



**Politechnika
Śląska**

Rozprawa doktorska

„Monitorowanie rodzin pszczelich z wykorzystaniem urządzeń IoT w celu wykrywania sytuacji zagrażających życiu pszczół”

mgr inż. Anna WACHOWICZ

PROWADZĄCY PRACĘ

dr hab. inż. Dariusz Mrozek, prof. Politechniki Śląskiej

Katedra Informatyki Stosowanej

Wydział Automatyki, Elektroniki i Informatyki

GLIWICE 2023



Rzeczpospolita
Polska

Unia Europejska
Europejski Fundusz Społeczny



Niniejsza praca doktorska powstała w ramach projektu:

***„CyPhiS – Projekt nowoczesnych studiów
doktoranckich w dziedzinie systemów cyber–fizycznych”***

Projekt ten był dofinansowany z Funduszy Europejskich
w ramach Programu Operacyjnego Wiedza Edukacja Rozwój 2014-2020
przez Wydział Automatyki, Elektroniki i Informatyki Politechniki Śląskiej
w ramach Działania 3.2 Studia doktoranckie
na podstawie umowy nr POWR.03.02.00-00-I007/17-00

Realizacja niniejszej rozprawy nie byłaby możliwa bez wsparcia wielu osób.

Chciałabym szczególnie podziękować:

Promotorowi – Panu dr hab. inż. Dariuszowi Mrozkowi – za zachętę do realizacji doktoratu, wsparcie, cenne wskazówki i motywację do działania;

Kierownikowi projektu CYPHiS – Panu dr hab. inż. Andrzejowi Pulce – za możliwość udziału w projekcie;

Pszczelarzom – w szczególności Panu Adamowi i członkom Karpackiego Związku Pszczelarzy w Nowym Sączu oraz Kasi – za umożliwienie realizacji badań w pasiekach oraz wiele merytorycznych informacji dotyczących pszczół;

Managerowi – Szymonowi – za wsparcie, motywację i możliwość połączenia pracy zawodowej z nauką;

Pani Sylwii – za pomoc w organizacji i zarządzaniu obowiązkami oraz wysłuchanie w chwilach zwątpienia;

Przyjaciółom i Rodzinie, w szczególności: wujkowi Adamowi – za długie rozmowy i techniczne wskazówki, Grzegorzowi – za pomoc w przygotowaniu zdjęć, Gabrysi – za wsparcie i obecność, Monice – za poświęcony czas i rozmowy o każdej porze dnia (i nocy), babciom i Mamie Marii – za pomoc w opiece nad synem;

Rodzicom – za pomoc i wsparcie w każdym aspekcie życia, wiarę w moje umiejętności i możliwości w chwilach zwątpienia;

Siostrze – za obecność, zrozumienie, motywację, pomoc, wsparcie, wielogodzinne rozmowy i wszystko inne;

Mężowi – za rozmowy, wsparcie, cierpliwość, wyjazdy umożliwiające przeprowadzenie badań, przejęcie domowych obowiązków, opiekę nad synem, dostarczanie kawy i słodkich przekąsek, i wiele innych – po prostu za wszystko.

Pracę dedykuję:

Synowi

oraz

Dziadkowi – pszczelarzowi z pasji, który – chociaż bardzo chciał – nie zdążył przeczytać niniejszej rozprawy.

Spis treści

1. Wstęp	1
1.1. Motywacja i definicja problemu badawczego	2
1.2. Teza i cele rozprawy	4
1.3. Układ rozprawy.....	5
2. Tematyka rozprawy na tle nauk przyrodniczych i technicznych	7
2.1. Pszczelarstwo	7
2.1.1. Rola pszczół w przyrodzie.....	7
2.1.2. Zespół Masowego Ginięcia Pszczoły Miodnej	9
2.1.3. Dręcz pszczeli i warroza.....	12
2.2. Podstawy techniczne.....	15
2.2.1. Czwarta rewolucja przemysłowa.....	15
2.2.2. Internet Rzeczy	18
2.2.3. Systemy cyber-fizyczne.....	22
2.2.4. Sztuczna inteligencja, uczenie maszynowe, uczenie głębokie	24
2.2.5. Przetwarzanie brzegowe	27
3. Problematyka pracy na tle światowej literatury naukowej	31
3.1. Monitorowanie pszczół w przeszłości	31
3.2. Nowoczesne systemy wspierające pszczelarzy	33
3.2.1. Klasyfikacja systemów ze względu na rodzaj przetwarzanych danych ..	34
3.2.2. Klasyfikacja systemów ze względu na sposób przetwarzania danych ...	41
4. Przyjęte rozwiązania architektoniczne i algorytmiczne	49
4.1. Wczesne wersje algorytmów identyfikacji pszczół i detekcji dręcza pszczelego.....	49
4.1.1. Algorytmy działające na platformie RaspberryPi z akceleratorem USB Google Coral	50
4.1.2. Algorytmy działające na platformie NVidia Jetson Nano w wersji 2GB	52
4.2. Architektura i działanie urządzenia IoT umożliwiającego detekcję pszczół i ich klasyfikację pod kątem obecności dręcza pszczelego	55
4.3. Modele sieci neuronowych oparte na technice Single Shot Detector.....	60
4.4. Algorytm zasadniczy realizacji procesu wykrywania sytuacji zagrażających życiu pszczół.....	65
4.4.1. Algorytm adaptacyjnego wyznaczania okna zagęszczenia	66
4.4.2. Właściwa detekcja pszczół na klatkach filmowych	72
4.4.3. Detekcja dręcza pszczelego	74

4.5. Miary oceny jakości modeli	75
5. Wyniki eksperymentów badawczych	79
5.1. Dane wykorzystane do badań	79
5.2. Detekcja pszczół na klatkach filmowych	81
5.3. Detekcja dręcza pszczelego – klasyfikacja pszczół.....	88
5.4. Działanie algorytmu adaptacyjnego wyznaczania okna zagęszczenia.....	90
5.5. Wydajność czasowa działania całego systemu wykrywającego sytuacje zagrożające życiu pszczół	97
5.5.1. Wydajność algorytmu wyznaczającego okno zagęszczenia	98
5.5.2. Wydajność czasowa algorytmu detekcji pszczół oraz dręcza pszczelego 100	
5.6. Uzyskane wyniki na tle poprzednio zrealizowanych prac i badań.....	107
6. Podsumowanie.....	111
6.1. Podsumowanie zrealizowanych prac i uzyskanych wyników.....	111
6.2. Autorski wkład badawczy	112
6.3. Możliwe kierunki rozwoju	113
Publikacje Autorki w trakcie realizacji rozprawy doktorskiej	115
Bibliografia	117
Spis skrótów i symboli	125
Spis rysunków.....	129
Spis tabel.....	131
Spis algorytmów	133
Załączniki.....	135

