



Karta przedmiotu

Nazwa i kod przedmiotu	Programowanie urządzeń brzegowych i mobilnych , PG_00053376						
Kierunek studiów	Inżynieria biomedyczna, Inżynieria biomedyczna, Inżynieria biomedyczna						
Data rozpoczęcia studiów	luty 2021 r.		Rok akademicki realizacji przedmiotu		2021/2022		
Poziom kształcenia	II stopnia		Grupa zajęć		Grupa zajęć fakultatywnych Grupa zajęć powiązanych z prowadzonymi badaniami naukowymi w dziedzinie nauki związanej z kierunkiem - profil ogólnoakademicki		
Forma studiów	stacjonarne		Sposób realizacji		na uczelni		
Rok studiów	1		Język wykładowy		polski		
Semestr studiów	2		Liczba punktów ECTS		2.0		
Profil kształcenia	ogólnoakademicki		Forma zaliczenia		zaliczenie		
Jednostka prowadząca	Wydział Elektroniki -> Telekomunikacji i Informatyki -> Katedra Systemów Geoinformatycznych						
Imię i nazwisko wykładowcy (wykładowców)	Odpowiedzialny za przedmiot		dr inż. Przemysław Falkowski-Gilski				
	Prowadzący zajęcia z przedmiotu		dr inż. Przemysław Falkowski-Gilski				
Formy zajęć i metody nauczania	Forma zajęć	Wykład	Ćwiczenia	Laboratorium	Projekt	Seminarium	RAZEM
	Liczba godzin zajęć	15.0	0.0	15.0	0.0	0.0	30
	W tym liczba godzin zajęć na odległość: 0.0						
Aktywność studenta i liczba godzin pracy	Aktywność studenta	Udział w zajęciach dydaktycznych, objętych planem studiów		Udział w konsultacjach		Praca własna studenta	RAZEM
	Liczba godzin pracy studenta	30		2.0		18.0	50
Cel przedmiotu	Celem przedmiotu jest zapoznanie studentów z podstawowymi technologiami programowania urządzeń brzegowych i mobilnych. W ramach kursu omawiane są techniki gromadzenia i przetwarzania danych z użyciem wbudowanych czujników, modułów zewnętrznych oraz modułów komunikacji bezprzewodowej, a także praktyczne wykorzystanie modeli uczenia głębokiego.						

Efekty uczenia się przedmiotu	Efekt kierunkowy	Efekt z przedmiotu	Sposób weryfikacji i oceny efektu
	[K7_W04] zna i rozumie w zaawansowanym stopniu zasady, metody i techniki programowania oraz zasady tworzenia oprogramowania komputerów albo programowania urządzeń lub sterowników wykorzystujących mikroprocesory albo inne elementy lub układy programowalne, specyficznych dla kierunku studiów, a także organizację pracy systemów wykorzystujących komputery lub te urządzenia	Student potrafi wskazać kluczowe elementy architektury systemów komputerowych, w szczególności urządzeń brzegowych oraz mobilnych.	[SW1] Ocena wiedzy faktograficznej
	[K7_U09] potrafi dokonać krytycznej analizy sposobu funkcjonowania istniejących rozwiązań technicznych i ocenić te rozwiązania, a także wykorzystać zdobyte w środowisku zajmującym się zawodowo działalnością inżynierską doświadczenie związane z utrzymaniem zaawansowanych urządzeń, obiektów i systemów technicznych typowych dla kierunku studiów	Student potrafi wykorzystać odpowiednie narzędzia i języki programowania w celu analizy danego zagadnienia.	[SU2] Ocena umiejętności analizy informacji
	[K7_K02] jest gotów do krytycznej oceny odbieranych treści, uznawania znaczenia wiedzy w rozwiązywaniu problemów poznawczych i praktycznych	Student potrafi dobrać odpowiednie metody, narzędzia oraz warstwę sprzętową i programową, w zależności od specyfiki analizowanego zagadnienia.	[SK5] Ocena umiejętności rozwiązywania problemów występujących w praktyce
	[K7_U04] potrafi wykorzystywać posiadaną wiedzę z zakresu metod i technik programowania oraz dobrać i zastosować właściwe metody i narzędzia programistyczne w tworzeniu oprogramowania komputerów albo programowania urządzeń lub sterowników wykorzystujących mikroprocesory albo elementy lub układy programowalne, charakterystycznych dla danego kierunku studiów, dokonując oceny i krytycznej analizy wykonanego oprogramowania, a także syntezy i twórczej interpretacji prezentowanych za jego pomocą informacji	Student potrafi właściwie zaprojektować i zaimplementować warstwę programową aplikacji na urządzenia brzegowe i mobilne.	[SU4] Ocena umiejętności korzystania z metod i narzędzi
Treści przedmiotu	1. Wprowadzenie do programowania na platformach brzegowych i mobilnych.  2. Konfiguracja środowiska programistycznego, bibliotek wewnętrznych i zewnętrznych.  3. Projektowanie i implementacja interfejsów graficznych użytkownika.  4. Gromadzenie i przetwarzanie danych z użyciem czujników wbudowanych, modułów zewnętrznych, modułów komunikacji bezprzewodowej.  5. Analiza i praktyczne wykorzystanie modeli uczenia głębokiego i środowisk brzegowych.		
Wymagania wstępne i dodatkowe	Podstawowa znajomość języków programowania Java, C/C++ oraz zagadnień z zakresu techniki programowania obiektowego.		
Sposoby i kryteria oceniania osiąganych efektów uczenia się	Sposób oceniania (składowe)	Próg zaliczeniowy	Składowa oceny końcowej
	Laboratorium	50.0%	50.0%
	Wykład	50.0%	50.0%

Zalecana lista lektur	Podstawowa lista lektur	<p>Murphy M., The Busy Coders Guide to Advanced Android Development, CommonsWare, 2011.</p> <p>Darwin I. F., Android Cookbook: Problems and Solutions for Android Development, ORiley Media, 2012.</p> <p>Płonkowski M., Android Studio. Tworzenie aplikacji mobilnych, Helion, 2017.</p>
	Uzupełniająca lista lektur	<p>Jeena Jacob I., Kolandapalayam Shanmugam S., Piramuthu S., Falkowski-Gilski P., Data Intelligence and Cognitive Informatics, Springer, 2021.</p> <p>Suresh A., Paiva S., Deep Learning and Edge Computing Solutions for High Performance Computing, Springer, 2021.</p> <p>Katangur A., Lin S. C., Wei J., Yang S., Zhang L. J., Edge Computing EDGE 2020, Springer, 2020.</p>
	Adresy eZasobów	
Przykładowe zagadnienia/ przykładowe pytania/ realizowane zadania	<ol style="list-style-type: none"> <li>1. Przygotowanie stanowiska laboratoryjnego, w tym konfiguracja emulatora i fizycznego urządzenia.</li> <li>2. Gromadzenie, przetwarzanie i prezentacja danych, dostęp do zasobów oraz pamięci wewnętrznej i zewnętrznej.</li> <li>3. Projekt i implementacja graficznego interfejsu użytkownika.</li> <li>4. Obsługa bibliotek uczenia maszynowego.</li> <li>5. Testowanie i optymalizacja wybranych modeli uczenia głębokiego.</li> </ol>	
Praktyki zawodowe w ramach przedmiotu	Nie dotyczy	



Karta przedmiotu

Nazwa i kod przedmiotu	Systemy internetowe i rozproszone , PG_00053373						
Kierunek studiów	Inżynieria biomedyczna, Inżynieria biomedyczna, Inżynieria biomedyczna						
Data rozpoczęcia studiów	luty 2021 r.		Rok akademicki realizacji przedmiotu		2021/2022		
Poziom kształcenia	II stopnia		Grupa zajęć		Grupa zajęć fakultatywnych Grupa zajęć powiązanych z prowadzonymi badaniami naukowymi w dziedzinie nauki związanej z kierunkiem - profil ogólnoakademicki		
Forma studiów	stacjonarne		Sposób realizacji		na uczelni		
Rok studiów	1		Język wykładowy		polski		
Semestr studiów	2		Liczba punktów ECTS		2.0		
Profil kształcenia	ogólnoakademicki		Forma zaliczenia		zaliczenie		
Jednostka prowadząca	Wydział Elektroniki -> Telekomunikacji i Informatyki -> Katedra Inżynierii Biomedycznej						
Imię i nazwisko wykładowcy (wykładowców)	Odpowiedzialny za przedmiot		dr hab. inż. Mariusz Kaczmarek				
	Prowadzący zajęcia z przedmiotu		dr hab. inż. Mariusz Kaczmarek mgr inż. Natalia Głowacka				
Formy zajęć i metody nauczania	Forma zajęć	Wykład	Ćwiczenia	Laboratorium	Projekt	Seminarium	RAZEM
	Liczba godzin zajęć	15.0	0.0	15.0	0.0	0.0	30
	W tym liczba godzin zajęć na odległość: 0.0						
Aktywność studenta i liczba godzin pracy	Aktywność studenta	Udział w zajęciach dydaktycznych, objętych planem studiów		Udział w konsultacjach		Praca własna studenta	RAZEM
	Liczba godzin pracy studenta	30		2.0		18.0	50
Cel przedmiotu	Celem przedmiotu jest zapoznanie studentów z technologiami internetowymi i sposobami przetwarzania rozproszonego w chmurze danych pochodzących z systemów wspomagania zdrowia i monitoringu stanu zdrowia. Pod uwagę brane są dane jednowymiarowe, dwuwymiarowe (obrazy) oraz sekwencje obrazów.						

Efekty uczenia się przedmiotu	Efekt kierunkowy	Efekt z przedmiotu	Sposób weryfikacji i oceny efektu
	[K7_W03] zna i rozumie w pogłębionym stopniu budowę i zasady działania komponentów i systemów związanych z kierunkiem studiów, w tym teorie, metody i złożone zależności między nimi oraz wybrane zagadnienia szczegółowe – właściwe dla programu kształcenia	Student rozumie zasady działania systemów wymiany treści pomiędzy urządzeniami klienta a serwerami zdalnymi.	[SW1] Ocena wiedzy faktograficznej
	[K7_W04] zna i rozumie w zaawansowanym stopniu zasady, metody i techniki programowania oraz zasady tworzenia oprogramowania komputerów albo programowania urządzeń lub sterowników wykorzystujących mikroprocesory albo inne elementy lub układy programowalne, specyficznych dla kierunku studiów, a także organizację pracy systemów wykorzystujących komputery lub te urządzenia	Potrafi zaprojektować i zaimplementować algorytm w wybranym języku programowania.	[SW3] Ocena wiedzy zawartej w opracowaniu tekstowym i projektowym
	[K7_U06] potrafi analizować działanie elementów, układów i systemów związanych z kierunkiem studiów oraz mierzyć ich parametry i badać charakterystyki techniczne, interpretować uzyskane wyniki i wyciągać wnioski	Potrafi przeanalizować diagram algorytmu oraz potrafi zinterpretować/zmodyfikować kod źródłowy aplikacji.	[SU4] Ocena umiejętności korzystania z metod i narzędzi [SU3] Ocena umiejętności wykorzystania wiedzy uzyskanej w ramach przedmiotu [SU1] Ocena realizacji zadania
	[K7_U03] potrafi zaprojektować, zgodnie z zadaną specyfikacją, oraz wykonać typowe dla kierunku studiów złożone urządzenie, obiekt, system lub zrealizować proces, używając odpowiednio dobranych metod, technik, narzędzi i materiałów, korzystając ze standardów i norm inżynierskich, stosując właściwe dla kierunków studiów technologie i wykorzystując doświadczenie zdobyte w środowisku zajmującym się zawodowo działalnością inżynierską	Potrafi zrealizować projekt informatyczny na podstawie zdefiniowanych wymagań funkcjonalnych i pozafunkcjonalnych.	[SU4] Ocena umiejętności korzystania z metod i narzędzi [SU3] Ocena umiejętności wykorzystania wiedzy uzyskanej w ramach przedmiotu [SU1] Ocena realizacji zadania

Treści przedmiotu	<p>Internetowe systemy rozproszone - definicje, pojęcia podstawowe, usługi bazowe w systemach rozproszonych</p> <p>Cechy systemów rozproszonych. Wymagania w zakresie systemów rozproszonych (m.in. komunikacja, synchronizacja i wymiana danych w systemach rozproszonych, modele spójności danych w SR)</p> <p>Architektury systemów przetwarzania rozproszonego (klient-serwer, systemy wielowęzłowe).</p> <p>Rekonstrukcja danych w systemach rozproszonych i transakcjach rozproszonych</p> <p>Współczesne języki programowania w realizacji systemów internetowych i rozproszonych (np. Python, biblioteki, JavaScript, itp.)</p> <p>Przetwarzanie rozproszone w systemach WWW (m.in. przetwarzanie wielowątkowe, równoległe, np. Python multiprocessing, multithreading)</p> <p>Przetwarzanie rozproszone w systemach WWW (m.in. REST, zdalna realizacja zadań uczenia maszynowego np. JavaScript vs. Python/TensorFlow)</p> <p>Rozwój aplikacji intensywnie obliczeniowych w środowiskach rozproszonych (np. klastry HPC; standard MPI, Apache Spark, itp.)</p> <p>Laboratorium</p> <p>Praktyczne wykorzystanie architektur systemów przetwarzania rozproszonego (klient-serwer, systemy wielowęzłowe, TCP vs. UDP, itp.)</p> <p>Praktyczna realizacja przetwarzania rozproszonego w systemach WWW z wykorzystaniem przetwarzania wielowątkowego, równoległego.</p> <p>Praktyczna realizacja przetwarzania rozproszonego w systemach WWW z wykorzystaniem technologii zdalnego wywoływania metod (np. REST)</p> <p>Praktyczna realizacja przetwarzania rozproszonego w realizacji zadań uczenia maszynowego (np. Python, TensorFlow, JavaScript)</p> <p>Praktyczna realizacja systemu internetowego w zakresie usług medycznych</p>		
Wymagania wstępne i dodatkowe	Znajomość podstaw programowania w dowolnym języku skrypcowym i obiektowym.		
Sposoby i kryteria oceniania osiągniętych efektów uczenia się	Sposób oceniania (składowe)	Próg zaliczeniowy	Składowa oceny końcowej
	Kolokwium	51.0%	40.0%
	Laboratorium	51.0%	60.0%
Zalecana lista lektur	Podstawowa lista lektur	<ol style="list-style-type: none"> <li>1. Distributed Systems: Principles and Paradigms 2nd Edition, . <a href="#">Andrew S. Tanenbaum</a>, <a href="#">Maarten van Steen</a>, ISBN: 978-1530281756</li> <li>2. INTERNET COMPUTING: PRINCIPLES OF DISTRIBUTED SYSTEMS AND EMERGING INTERNET-BASED TECHNOLOGIES, Ali Sunyaev, Springer, 2020, ISBN: 9783030349561</li> </ol>	
	Uzupełniająca lista lektur	<ol style="list-style-type: none"> <li>1. M. Ben-Ari, "Podstawy programowania współbieżnego i rozproszonego", WNT 2009.</li> <li>2. Dokumentacja języka Python (<a href="http://www.python.org">www.python.org</a>).</li> <li>3. Dokumentacja frameworka Django (<a href="http://www.djangoproject.com">http://www.djangoproject.com</a>).</li> <li>4. Dokumentacja biblioteki Wt (<a href="http://www.webtoolkit.eu/wt">http://www.webtoolkit.eu/wt</a>)</li> </ol>	
	Adresy eZasobów		

Przykładowe zagadnienia/ przykładowe pytania/ realizowane zadania	
Praktyki zawodowe w ramach przedmiotu	Nie dotyczy



Karta przedmiotu

Nazwa i kod przedmiotu	Uczenie głębokie, PG_00053375						
Kierunek studiów	Inżynieria biomedyczna, Inżynieria biomedyczna, Inżynieria biomedyczna						
Data rozpoczęcia studiów	luty 2021 r.		Rok akademicki realizacji przedmiotu		2021/2022		
Poziom kształcenia	II stopnia		Grupa zajęć		Grupa zajęć fakultatywnych Grupa zajęć powiązanych z prowadzonymi badaniami naukowymi w dziedzinie nauki związanej z kierunkiem - profil ogólnoakademicki		
Forma studiów	stacjonarne		Sposób realizacji		na uczelni		
Rok studiów	1		Język wykładowy		polski		
Semestr studiów	2		Liczba punktów ECTS		4.0		
Profil kształcenia	ogólnoakademicki		Forma zaliczenia		egzamin		
Jednostka prowadząca	Wydział Elektroniki -> Telekomunikacji i Informatyki -> Katedra Inżynierii Biomedycznej						
Imię i nazwisko wykładowcy (wykładowców)	Odpowiedzialny za przedmiot		dr hab. inż. Jacek Rumiński				
	Prowadzący zajęcia z przedmiotu		mgr inż. Natalia Głowacka mgr inż. Szymon Zaporowski dr inż. Adam Kurowski dr hab. inż. Jacek Rumiński				
Formy zajęć i metody nauczania	Forma zajęć	Wykład	Ćwiczenia	Laboratorium	Projekt	Seminarium	RAZEM
	Liczba godzin zajęć	30.0	0.0	15.0	0.0	0.0	45
	W tym liczba godzin zajęć na odległość: 0.0						
	Adres zajęć na odległość: <a href="https://enauczanie.pg.edu.pl/moodle/course/view.php?id=17105">https://enauczanie.pg.edu.pl/moodle/course/view.php?id=17105</a>						
Aktywność studenta i liczba godzin pracy	Aktywność studenta	Udział w zajęciach dydaktycznych, objętych planem studiów		Udział w konsultacjach		Praca własna studenta	RAZEM
	Liczba godzin pracy studenta	45		4.0		51.0	100
Cel przedmiotu	Celem przedmiotu jest przekazanie słuchaczom wiedzy z zakresu głębokich, sztucznych sieci neuronowych oraz wykształcenia umiejętności praktycznych w tym zakresie.						



Efekty uczenia się przedmiotu	Efekt kierunkowy	Efekt z przedmiotu	Sposób weryfikacji i oceny efektu
	[K7_U04] potrafi wykorzystywać posiadaną wiedzę z zakresu metod i technik programowania oraz dobrać i zastosować właściwe metody i narzędzia programistyczne w tworzeniu oprogramowania komputerów albo programowania urządzeń lub sterowników wykorzystujących mikroprocesory albo elementy lub układy programowalne, charakterystycznych dla danego kierunku studiów, dokonując oceny i krytycznej analizy wykonanego oprogramowania, a także syntezy i twórczej interpretacji prezentowanych za jego pomocą informacji	Efektom procesu uczenia jest zdobycie przez studenta umiejętności praktycznego zastosowania algorytmów uczenia głębokiego, w szczególności sieci spłotowych i rekurencyjnych w szczególności poprzez implementację oprogramowania modeli sieci, ich trening, testowanie i interpretowanie wyników.	[SU1] Ocena realizacji zadania
	[K7_W01] zna i rozumie w pogłębionym stopniu matematykę w zakresie niezbędnym do formułowania i rozwiązywania złożonych zagadnień związanych z kierunkiem studiów	Efektom procesu uczenia jest zdobycie przez studenta wiedzy w zakresie rozumienia definicji algorytmów uczenia głębokiego, w szczególności sieci spłotowych i rekurencyjnych powiązanych wzorów oraz problemów dotyczących procesu uczenia sieci głębokich.	[SW1] Ocena wiedzy faktograficznej
	[K7_U05] potrafi planować i przeprowadzać eksperymenty związane z kierunkiem studiów, w tym pomiary i symulacje komputerowe, interpretować uzyskane wyniki i wyciągać wnioski	Efektom procesu uczenia jest zdobycie przez studenta umiejętności praktycznego zastosowania algorytmów uczenia głębokiego, w szczególności sieci spłotowych i rekurencyjnych w szczególności poprzez projektowanie modeli sieci oraz planowanie eksperymentów związanych z treningiem i walidacją modeli.	[SU1] Ocena realizacji zadania
	[K7_W03] zna i rozumie w pogłębionym stopniu budowę i zasady działania komponentów i systemów związanych z kierunkiem studiów, w tym teorie, metody i złożone zależności między nimi oraz wybrane zagadnienia szczegółowe – właściwe dla programu kształcenia	Efektom procesu uczenia jest zdobycie przez studenta wiedzy w zakresie metod uczenia głębokiego, w szczególności w zakresie zadań klasyfikacji danych, detekcji obiektów i innych zadań związanych z kierunkiem studiów.	[SW1] Ocena wiedzy faktograficznej
Treści przedmiotu	Wprowadzenie do uczenia głębokiego Operacja spłotu i jej znaczenie Warstwy spłotowe i ich wersje Spłotowe sieci neuronowe (rodzaje warstw, uczenie) Klasyfikacja z wykorzystaniem sieci spłotowych Problemy z uczeniem sieci głębokich (przeuczenie, zanikający gradient, itp.) Metody przeciwdziałania problemom związanym z uczeniem sieci głębokich (regularyzacja, augmentacja, dropout, early stopping, itp.) Uczenie z przeniesieniem (transfer learning) Modele RNN Rozwój modeli RNN (m.in. LSTM itd.) Zastosowaniem modeli RNN w NLP Modele generacyjne Autoenkodery Zastosowanie modeli generacyjnych Uczenie ze wzmocnieniem Uczenie ze wzmocnieniem z wykorzystaniem modeli głębokich cz.1 Uczenie ze wzmocnieniem z wykorzystaniem modeli głębokich cz.2		
Wymagania wstępne i dodatkowe	Realizacja przedmiotów z semestru pierwszego.		
Sposoby i kryteria oceniania osiąganych efektów uczenia się	Sposób oceniania (składowe)	Próg zaliczeniowy	Składowa oceny końcowej
	Egzamin	50.0%	40.0%
	Laboratorium	50.0%	50.0%
	Zadania domowe	0.0%	10.0%
Zalecana lista lektur	Podstawowa lista lektur	Bengio Yoshua, Courville Aaron, Goodfellow Ian, Deep Learning, Systemy uczące się, PWN 2018  Andrew W. Trask, Zrozumieć głębokie uczenie, PWN, 2019	

	Uzupełniająca lista lektur	brak
	Adresy eZasobów	
Przykładowe zagadnienia/ przykładowe pytania/ realizowane zadania		
Praktyki zawodowe w ramach przedmiotu	Nie dotyczy	



Karta przedmiotu

Nazwa i kod przedmiotu	Wizja komputerowa, PG_00053374						
Kierunek studiów	Inżynieria biomedyczna, Inżynieria biomedyczna, Inżynieria biomedyczna						
Data rozpoczęcia studiów	luty 2021 r.		Rok akademicki realizacji przedmiotu		2021/2022		
Poziom kształcenia	II stopnia		Grupa zajęć		Grupa zajęć fakultatywnych Grupa zajęć powiązanych z prowadzonymi badaniami naukowymi w dziedzinie nauki związanej z kierunkiem - profil ogólnoakademicki		
Forma studiów	stacjonarne		Sposób realizacji		na uczelni		
Rok studiów	1		Język wykładowy		polski		
Semestr studiów	2		Liczba punktów ECTS		4.0		
Profil kształcenia	ogólnoakademicki		Forma zaliczenia		egzamin		
Jednostka prowadząca	Wydział Elektroniki -> Telekomunikacji i Informatyki -> Katedra Inżynierii Biomedycznej						
Imię i nazwisko wykładowcy (wykładowców)	Odpowiedzialny za przedmiot		dr inż. Magdalena Mazur-Milecka				
	Prowadzący zajęcia z przedmiotu		dr inż. Magdalena Mazur-Milecka				
Formy zajęć i metody nauczania	Forma zajęć	Wykład	Ćwiczenia	Laboratorium	Projekt	Seminarium	RAZEM
	Liczba godzin zajęć	30.0	0.0	15.0	15.0	0.0	60
	W tym liczba godzin zajęć na odległość: 0.0						
Aktywność studenta i liczba godzin pracy	Aktywność studenta	Udział w zajęciach dydaktycznych, objętych planem studiów		Udział w konsultacjach		Praca własna studenta	RAZEM
	Liczba godzin pracy studenta	60		5.0		35.0	100
Cel przedmiotu	Celem przedmiotu jest zapoznanie studentów z algorytmami widzenia komputerowego z szczególnym uwzględnieniem metod opartych na sieciach neuronowych i uczeniu maszynowym.						

Efekty uczenia się przedmiotu	Efekt kierunkowy	Efekt z przedmiotu	Sposób weryfikacji i oceny efektu
	[K7_U02] potrafi wykonywać zadania związane z kierunkiem studiów oraz formułować i rozwiązywać problemy z wykorzystaniem nowej wiedzy z fizyki i innych dziedzin nauki	Efektem procesu uczenia jest zdobycie przez studenta umiejętności poprawnego rozwiązania rzeczywistych problemów wizji komputerowej w dziedzinie inżynierii biomedycznej, odpowiednie dobranie metod oraz ocena wyników.	[SU1] Ocena realizacji zadania
	[K7_U04] potrafi wykorzystywać posiadaną wiedzę z zakresu metod i technik programowania oraz dobrać i zastosować właściwe metody i narzędzia programistyczne w tworzeniu oprogramowania komputerów albo programowania urządzeń lub sterowników wykorzystujących mikroprocesory albo elementy lub układy programowalne, charakterystycznych dla danego kierunku studiów, dokonując oceny i krytycznej analizy wykonanego oprogramowania, a także syntezy i twórczej interpretacji prezentowanych za jego pomocą informacji	Efektem procesu uczenia jest zdobycie przez studenta umiejętności praktycznego wykorzystania zdobytej wiedzy na temat algorytmów wizji komputerowej: zastosowanie odpowiednich metod i narzędzi, ocena ich skuteczności poprzez zastosowanie odpowiednich miar i ich właściwą interpretację.	[SU4] Ocena umiejętności korzystania z metod i narzędzi
	[K7_W03] zna i rozumie w pogłębionym stopniu budowę i zasady działania komponentów i systemów związanych z kierunkiem studiów, w tym teorii, metody i złożone zależności między nimi oraz wybrane zagadnienia szczegółowe – właściwe dla programu kształcenia	Efektem procesu uczenia jest zdobycie przez studenta wiedzy w zakresie teorii i metod dedykowanych rozwiązaniom zagadnień wizji komputerowej w inżynierii biomedycznej.	[SW1] Ocena wiedzy faktograficznej
	[K7_W04] zna i rozumie w zaawansowanym stopniu zasady, metody i techniki programowania oraz zasady tworzenia oprogramowania komputerów albo programowania urządzeń lub sterowników wykorzystujących mikroprocesory albo inne elementy lub układy programowalne, specyficznych dla kierunku studiów, a także organizację pracy systemów wykorzystujących komputery lub te urządzenia	Efektem procesu uczenia jest zdobycie przez studenta wiedzy w zakresie znajomości i umiejętności zastosowania metod i technik programowania a także bibliotek stosowanych podczas rozwiązywania zagadnień wizji komputerowej, m.in. detekcji i segmentacji obiektów, rozpoznania obrazu, czy klasyfikacji.	[SW1] Ocena wiedzy faktograficznej

Treści przedmiotu	1. Wprowadzenie do wizji komputerowej  2. Analiza tekstury i cechy tekstury (ocena ruchu)  3. Analiza kształtu i cechy opisu kształtu w obrazach  4. Analiza koloru/intensywności i powiązane cechy  5. Metody redukcji/selekcji cech, optymalizacja  6. Autoenkodery - poprawa jakości obrazów  7. Klasyfikacja obrazów z wykorzystaniem metod uczenia głębokiego  8. Metody segmentacji obrazów  9. Segmentacja obrazów (semantyczna)  10. Segmentacja obrazów (instance)  11. Metody lokalizacji i detekcji obiektów  12. Metody lokalizacji i detekcji obiektów  13. Generacja obrazów, adversarial images, poprawa jakości  14. Modele GAN w wizji komputerowej  15. Modele GAN w uczeniu maszynowym (augmentacje)		
Wymagania wstępne i dodatkowe	Wymagania wstępne: <ul style="list-style-type: none"> <li>w zakresie wiedzy teoretycznej - znajomość algorytmów przetwarzania i analizy obrazów oraz podstaw sieci neuronowych,</li> <li>w zakresie wiedzy praktycznej - podstawy języka Python oraz znajomość bibliotek dedykowanych do przetwarzania obrazów (np. OpenCV)</li> </ul>		
Sposoby i kryteria oceniania osiągniętych efektów uczenia się	Sposób oceniania (składowe)	Próg zaliczeniowy	Składowa oceny końcowej
	wykład	50.0%	40.0%
	laboratorium	50.0%	30.0%
	projekt	50.0%	30.0%
Zalecana lista lektur	Podstawowa lista lektur	1. <b>Computer Vision: Algorithms and Applications</b> , <i>Richard Szeliski</i>  2. <b>Programming Computer Vision with Python: Tools and algorithms for analyzing images</b> , <i>Erik Solem</i>  3. <b>Computer Vision: A Modern Approach</b> , <a href="#">David Forsyth</a> , <a href="#">Jean Ponce</a>	
	Uzupełniająca lista lektur	1. Deep Learning (Adaptive Computation and Machine Learning series), <a href="#">Ian Goodfellow</a> , <a href="#">Yoshua Bengio</a> , <a href="#">Aaron Courville</a>  2. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems 2nd Edition, <a href="#">Aurélien Géron</a>	

	Adresy eZasobów	
Przykładowe zagadnienia/ przykładowe pytania/ realizowane zadania	1. Autoenkodery i GAN  2. Detekcja obiektów  3. Rozpoznanie twarzy  4. Segmentacja  5. Klasyfikacja obrazów - inferencja na urządzeniach mobilnych	
Praktyki zawodowe w ramach przedmiotu	Nie dotyczy	