



## KARTA OPISU PRZEDMIOTU - SYLABUS

Nazwa przedmiotu

Systemy wbudowane i przetwarzanie brzegowe

### Przedmiot

Kierunek studiów

Automatyka i Robotyka

Studia w zakresie (specjalność)

Roboty i Systemy Autonomiczne

Poziom studiów

drugiego stopnia

Forma studiów

stacjonarne

Rok/semestr

2/3

Profil studiów

ogólnoakademicki

Język oferowanego przedmiotu

polski

Wymagalność

obieralny

### Liczba godzin

Wykład

15

Laboratoria

30

Inne (np. online)

0

Ćwiczenia

0

Projekty/seminaria

0

### Liczba punktów

3

### Wykładowcy

Odpowiedzialny za przedmiot/wykładowca:

dr inż. Michał Fularz

email: [michal.fularz@put.poznan.pl](mailto:michal.fularz@put.poznan.pl)

Wydział Automatyki, Robotyki i Elektrotechniki

ul. Piotrowo 3A, 60-965 Poznań

Odpowiedzialny za przedmiot/wykładowca:

### Wymagania wstępne

Wiedza:

Student powinien posiadać ogólną wiedzę na poziomie studiów inżynierskich, w szczególności w zakresie matematyki: algebry, analizy matematycznej, logiki oraz probablistyki. Z zagadnień kierunkowych niezbędna jest wiedza na temat algorytmów przetwarzania obrazu, metod uczenia maszynowego i głębokiego, a także reprezentacji danych w systemach komputerowych.



### Umiejętności:

Student powinien potrafić efektywnie wykorzystać komputer i posiadać umiejętności implementacji algorytmów oraz zadań programistycznych, w szczególności, w obszarze metod przetwarzania obrazów oraz uczenia głębokiego. Ponadto niezbędne są również umiejętności z zakresu rozwiązywania problemów oraz nabywania wiedzy ze wskazanych źródeł.

### Cel przedmiotu

Celem kursu jest zdobycie wiedzy z zakresu architektury systemów wbudowanych przeznaczonych do przetwarzania brzegowego. W szczególności dotyczy to wykorzystania algorytmów przetwarzania obrazów oraz głębokiego uczenia na urządzeniach o ograniczonej mocy obliczeniowej w aplikacjach związanych z automatyką i robotyką.

### Przedmiotowe efekty uczenia się

#### Wiedza

Student, który ukończy kurs zdobędzie wiedzę z zakresu architektury, możliwości oraz zastosowań urządzeń wbudowanych do realizacji przetwarzania brzegowego w aplikacjach z dziedziny automatyki i robotyki. Będzie posiadał teoretyczną wiedzę na temat metod optymalizacji algorytmów, kwantyzacji wag modeli głębokiego uczenia, doboru modelu oraz konkretnych warstw. Student będzie w stanie właściwie dobrać typ i rodzaj urządzenia wbudowanego do rozwiązywanego problemu.

#### Umiejętności

Student po ukończeniu kursu zdobędzie praktyczne umiejętności związane z wykorzystaniem urządzeń wbudowanych do realizacji zadania przetwarzania brzegowych, a w szczególności w obszarze wnioskowania sieci neuronowych w aplikacjach z zakresu automatyki i robotyki. Ponadto będzie umiał zastosować w praktyce zagadnienia takie jak kwantyzacja wag, konwersja modelu, wdrożenie algorytmu na urządzenia brzegowe. Rozwinie również kompetencje związane z wyszukiwaniem informacji w specyfikacjach producentów oraz rozwiązywaniem złożonych problemów matematycznych, algorytmicznych oraz programistycznych.

#### Kompetencje społeczne

Student rozumie potrzebę i zna możliwości ciągłego dokształcania się – podnoszenia kompetencji zawodowych, osobistych i społecznych, potrafi inspirować i organizować proces uczenia się innych osób. Posiada świadomość odpowiedzialności za pracę własną oraz gotowość podporządkowania się zasadom pracy w zespole

### Metody weryfikacji efektów uczenia się i kryteria oceny

Efekty uczenia się przedstawione wyżej weryfikowane są w następujący sposób:

Przedstawione efekty kształcenia zostaną zweryfikowane w następujący sposób:

- Wykład - test zaliczeniowy przeprowadzony na platformie eKursy.
- Laboratoria - praktyczny projekt końcowy.

### Treści programowe



Ogólny plan kursu obejmuje realizację następujących zagadnień:

1. Złożoność obliczeniowa algorytmów - metody analizy i wyznaczania.
2. Akceleracja obliczeń - metody.
3. Architektury systemów wbudowanych.
4. Przetwarzania brzegowe - zalety i wady, wyzwania.
5. Metody rozwoju oprogramowania dla urządzeń wbudowanych - zdalna komunikacja.
6. Złożoność obliczeniowa metod głębokiego uczenia, dobór modelu oraz warstw. Kwantyzacja wag sieci neuronowej.
7. Wprowadzenie do TinyML - wykorzystania algorytmów uczenia głębokiego na mikrokontrolerach.

### Metody dydaktyczne

Wykłady: prezentacje multimedialne (slajdy) ilustrowane praktycznymi przykładami użycia oprogramowania narzędziowego oraz fragmentami kodu realizującymi wybrane treści przedstawione podczas wykładu.

Ćwiczenia laboratoryjne realizowane zgodnie z instrukcjami na komputerach oraz dedykowanych urządzeniach wbudowanych (m.in. mikrokontrolery, komputery jedno płytowe (SBC), dedykowane akceleratory sprzętowe, układy FPGA).

### Literatura

#### Podstawowa

1. NVIDIA Jetson Software Documentation
2. R. Szeliski, Computer Vision: Algorithms and Applications, Springer, 2010
3. P. Warden, D. Situnayake, TinyML. Machine Learning with TensorFlow Lite on Arduino and Ultra-Low-Power Microcontrollers, O'Reilly Media, 2019
4. Materiały uzupełniające opublikowane na platformie Moodle

#### Uzupełniająca

1. I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press, 2016
2. A. Koul, S. Ganju, M. Kasam, Practical Deep Learning for Cloud, Mobile, and Edge, O'Reilly Media, 2019



**Bilans nakładu pracy przeciętnego studenta**

	Godzin	ECTS
Łączny nakład pracy	75	3,0
Zajęcia wymagające bezpośredniego kontaktu z nauczycielem	45	2,0
Praca własna studenta (studia literaturowe, przygotowanie do zajęć laboratoryjnych/ćwiczeń, przygotowanie do kolokwίων/egzaminu, wykonanie projektu) <sup>1</sup>	30	1,0

<sup>1</sup> niepotrzebne skreślić lub dopisać inne czynności