1. Wstęp

Świat zmienia się każdego dnia – to stwierdzenie jest prawdziwe, zwłaszcza w kontekście rozwoju technologicznego. Nowoczesne systemy i urządzenia są wymyślane, produkowane, a następnie użytkowane praktycznie w każdym aspekcie naszego życia. Urządzenia mobilne (tablety oraz telefony) stały się narzędziami pracy, często niezbędnymi. Coraz większa liczba sportowców, trenujących zarówno amatorsko, jak i profesjonalnie, korzysta z inteligentnych opasek (*ang. smartband*) i zegarków (*ang. smartwatch*) mierzących parametry życiowe (np. tętno) i kontrolujących efektywność treningu. Nowe samochody są wyposażane w elektroniczne systemy, znacznie bardziej zaawansowane niż nawigacja czy podgrzewane siedzenia. Co więcej, autonomiczne pojazdy zaczynają pojawiać się na zagranicznych drogach – wciąż wymagany jest kierowca, który będzie nadzorował pracę systemu i zachowa czujność, natomiast nie jest konieczne używanie kierownicy, a nawet obserwacja drogi przez większość czasu jazdy. Każdego dnia miliony osób na całym świecie używają aplikacji i systemów bankowych, robią zakupy, kupują bilety lotnicze czy rezerwują wizyty u fryzjera z wykorzystaniem Internetu. Inteligentne domy nie występują już wyłącznie w filmach o przyszłości – systemy umożliwiające sterowanie oświetleniem za pomocą głosu, roboty sprzątające, zamki w drzwiach otwierane za pomocą aplikacji w telefonie to tylko niektóre przykłady nowoczesnych technologii wykorzystywanych na co dzień przez zwykłych ludzi. Postęp zauważalny jest także w medycynie. Zaawansowany sprzęt umożliwia diagnostykę medyczną niemalże w każdych warunkach, precyzyjne ramiona robotów, sterowane i nadzorowane przez wykwalifikowany personel, pozwalają na wykonywanie skomplikowanych operacji, a części wykorzystywane w protezach mogą być wydrukowane za pomocą drukarek 3D. Powyższe przykłady pokazują tylko część obszarów, w których zauważalna jest tzw. Czwarta Rewolucja Przemysłowa (*ang. The Fourth Industrial Revolution*), określana także jako Przemysł 4.0 (*ang. Industry 4.0*). Nowoczesne technologie i inteligentne rozwiązania są wdrażane także w rolnictwie. Systemy Internetu Rzeczy (*ang. Internet of Things, IoT*), a właściwie Przemysłowego

Internetu Rzeczy (ang. *Industrial Internet of Things, IIoT*) pomagają rolnikom w planowaniu właściwego nawadniania upraw, nawożenia i zbierania plonów. Ma to na celu usprawnienie procesów produkcji żywności oraz ograniczenie strat powstających w ich czasie. Naukowcy w laboratoriach pracują nad syntetycznymi odpowiednikami pożywienia, mającymi zmniejszyć ślad węglowy jego produkcji i zapewnić dostęp do żywności jak największej liczbie ludzi. Założenie to jest szczególnie ważne, w związku z globalnym kryzysem żywnościowym (ang. *Global Food Crisis*), którego ludzkość może doświadczyć szybciej, niż na początku szacowano.

Zastosowanie nowoczesnych technologii w obszarze rolnictwa jest bardzo ważne, należy jednak pamiętać, że roślinom do rozwoju i reprodukcji niezbędne jest zapylanie. Proces przenoszenia pyłku może być wykonany przez owady, zwierzęta, a także za pomocą czynników środowiskowych, takich jak wiatr czy woda. Wśród owadów zapylających można wyróżnić muchy, żuki, motyle, ćmy, trzmiele, osy oraz pszczoły, które są jedną z najbardziej znaczących grup zapylaczy. Stanowią one zdecydowaną większość insektów zapylających i są odpowiedzialne za wspomaganie blisko 90% światowej produkcji żywności [1]. Niestety, z roku na rok na całym świecie obserwuje się zmniejszanie populacji wielu gatunków pszczół, co może wpływać na zaburzenie różnorodności biologicznej.

1.1. Motywacja i definicja problemu badawczego

Doniesienia naukowe ostatnich lat w tematyce masowego wymierania pszczół (ang. *Colony Collapse Disorder, CCD*), kampanie organizacji pozarządowych zachęcających do ochrony i adopcji pszczół oraz niepokojące obserwacje lokalnych pszczelarzy były motywacją do zgłębienia tematyki związanej ze zjawiskiem depopulacji pszczół. Każdego roku, z różnych, często niejednoznacznie określonych przyczyn, giną tysiące osobników wielu gatunków pszczół, w tym popularnej w Polsce pszczoły miodnej (łac. *Apis Mellifera*). Wśród czynników mogących mieć wpływ na depopulację tego gatunku wymienia się między innymi: zmiany klimatyczne i środowiskowe, a także częste wykorzystanie pestycydów w rolnictwie [2], [3]. Sytuacja zmniejszania się światowej populacji pszczół jest niebezpieczna – bez procesu zapylania uprawy rolne, kwiaty

i drzewa nie będą się rozwijały, a to może doprowadzić do wymierania wielu gatunków zwierząt, a – w najgorszym scenariuszu – również ludzi.

Jednym z popularnych sposobów na ochronę pszczół przed niebezpiecznymi sytuacjami mogącymi doprowadzić do osłabienia bądź wymarcia pszczelej rodziny jest regularne przeprowadzanie przeglądów ula. Takie inspekcje umożliwiają zauważenie nieprawidłowości (np. obecność pasożyta, złą kondycję zdrowotną pszczół) na wczesnym etapie oraz podjęcie odpowiednich działań ochronnych. Pomimo dużej czasochłonności tego zadania (należy obejrzeć każdą ramkę w ulu), pszczelarze posiadający niewielkie pasieki są w stanie dokonywać regularnych kontroli uli. Niestety, posiadając większą pasiekę, kontrolowanie kondycji każdej rodziny pszczelej tak, żeby na wczesnym etapie zauważyć potencjalne zagrożenia, jest trudne do zrealizowania, nawet przy zaangażowaniu kilku osób.

Z innej perspektywy, technologia rozwija się na wielu płaszczyznach i jest wdrażana w wielu obszarach. Również w pszczelarstwie widoczne są zmiany, aczkolwiek w tej dziedzinie proces unowocześniania i usprawniania zachodzi dużo wolniej niż w innych. Termin określający wykorzystanie nowych technologii w pszczelarstwie został zdefiniowany w 2017 roku jako *Precyzyjne pszczelarstwo* (ang. *Precision Beekeeping (PB)* lub *Precision Apiculture (PA)*) [4], [5]. Pomimo rozwoju *precyzyjnego pszczelarstwa* w ostatnich latach, literatura wciąż nie opisuje zbyt wielu systemów Internetu Rzeczy dedykowanych dla pszczelarstwa. Umożliwienie monitorowania ula z wykorzystaniem inteligentnego systemu mogłoby pomóc pszczelarzom w walce z depopulacją pszczół, jednocześnie nie stanowiąc zagrożenia dla owadów.

Zarówno niepokojące zjawisko zmniejszania się liczby pszczół na świecie, jak i brak kompleksowych rozwiązań technologicznych w tej dziedzinie był motywacją do poruszenia w niniejszej dysertacji problemu wykrywania sytuacji, mogących stanowić zagrożenie dla zdrowia i/lub życia pszczół. Niewątpliwie jednym z przykładów takich zagrożeń jest obecność dręcza pszczelego, pajęczaka wywołującego warrozę, w rodzinie pszczelej. W literaturze warroza jest jedną z wymienianych przyczyn masowego ginięcia pszczół [2], [6], [7], a wśród nowoczesnych systemów dedykowanych pszczelarzom brakuje takich, które umożliwiałyby jej wykrycie na wczesnym etapie.

1.2. Teza i cele rozprawy

Po wnikliwej analizie istniejących rozwiązań, a także konsultacjach z pszczelarzami, mających na celu określenie ich potrzeb i oczekiwań w kwestii monitorowania uli, został zdefiniowany następujący cel rozprawy:

Zaprojektowanie i implementacja kompleksowego systemu Internetu Rzeczy (IoT), umożliwiającego monitorowanie rodzin pszczelich, który pomoże pszczelarzom wykrywać potencjalnie niebezpieczne sytuacje – obecność dręczca pszczelego (ang. *Varroa Destructor*) – w rodzinie pszczelej, który może spowodować chorobę (warrozę) oraz doprowadzić do śmierci pszczół.

Mając na uwadze powyższy cel, postawiona została następująca teza:

Możliwe jest wykrycie występowania dręczca pszczelego w rodzinie pszczelej w czasie rzeczywistym za pomocą urządzenia IoT wyposażonego w kamerę, wykorzystującego model przetwarzania brzegowego i ograniczając obszar analizy materiału filmowego do analitycznego okna zagęszczenia wyznaczanego w sposób adaptacyjny.

Aby ułatwić realizację postawionej tezy, zdefiniowane zostały następujące wymagania:

1. Urządzenie nie powinno zakłócać życia pszczół w pasiece oraz powinno mieć neutralny wpływ na środowisko.
2. Zaprojektowany system powinien przetwarzać obrazy z kamery z uwzględnieniem różnych poziomów oświetlenia (nasłonecznienia) otoczenia.
3. Modele detekcji pszczół oraz ich klasyfikacji, użyte do wykrywania osobników z dręczem pszczelim powinny umożliwić pracę systemu w czasie rzeczywistym.
4. Rozdzielczość kamery powinna stanowić kompromis pomiędzy jakością obrazu, a czasem jego przetwarzania.
5. Wykorzystanie procesora GPU powinno umożliwić przetwarzanie jak największej liczby klatek analizowanego filmu.

Na potrzeby opisu zaimplementowanych algorytmów, przeprowadzonych badań oraz zwiększenia czytelności przedstawionych wyników, w dalszej części pracy stwierdzenie „*chora pszczoła*” będzie odnosiło się do osobnika, u którego została wykryta obecność dręcza pszczelego.

1.3. Układ rozprawy

W dalszej części niniejszej rozprawy doktorskiej przedstawiono podstawy teoretyczne poruszanego zagadnienia, opis konstrukcji urządzenia IoT oraz architekturę systemu umożliwiającego monitorowanie pszczół, a także badania eksperymentalne pozwalające dowieść słuszności postawionej tezy.

W rozdziale 2 opisane zostały przyrodnicze i techniczne podstawy, które mają na celu wprowadzenie czytelnika w tematykę poruszaną w dysertacji. Część poświęcona pszczelarstwu przedstawiona została w rozdziale 2.1. Najważniejsze informacje dotyczące pszczół miodnych oraz ich roli w przyrodzie zaprezentowane są w rozdziale 2.1.1. Zjawisko depopulacji pszczół (zespół masowego ginięcia pszczoły miodnej, CCD), wraz z najczęstszymi objawami oraz możliwymi przyczynami, zarówno środowiskowymi, jak i wynikającymi z działalności człowieka, zostały opisane w rozdziale 2.1.2. Charakterystyka niebezpiecznego dla pszczół dręcza pszczelego, wywołującego chorobę – warrozę, którego wykrywanie jest określone w celach tej pracy, została przedstawiona w rozdziale 2.1.3. W części dotyczącej podstaw technicznych (rozdział 2.2), opisane zostały zagadnienia, takie jak Czwarta Rewolucja Przemysłowa, w literaturze określana też terminem Przemysł 4.0 (rozdział 2.2.1), Internet Rzeczy i przykłady systemów opartych o tę koncepcję (rozdział 2.2.2), systemy cyberfizyczne (rozdział 2.2.3), algorytmy uczenia maszynowego (w tym używane do klasyfikacji obrazów, rozdział 2.2.4) oraz przetwarzanie brzegowe (ang. *edge computing*, rozdział 2.2.5).

Przegląd literaturowy został przedstawiony w rozdziale 3. Omówione zostały prace powiązane z monitorowaniem pszczół, wraz z systemami o podobnej charakterystyce, wykorzystywane w rolnictwie oraz użyte w nich rozwiązania sprzętowe i algorytmiczne. W rozdziale 3.1 zaprezentowane zostały sposoby i systemy umożliwiające monitorowanie pszczół wykorzystywane w przeszłości. Następny podrozdział 3.2 poświęcony jest rozwiązaniom dla pszczelarstwa oraz rolnictwa z ostatnich lat.

Klasyfikacja systemów ze względu na przetwarzane dane została szczegółowo omówiona w rozdziale 3.2.1, natomiast klasyfikacja systemów ze względu na sposób przetwarzania danych – w rozdziale 3.2.2.

Kolejny rozdział 4 opisuje architekturę systemu, jego działanie, wykorzystane algorytmy oraz modele sieci neuronowych. W 4.1 opisane zostały wstępne wersje algorytmów detekcji pszczół i dręcza pszczelego. Następnie, w rozdziale 4.2, przedstawiona została architektura i sposób działania aktualnego rozwiązania, będącego przedmiotem niniejszej rozprawy. W kolejnym rozdziale 4.3 omówiona została architektura modeli konwolucyjnych sieci neuronowych opartych na technice Single Shot Detector. Rozdział 4.4 opisuje algorytm zasadniczy realizacji procesu detekcji pszczół i dręcza pszczelego, w tym autorski algorytm adaptacyjnego wyznaczania okna zagęszczenia, który umożliwi dopasowanie obszaru analizowanego obrazu, w taki sposób, aby uniknąć przetwarzania fragmentów, na których w danym momencie nie znajdują się obiekty pszczół. Ponadto, w rozdziale 4.4 opisano metryki zastosowane do oceny jakości sposobu rozpoznawania pszczół oraz detekcji dręcza pszczelego.

W rozdziale 5 przedstawione zostały szczegóły dotyczące przeprowadzonych badań. W podrozdziale 5.1 zostały opisane dane, których użyto w procesie uczenia modeli i ich walidacji, a także sposób pozyskiwania i przygotowania tych danych. Następnie przedstawione zostały wyniki opisujące jakość detekcji pszczół (rozdział 5.2), dręcza pszczelego (rozdział 5.3) oraz wyznaczone w procesie działania algorytmu adaptacyjnego analizy danych okna zagęszczenia (rozdział 5.4). Rozdział 5.5 poświęcony jest wynikom przeprowadzonych eksperymentów badawczych w kontekście czasu działania poszczególnych etapów urządzenia IoT umożliwiającego wykrywanie sytuacji zagrażających życiu pszczół. W ostatniej części rozdziału (5.6) znajduje się porównanie wszystkich uzyskanych rezultatów.

Rozdział 6 stanowi podsumowanie całej rozprawy. Na początku rozdziału uzyskane wyniki zostały poddane analizie pod kątem prawdziwości tezy oraz realizacji celów postawionych w pierwszym rozdziale. Następnie opisany został autorski wkład badawczy. Ostatni podrozdział 6.3 opisuje możliwości dalszych prac w tym temacie.

2. Tematyka rozprawy na tle nauk przyrodniczych i technicznych

Motywacja i realizacja niniejszej rozprawy doktorskiej jest głęboko osadzona zarówno w dziedzinie nauk przyrodniczych (konkretnie – w pszczelarstwie), jak również technicznych – informatyce. Aby dobrze zrealizować założone cele, konieczne jest ich odpowiednie umiejscowienie w tychże dziedzinach nauki. W poniższych podrozdziałach zaprezentowane zostaną najważniejsze zagadnienia związane z przyrodniczymi i technicznymi aspektami realizowanego w ramach niniejszej rozprawy tematu.

2.1. Pszczelarstwo

Pszczelarstwo, podobnie jak prowadzenie upraw i plantacji roślinnych oraz hodowla bydła, jest jedną z gałęzi rolnictwa [8]. Wywodzi się z bartnictwa, czyli rzemiosła polegającego na hodowli pszczół leśnych w specjalnie wydrążonych dziuplach w drzewach – barciach. Współczesne metody stosowane gospodarce pasiecznej są nieustannie rozwijane, a ich obecny kształt zawdzięcza się odkryciom wybitnych pszczelarzy ostatnich stuleci – Jana Dzierżona, Karla von Frischa oraz Jerzego Woykego [8].

2.1.1. Rola pszczół w przyrodzie

Zapylenie jest procesem polegającym na przeniesieniu pyłku kwiatu. W zależności od rodzaju rośliny, pyłek przenoszony jest z pylników kwiatu na znamiona słupka (w przypadku roślin okrytonasiennych) lub bezpośrednio na zalążek (u roślin nagonasiennych). Wyróżniane są dwa rodzaje zapylenia – samozapylenie (autogamia) oraz zapylenie krzyżowe (ksenogamia). Samozapylenie ma miejsce, gdy pyłek pochodzi z tego samego bądź innego kwiatu tej samej rośliny, natomiast do zapylenia krzyżowego dochodzi, kiedy pyłek jest przeniesiony z innej rośliny w ramach jednego gatunku. Owoce i nasiona

powstające w wyniku samozapylenia są mniej dorodne niż zapyłone krzyżowo, dlatego też wiele gatunków roślin posiada naturalne „blokady” uniemożliwiające autogamię. Pyłek może zostać przeniesiony przez czynnik środowiskowy – wiatr lub wodę, oraz zwierzęta, zarówno te mniejsze – owady, jak i większe – ptaki czy ssaki. Zwierzęta umożliwiają rozmnażanie się 90% roślin okrytonasiennych – zapewnia to bioróżnorodność na Ziemi i dzięki temu zachowywana jest ciągłość ekosystemów [9]. Światowa produkcja żywności w 60% opiera się na roślinach wiatropylnych i wodopylnych, podczas gdy zwierzęta zapyłające umożliwiają globalną produkcję żywności w blisko 35% (z czego dla upraw tropikalnych zwierzęta zapyłają ok. 70% plonów, a w przypadku upraw w Europie ok. 84%) [10]. Uprawy zbóż, które są podstawą diety człowieka – są wiatropylne, jednak produkty dostarczające duże ilości wartości odżywczych, takie jak owoce i warzywa, potrzebują pszczoł, aby się rozmnażać i rozwijać.

Pszczoły stanowią dwie trzecie wszystkich zapyłających owadów. Jest jednak kilka cech, które czynią je wyjątkowymi i wydajnymi [11]. Pszczoły żywią się nektarem roślinnym i mogą przenosić duże ilości pyłku na swoich odwłokach [12]. W porównaniu do innych owadów, potrafią pokonywać większe odległości [13], oraz zapyłać wiele gatunków roślin, wzbogacając jednocześnie ekosystemy [14]. Ponadto, pszczoły należą do grupy gatunków tworzących społeczności [15] – na jedną pasiekę składa się kilka a nawet kilkanaście rodzin. W 1989 roku Thomas D. Seeley określił pszczołę rodziny jako *superorganizm* [16], w którym poszczególne osobniki współpracują ze sobą, razem dbając o dobro rodziny. Pszczoły te pełnią także określone role, które są zależne od płci. Każda rodzina posiada matkę – królową, odpowiedzialną za reprodukcję swojej rodziny. Pszczela matka może złożyć nawet 1500 jaj dziennie. Jej wygląd różni się od robotnic – jest większa, ma owalną głowę, a jej korpus ma budowę uniemożliwiającą zbieranie i przenoszenie pyłku. Królowa może przeżyć 5 lat, jednak w pasiekach zaleca się częstsze wymiany matek, nawet co roku. Pszczoły płci męskiej – trutnie – nie mają innych „obowiązków” poza zapładnianiem królowej. Te osobniki, którym uda się wypełnić zadanie, umierają krótko po zapłodnieniu matki. Trutnie, którym się nie powiedzie zostają wypędzane z ula przez robotnice i giną z głodu oraz zimna poza ulem. Ostatnią, ale na pewno nie najmniej ważną grupą, są pszczoły – robotnice. Do ich obowiązków należy zbieranie pyłku, karmienie larw, dbanie o jaja złożone przez królową, robienie plastrów miodu oraz dbanie o dobrą kondycję ula – jego wentylację i zaopatrzenie w pożywienie. Zimą robotnice zapewniają utrzymanie optymalnej temperatury w ulu – dzięki wytwarzanym przez nie wibracjom ogrzewają się wzajemnie, rotują wewnątrz ula tak, by osobniki będące przy ścianach nie zmarły z wyziębienia [17].

Australijscy badacze w swojej pracy [18] pokazują, że pszczoły mogą przyczyniać się do realizacji nawet piętnastu ze zdefiniowanych przez Zgromadzenie Ogólne Organizacji Narodów Zjednoczonych w 2015 roku celów zrównoważonego rozwoju (ang. *Sustainable Development Goals*, SDGs). W rezolucji, znanej też jako Agenda 2030 [19], określonych zostało siedemnaście głównych celów, które powinny być zrealizowane do końca 2030 roku. Ich osiągnięcie ma prowadzić do poprawy warunków życia ludzi oraz chronić zasoby naturalne na całym świecie. Przykładowo, pszczoły mogą pomóc w realizacji celów *Zero głodu* (ang. *Zero hunger*) oraz *Dobre zdrowie i samopoczucie* (ang. *Good health and well-being*) – zapylenie roślin przez pszczoły wzbogaca żywność w wartości odżywcze oraz zwiększa ilość plonów. Produkty pochodzenia pszczelego (np. miód, pyłek, wosk) mogą być używane w medycynie tradycyjnej oraz nowoczesnej jako wspomaganie leczenia chorób niezakaźnych. Udział pszczół w osiągnięciu tych celów wydaje się naturalnym, jednak autorzy powyższej pracy wymieniają także mniej oczywiste przykłady. Są to między innymi: wpływ na jakość edukacji – szkolenia z opieki nad pszczołami mogą zwiększyć szansę na lepsze warunki zatrudnienia (lub dodatkowy dochód); równość płci (ang. *gender equality*) – uprawianie pszczelarstwa przez kobiety może je skłonić do większego zaangażowania się w sprawy społeczne i polityczne; zrównoważone miasta i społeczności (ang. *sustainable cities and communities*) – zapylenie może prowadzić do samowystarczalności przestrzeni publicznych, parków i ogrodów. Uwzględniając powyższe informacje, znaczenie pszczół dla ludzkości powinno być określane jako nie do przecenienia.

2.1.2. Zespół Masowego Ginięcia Pszczoły Miodnej

Straty rodzin pszczelich są zauważane od blisko dwudziestu lat na całym świecie – między innymi w Stanach Zjednoczonych [3], [20], Kanadzie [6], Azji [21] oraz Europie [22]. Zjawisko to zostało formalnie nazwane jako Zespół Masowego Ginięcia Pszczoły Miodnej (ang. *Colony Collapse Disorder, CCD*) w 2007 roku [20]. Wśród cech charakteryzujących jego występowanie wymienia się następujące sytuacje: *nagłe zniknięcie rodziny pszczelnej, z niewielką liczbą martwych pszczół pozostawionych w ulu; młode osobniki pozostawione w ulu, bez królowej i robotnic; pszczela matka zostawiona w ulu z małą liczbą pszczół oraz niskimi zapasami pyłku i miodu* [23].

Straty pszczół, obserwowane w tak wielkiej skali, bez wątpienia mają ogromny wpływ na środowisko. Jednak inną, dość ważną kwestią, są straty finansowe, liczone w milionach dolarów każdego roku. W 2007 procesy zapyłania upraw, które zostały zagrożone z powodu występowania zjawiska masowego ginięcia pszczół, przyczyniły się do produkcji owoców, orzechów i warzyw o łącznej wartości 75 miliardów dolarów [24]. W związku z tym koszty ponoszone w wyniku strat upraw spowodowanych zmniejszoną liczbą owadów zapyłających mogą być liczone w milionach dolarów rocznie.

Do dzisiaj nie udało się ustalić pojedynczej przyczyny zjawiska CCD. Wśród możliwych czynników wymienia się wiele zmiennych. Pierwszą z nich są zauważalne zmiany klimatyczne [25], które mogą przyczyniać się do zmniejszania populacji pszczół. Jednym z najbardziej znanych objawów zmian klimatycznych jest globalne ocieplenie – w Europie i Stanach Zjednoczonych, gdzie pszczoły miodne gatunku *Apis Mellifera* występują najczęściej, co roku odnotowywane są bardzo wysokie, często rekordowe, temperatury. Pszczoły są wrażliwe na warunki atmosferyczne – w lecie nie opuszczają uli, gdy na zewnątrz są bardzo wysokie temperatury, co pozwala na utrzymanie chłodniejszej temperatury w środku. Natomiast jesienią, zimą i wczesną wiosną, pozostając w ulu, ogrzewają go oraz wszystkie osobniki w środku. Pszczoły nie opuszczają też ula w deszczową pogodę. Pozostanie w ulu prowadzi do zmniejszenia ilości pożywienia zapewnionego dla danej rodziny i sprawia, że niektóre pszczoły i larwy mogą być niedożywione, co z kolei może prowadzić do depopulacji rodziny. Ponadto, gorące temperatury mogą znacząco wpływać na zmniejszenie liczby plonów [26], a mniejsza liczba plonów to mniej pożywienia dla pszczół w następnym sezonie itd.

Zmiany klimatyczne, zaliczane do czynników środowiskowych, odpowiadają również za rozprzestrzenianie się pasożytów atakujących całe pasieki [2]. Grzyby, bakterie, wirusy, roztocza – każda z tych grup zalicza się do stresorów (ang. *stressors*) pszczół i ma negatywny wpływ na ich zdolności do nauki i zapamiętywania, powodując problemy z nawigacją i powrotem do ula [7]. Pasożyty rozprzestrzeniają się szybko – mogą w krótkim czasie zaatakować wiele pasiek na sąsiadujących terenach. Wśród szkodników najczęściej atakujących pszczoły gatunku *Apis Mellifera* wymienia się wirusy: kaszmirski (ang. *Kashmir Bee Virus, KBV*), izraelskiego ostrego paraliżu pszczół (ang. *Israeli Acute Paralysis Virus, IAPV*) [27], ostrego paraliżu pszczół, (ang. *Acute Bee Paralysis Virus, ABPV*), wirusa zdeformowanych skrzydeł (ang. *Deformed Wing Virus, DWV*) oraz pierwotniaki gatunku *Nosema* [2]. Wirusy: kaszmirski, ostrego paraliżu pszczół IAPV oraz ABPV, są ze sobą blisko spokrewnione i należą do wirusów rodziny *Dicistroviridae* [28]. Wirus kaszmirski

został wyizolowany w 1974 roku i, chociaż nie daje widocznych objawów, jest nośnikiem wielu różnych chorób, w tym warrozy. Izraelski wirus paraliżu pszczoł IBPV jest wirusem opisanym przez izraelskich badaczy w 2002 [29], który powoduje drżenie skrzydeł i prowadzi do paraliżu pszczoły. Podobnie jak wirus kaszmirski, atakuje całe rodziny, bez względu na płeć i stadium rozwoju pszczoły – jaja, larwy, dorosłe osobniki, trutnie i robotnice, a nawet matki, jednak najbardziej podatne na zakażenie są młode, niedojrzałe osobniki. Wirus izraelski oddziałuje na mózg, mięśnie, jelita i inne organy [30]. Jego obecność potwierdzono również w pyłku, nektarze oraz nasieniu trutni z zarażonych rodzin. Wirus ABPV, również powodujący paraliż, atakuje głównie dorosłe osobniki, chociaż zdarza się, że zainfekowane są też larwy. Narażona na działanie wirusa pszczoła, traci umiejętność latania, zaczyna się trząść i gubi włosy z odwłoku. Na końcu dochodzi do paraliżu i śmierci. Wirus zdeformowanych skrzydeł należy do grupy jednych z najbardziej rozpowszechnionych wirusów atakujących pszczoły, zarówno te w pasiekach, jak i żyjące dziko. Ponadto, jego występowanie zostało potwierdzone również wśród innych gatunków, np. trzmieli. U dorosłego zainfekowanego osobnika obecność wirusa można rozpoznać poprzez wyglądające jak zgniecione skrzydła i zdeformowany odwłok. Każdy z powyżej opisanych wirusów jest wymieniany jako możliwa przyczyna masowego ginięcia pszczoł. Wirusy te mogą być przenoszone przez pajęczaka – dręczkę pszczelego, który został szczegółowo opisany w dalszej części tej rozprawy (rozdział 2.1.3). Wśród innych pasożytów, grzyb jednokomórkowy gatunku *Nosema ceranae* jest często opisywany w literaturze [23], [31], [32]. Został on zidentyfikowany w 1994 roku [33], [34] i określony jako potencjalnie niebezpieczny kilka lat później [35]. Jego występowanie zostało odnotowane na całym świecie [32]. Nosemoza, choroba wywoływana przez tego jednokomórkowego pasożyta, atakuje jelita pszczoł i w nich się rozwija. Chore osobniki są mniej wydajne w zapewnianiu pożywienia rodzinie i mają problemy z powrotem do ula. Zarażone chorobą, umierają w przeciągu 8-10 dni. Zmniejszająca się liczba pracownic, coraz większa liczba zarażonych pszczoł w połączeniu z niewystarczającą ilością pożywienia w ulu prowadzi do szybkiego wymarcia rodziny.

Analizując możliwe przyczyny zjawiska masowego ginięcia pszczoł należy wziąć pod uwagę wykorzystanie pestycydów w rolnictwie [1], [36]. Substancje chemiczne zakłócają naturalny ekosystem – są wykorzystywane do spryskiwania roślin, jednak niewielkie cząsteczki przenikają do wód gruntowych, negatywnie oddziałując na dzikie drzewa i rośliny. Pestycydy są znajdowane w nektarze i pyłku, co powoduje, że owady zapylające są na nie narażone w czasie wiosenno-letnim. Ekspozycja pszczoł na pestycydy może wpływać na zdolność zapamiętywania lokalizacji ula, włączając w to trudności w poruszaniu się

i orientację w terenie. Pszczoły, które nie potrafią wrócić do ula giną, a taka sytuacja również prowadzi do zmniejszania się ich populacji. Pierwsze próby udowodnienia, że substancje chemiczne stosowane w rolnictwie mogą przyczyniać się do CCD pojawiły się w 2008 roku [37], krótko po zdefiniowaniu samego zjawiska. Po analizie 108 próbek pyłku i wosku pszczelego, autorzy znaleźli pozostałości blisko 50 różnych pestycydów. Okazało się, że fluwalinat, początkowo uznawany za relatywnie bezpieczny dla pszczół związek chemiczny, jest dużo bardziej toksyczny. Co więcej, kilka innych rodzajów pestycydów zostało uznanych jako substancje niebezpieczne, które mogą wchodzić w interakcję z innymi stresorami pszczół, np. wirusem izraelskim. Badanie to pokazało, że konieczna jest dalsza analiza powiązań pomiędzy masowym ginięciem pszczół, a użyciem pestycydów. W innej pracy [38], sprawdzona została zależność wpływu chemicznych substancji ochrony roślin na czas powrotu pszczół do ula po zebraniu pyłku. Jej autorzy zaproponowali użycie znaczników RFID (ang. *Radio-Frequency Identification*), co umożliwiło kontrolowanie czasu, jakiego pszczoły potrzebowały na pokonanie dystansu pomiędzy rośliną i ulem. Badanie pokazało, że zastosowanie fipronilu może zwiększyć czas powrotu pszczoły do ula, jednak autorzy zastrzegli informację, że eksperymenty nie były przeprowadzane w naturalnym środowisku pszczół.

2.1.3. Dręcz pszczeli i warroza

Dręcz pszczeli (łac. *Varroa Destructor*) – gatunek pajęczaka, jest bez wątpienia jednym z najbardziej rozpoznawalnych oraz najgroźniejszych szkodników atakujących rodziny pszczele. Szybko się rozprzestrzenia – w krótkim czasie jego obecność można zauważyć w kilku sąsiadujących ze sobą pasiekach, atakuje pszczoły bez względu na etap rozwoju, czy pełnią w społeczności rolę – zagraża zarówno larwom, jak i robotnicom czy trutniom. Dręcz pszczeli jest dostrzegalny gołym okiem – ma długość około 1mm, a jego szerokość może osiągać 1.7 mm. Osobniki płci żeńskiej mają kolor brązowo-czerwony, natomiast męskie – biały. [6], [39]. Jego obecność została odnotowana w pasiekach pszczół europejskiego gatunku *Apis Mellifera* oraz azjatyckiego *Apis Cerana*. Rys. 1 przedstawia pszczołę zaatakowaną przez dręcza pszczelego.

Na początku, dręcz pszczeli atakuje dorosłe pszczoły. Pożywienie pobiera z komórek tłuszczowych (a nie, jak pierwotnie uważano, [40] z hemolimfy) żywiciela [41], żerując na nim przez około dwa tygodnie. Następnie, przenosi się do środka ula i atakuje larwy pszczół. Żerując na nich, rozmnaża się i rozprzestrzenia w całym ulu. Pajęczak żyje na swoim

żywicielu około dwóch miesięcy w okresie letnim. Gdy jednak dostanie się do ula i rozmnoży na dużą skalę, potrafi przeżyć całą zimę wykorzystując dorosłe osobniki do odżywiania się. Bez żywiciela cykl życia dręczka pszczelego skraca się do kilku dni [42].



Rys. 1 Pszczoła zaatakowana przez dręczka pszczelego

Dręcz pszczeli wywołuje chorobę pszczół – warrozę, która wpływa negatywnie zarówno na całą rodzinę, jak i pojedyncze osobniki. Funkcje życiowe zarażonej pszczoły zostają zaburzone – mniejsza liczba komórek tłuszczowych oraz niewystarczająca ilość wody w organizmie powoduje zmniejszenie korpusu, prowadzi do zachwiania gospodarki hormonalnej oraz problemów z termoregulacją. Ma również wpływ na rozmnażanie – w 2020 roku zaobserwowane zostały powiązania pomiędzy wielkością trutni a ilością nasienia, które produkują [43]. Ponadto, pszczoły z mniejszą liczbą komórek tłuszczowych mają mniej energii, nie są zdolne do wykonywania swoich zadań wystarczająco dobrze – np. zdobywają mniej pożywienia, wykazują problemy z orientacją w terenie i powrotem do ula, co ma wpływ na wydajność robotnic. W porównaniu do zdrowych osobników, pszczoły z warrozą żyją krócej. Choroba może także powodować zmiany w wyglądzie, np. deformować skrzydła. Co więcej, rany spowodowane warrozą są szkodliwe dla systemu odpornościowego i mogą narażać pszczoły na zakażenie innymi infekcjami. Często, po zachorowaniu na warrozę, pszczoła zostaje zakażona innymi wirusami – wirusem zdeformowanych skrzydeł (DWV) oraz wirusem ostrego paraliżu pszczół (ABPV), które zostały opisane we wcześniejszych częściach tego rozdziału.

Występowanie warrozy w pasiekach stanowi duże zagrożenie, dlatego pszczelarze próbują różnych metod, zarówno wykorzystujących środki chemiczne, jak i bardziej naturalne substancje, żeby jej zapobiegać. Wśród syntetycznych metod często wykorzystywane są mitycydy, które zabijają dręczka pszczelego. Ich działanie jest zadowalające w przypadku

warrozy znajdującej się na dorosłych pszczołach, natomiast nie sprawdzają się w przypadku zarażonych jaj wewnątrz ula. Ponadto, lista substancji i produktów przeznaczonych do eliminacji warrozy jest ograniczona do kilku – pyretroidy, tau-fluwalinat, flumethrin, formamidyna, amitraza oraz kwas fosforoorganiczny [42]. Rozwiązaniem tego problemu było wprowadzenie na rynek produktów ze związkami akarycydowymi, które mają na celu unicestwienie dręcza pszczelego, gdy opuści larwy. Ich zaletą jest łatwość użycia, natomiast wśród wad wymienia się dwie cechy. Po pierwsze, związki te przedostają się do produktów pszczelich i mogą być toksyczne dla pszczół przy dłuższym czasie ekspozycji. Drugą wadą jest nabywanie przez dręcza pszczelego odporności na te związki (w podobny sposób, jak bakterie uodparniają się na antybiotyki), poprzez mutacje w DNA tego pajęczaka.

Wady produktów z substancjami syntetycznymi wymusiły na pszczelarzach konieczność poszukiwania nowych, bardziej naturalnych metody eliminacji warrozy z pasiek. Wśród organicznych substancji wykorzystywanych w tym celu znajdują się kwasy – szczawiowy i mrówkowy oraz tymol [44]. Choć mniej skuteczne, są bardziej bezpieczne dla pszczół, a wśród osobników dręcza pszczelego nie uruchamiają reakcji odpornościowej w takim stopniu, jak substancje syntetyczne. Wśród naturalnych metod zapobiegania występowaniu warrozy wymienia się oczyszczanie ula z zarażonego czerwiu i martwych pszczół – jest to zabieg higieniczny, który pomaga w zapobieganiu występowania także innych wirusów i powinien być wykonywany przez pszczelarzy regularnie. Inną, niechemiczną metodą przynoszącą dobre rezultaty w zakresie eliminacji warrozy jest hipertermia. Stosuje się ją od blisko pięćdziesięciu lat, a jej działanie jest oparte o różnicę w odporności pszczół i dręcza pszczelego na ciepło – dręcz ginie w temperaturze 38°C, a w temperaturze 36,5°C jego zdolności reprodukcyjne znacznie maleją [45]. W przypadku tej metody należy zachować ostrożność przy ogrzewaniu ula, ponieważ zbyt częste lub niekontrolowane nagrzewanie ula może wpłynąć na jakość nasienia trutni [46]. Prawidłowe zastosowanie hipertermii przynosi dobre rezultaty, tym samym pozostając bezpieczną dla pszczół metodą prewencyjną.

Oprócz stosowania metod mających na celu ochronę przed zarażeniem warrozą, ważne jest też ciągle monitorowanie stanu zdrowia pszczół. W wykrywaniu występowania warrozy w ulu czas ma kluczowe znaczenie – im szybciej obecność dręcza pszczelego w pasiece zostanie zauważona, tym większe są szanse na ochronę przed jego rozprzestrzenieniem się wśród rodzin pszczelich na sąsiednich obszarach. Zastosowanie nowoczesnych technologii, opisanych szczegółowo w kolejnych rozdziałach, może

przyczynić się do skrócenia czasu potrzebnego do wykrycia obecności pasożytów atakujących pszczoły, w tym dręcza pszczelego, a tym samym zmniejszyć liczbę chorujących osobników.

2.2. Podstawy techniczne

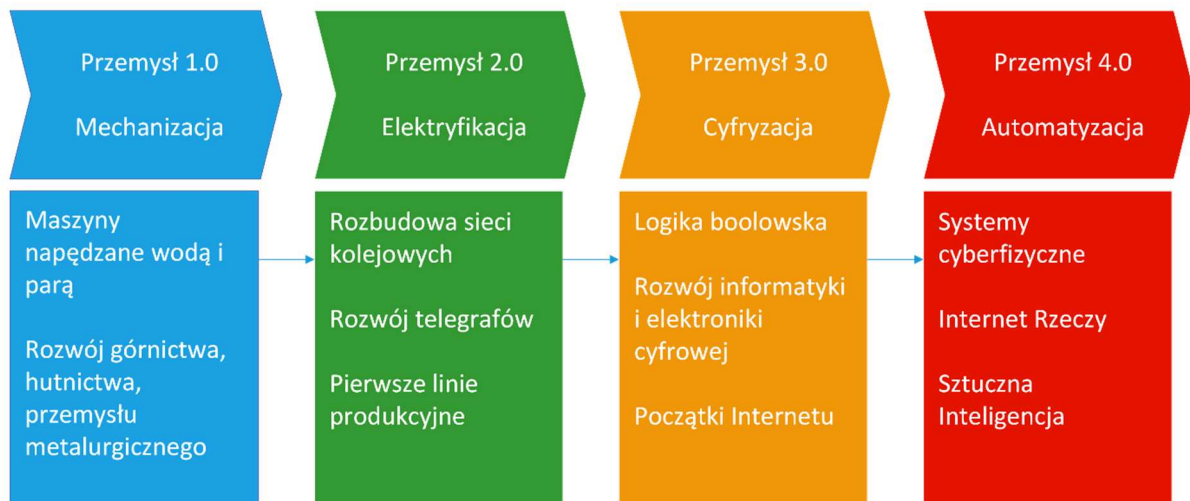
Nowoczesne technologie są stosowane powszechnie w coraz większej skali, zarówno dla użytkowników indywidualnych jak również przedsiębiorstw i fabryk przemysłowych. Jest to związane z rozwojem elektroniki, który umożliwił trzecią rewolucję przemysłową i dał podstawy do zaistnienia w nauce takich pojęć jak *Przemysł 4.0*, Internet Rzeczy, systemy cyber-fizyczne, uczenie maszynowe czy systemy przetwarzania brzegowego, które zostały szerzej opisane w tym rozdziale.

2.2.1. Czwarta rewolucja przemysłowa

Od pradawnych czasów ludzie starali się uprościć sobie życie. W starożytności ważnym odkryciem, które wprowadziło wiele innych, było koło. Starożytni Rzymianie są znani z utworzenia dobrze działającej sieci wodociągowej i dróg, Egipcjanie – z piramid, Chińczycy – z wynalezienia papieru. W baroku powstały zegarki kieszonkowe, wynaleziono mikroskop, nastąpił duży rozwój nauk ścisłych i przyrodniczych. Przełomem okazało się zastąpienie rzemieślniczych metod produkcji maszynami napędzanym parą wodną i wodą. Spowodowało to gwałtowny rozwój wielu gałęzi przemysłu – górnictwa, rolnictwa, przemysłu tekstylnego czy metalurgicznego. Okres ten, przypadający na XVIII i XIX wiek (w zależności od kraju i kontynentu) jest opisywany jako pierwsza rewolucja przemysłowa (ang. *First Industrial Revolution*). *Wiek pary* – jak również określa się powyższy czas, został następnie zastąpiony drugą, technologiczną, rewolucją przemysłową. Rozwój telegrafów, sieci połączeń kolejowych oraz powszechne wykorzystanie energii elektrycznej w fabrykach umożliwiło produkcję na skalę masową. Przełom XIX i XX wieku był też czasem ważnych odkryć naukowych – wynaleziono żarówkę, zdefiniowano proces produkcji stali oraz chemiczne procesy wytwarzania papieru. Wraz z początkiem XX wieku nastąpił czas wojen, które spowolniły rozwój przemysłu. Po ich zakończeniu rozpoczęła się trzecia rewolucja przemysłowa, nazywana też erą digitalizacji. Zastosowanie logiki Boole'a w komputerze

Z1 zapoczątkowało szybki rozwój w dziedzinie informatyki i elektroniki cyfrowej, który trwa do dzisiaj. Tranzystory i układy scalone umożliwiły powstawanie komputerów, mikroprocesorów oraz telefonów komórkowych. Wprowadzenie Internetu do powszechnego użycia umożliwiło szybką wymianę informacji i przyspieszyło proces globalizacji. Obecnie, 99% światowych danych jest przechowywanych cyfrowo, a blisko 70% populacji ma dostęp do Internetu [47]. Aktualną, