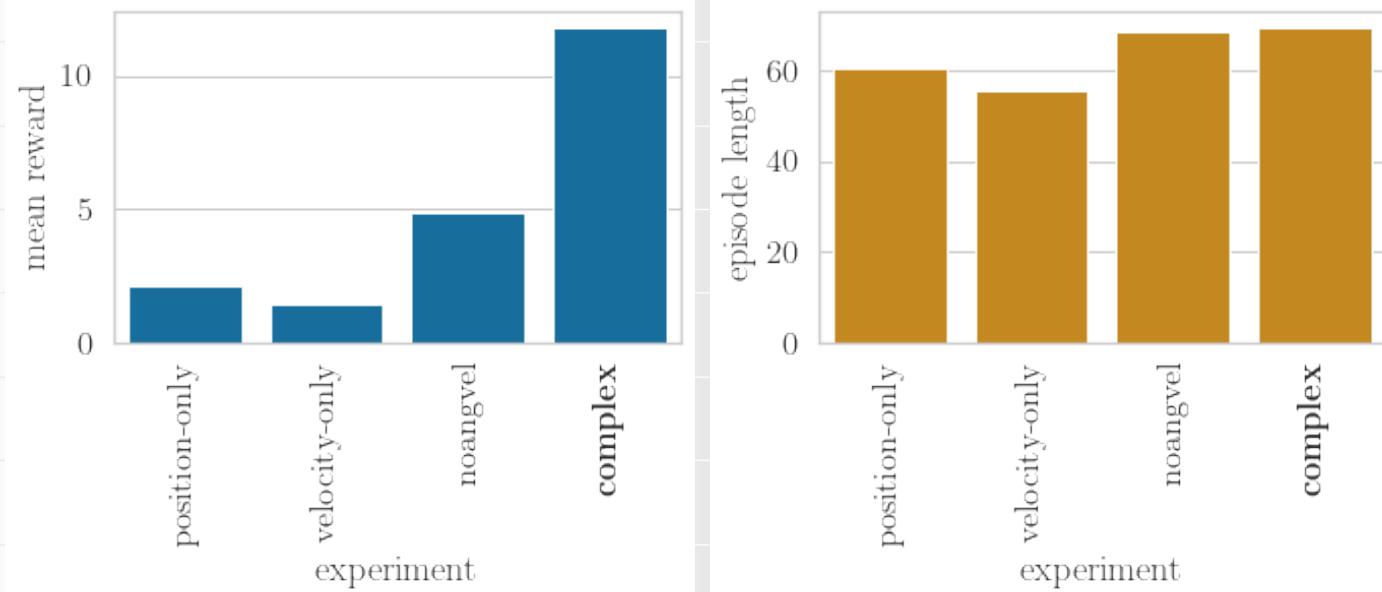
# Postać funkcji nagrody

- **cel:** wyznaczenie optymalnej postaci funkcji nagrody oraz zbadanie wpływu poszczególnych składowych na zdolność do wyuczenia

# Postać funkcji nagrody



# Postać funkcji nagrody





# Postać funkcji nagrody

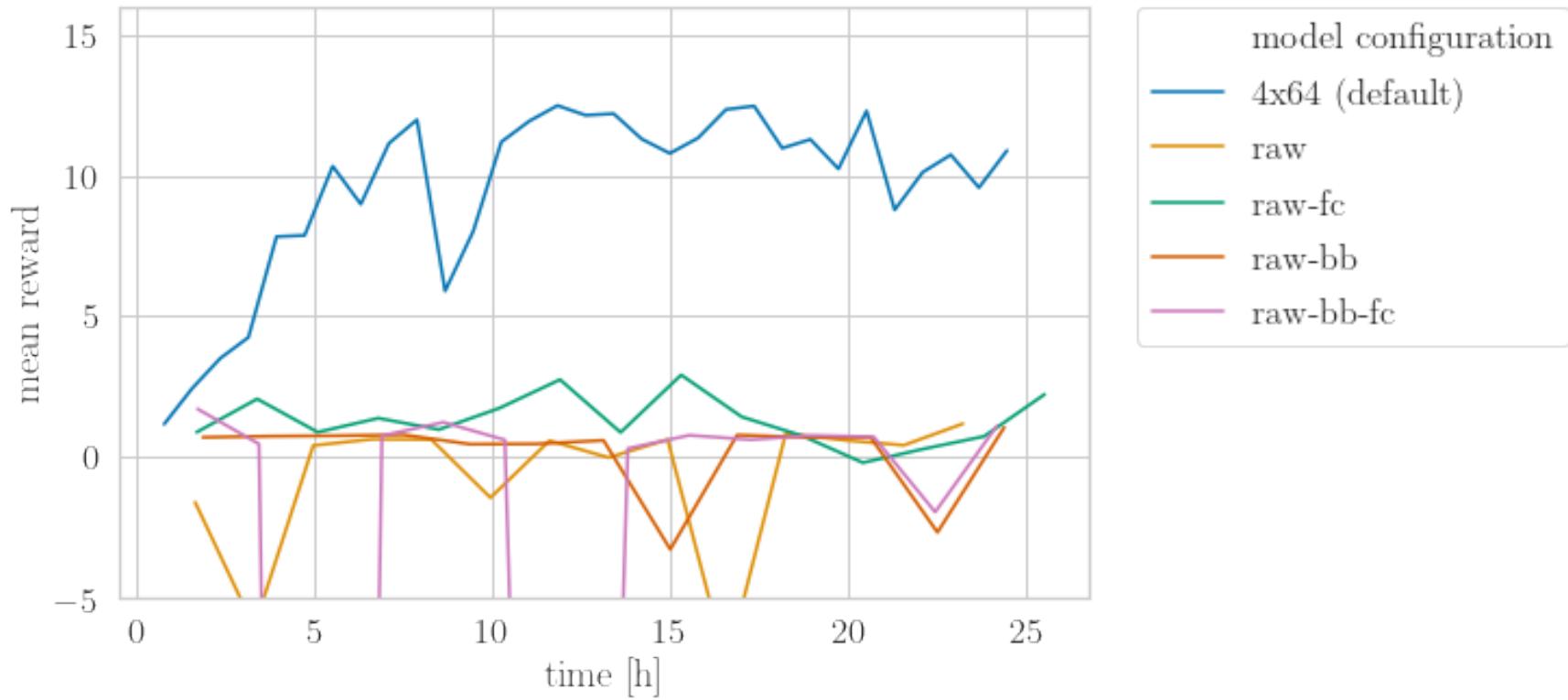
- **cel:** wyznaczenie optymalnej postaci funkcji nagrody oraz zbadanie wpływu poszczególnych składowych na zdolność do wyuczenia
- **wnioski:**
  - uproszczone funkcje nagrody nie pozwalają na skuteczne wytrenowanie modelu
  - uwzględnienie czterech składowych funkcji nagrody daje najlepsze wyniki uczenia i skuteczność działania



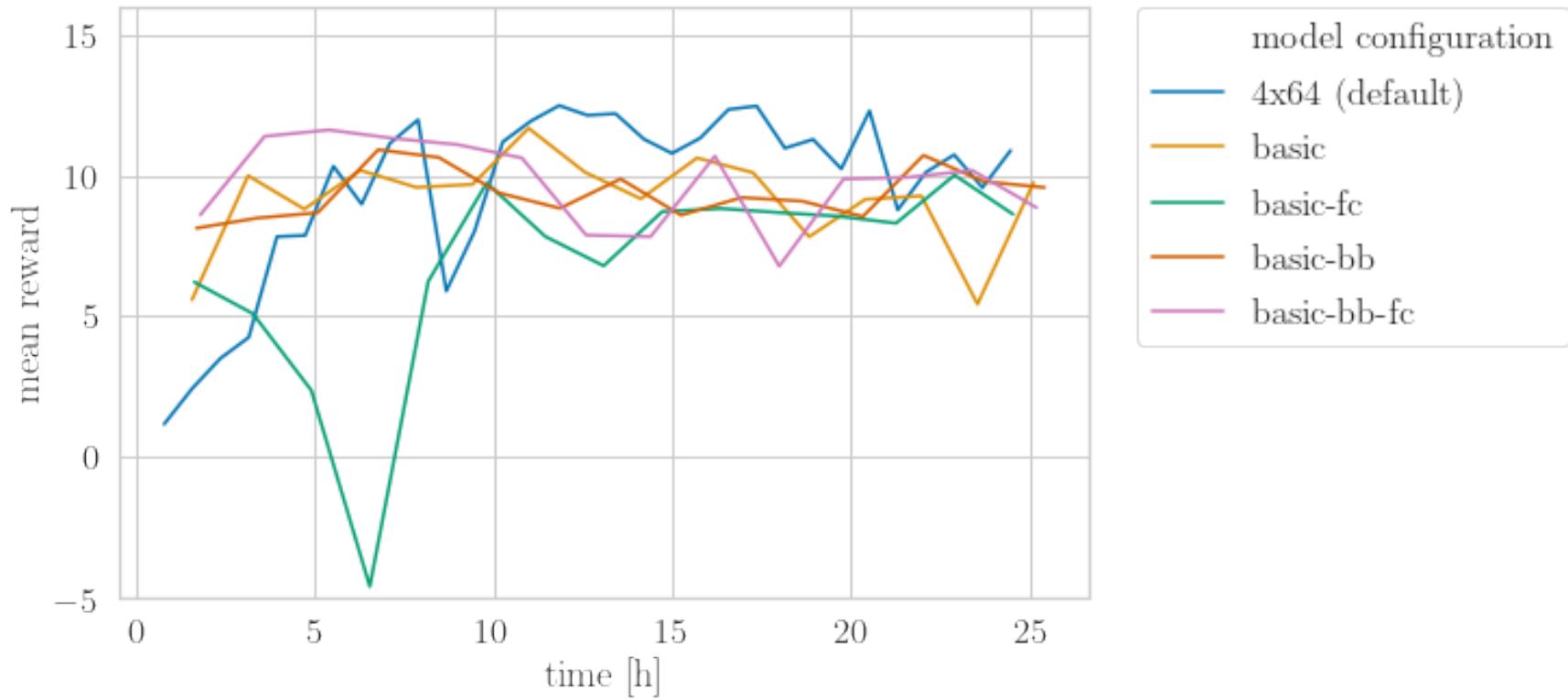
# Konfiguracja modelu

- **cel:** sprawdzenie skuteczności podejścia opartego na wytrenowanym modelu YOLO oraz wpływu konfiguracji modułu analizy obrazu na proces trenowania i osiągane wyniki

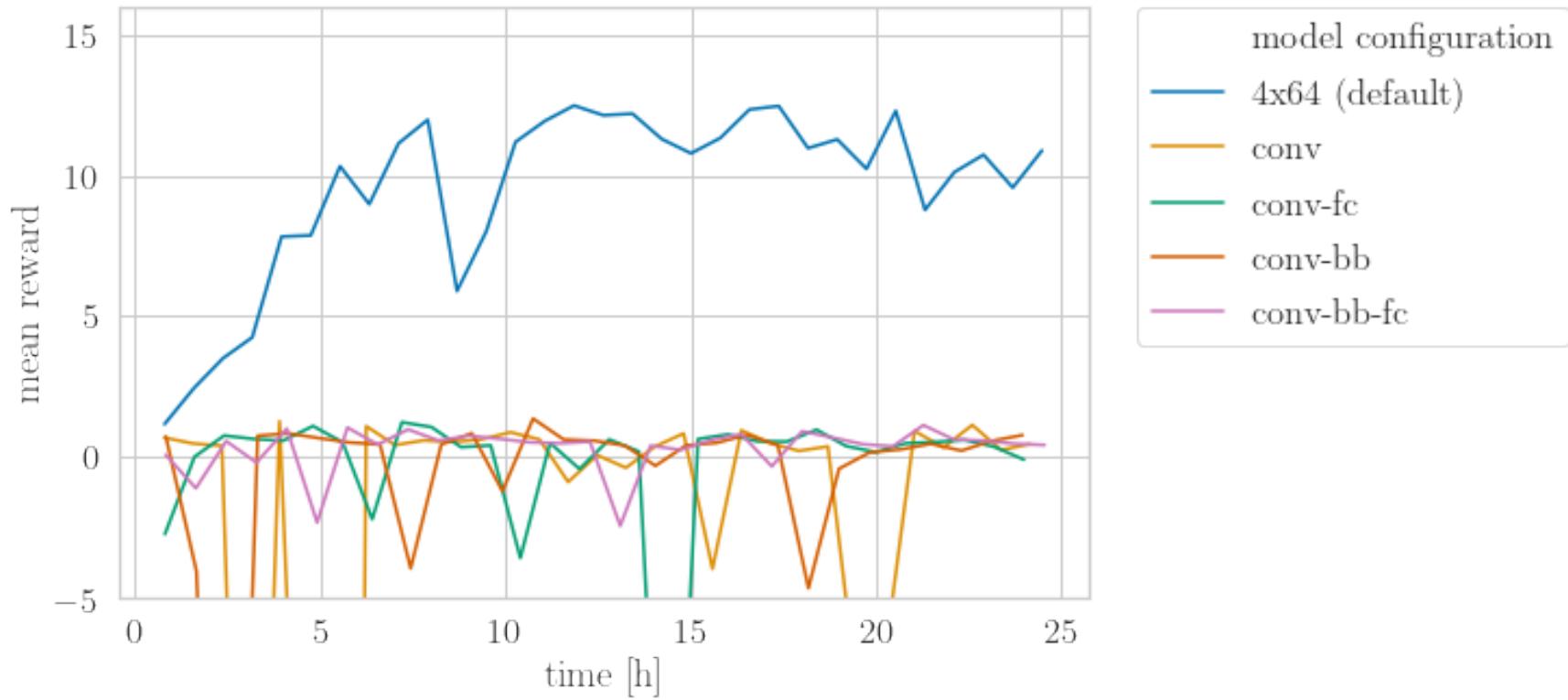
# Konfiguracja modelu – cechy surowe



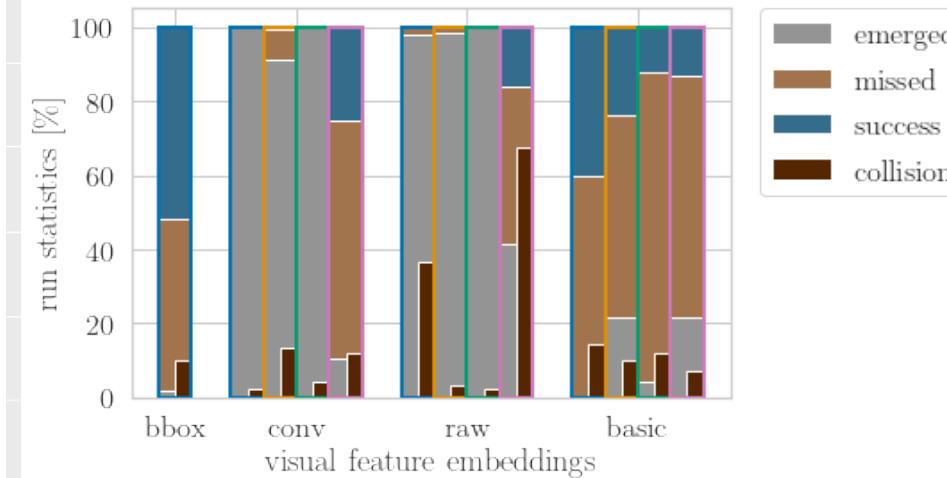
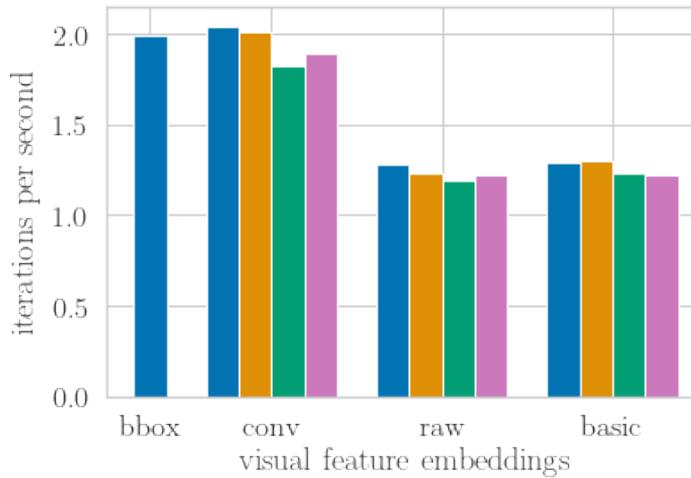
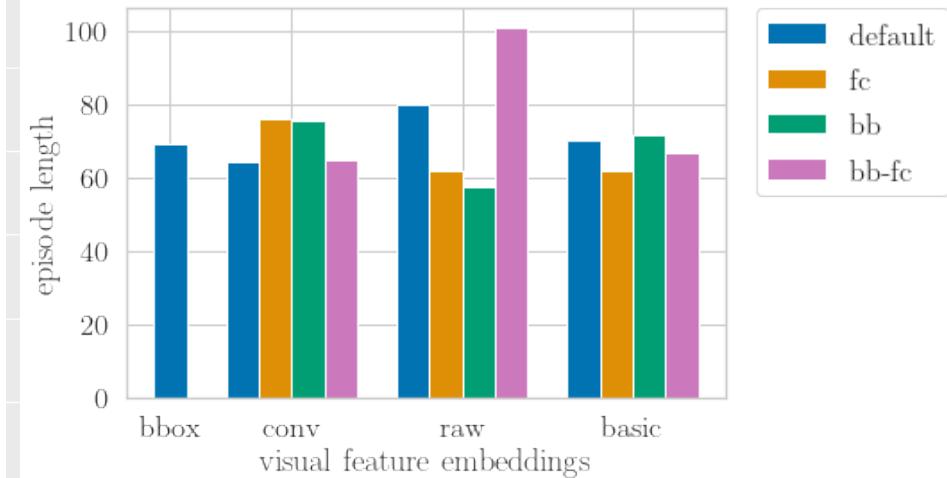
# Konfiguracja modelu – cechy proste



# Konfiguracja modelu – sieć CNN



# Konfiguracja modelu





# Postać funkcji nagrody

- **cel:** wyznaczenie optymalnej postaci funkcji nagrody oraz zbadanie wpływu poszczególnych składowych na zdolność do wyuczenia
- **wnioski:**
  - model wykorzystujący predykcje ramek okalających osiąga najlepsze rezultaty
  - spośród modeli opartych na ekstrakcji cech wizualnych jedynie cechy „proste” pozwalają na poprawne działanie
  - wykorzystanie cech wizualnych wyraźnie spowalnia działanie modelu
  - dodatkowe informacje nieznacznie poprawiają skuteczność



# Podsumowanie





# Dziękuję za uwagę!

Nawigowanie podwodnym robotem autonomicznym przy użyciu technik głębokiego uczenia

Piotr Zieliński

Promotor: dr hab. inż. Urszula Markowska-Kaczmar

Katedra Inteligencji Obliczeniowej  
Politechnika Wrocławskiego



# Literatura

1. Abadi, M. et al., *TensorFlow: Large-scale machine learning on heterogeneous systems*, tensorflow.org 2015
2. Hill, A. et al., *Stable Baselines*, GitHub 2018
3. Juliani, A. et al., *Unity: A general platform for intelligent agents*, arXiv 2018
4. Mnih, V. et al., *Asynchronous Methods for Deep Reinforcement Learning*, arXiv 2016
5. Redmon, J. et al., *YOLOv3: An Incremental Improvement*, arXiv 2018
6. Schulman, J. et al., *Proximal Policy Optimization Algorithms*, arXiv 2017
7. Sutton, R. S., Barto, A. G., *Reinforcement Learning: An Introduction*, The MIT Press 2018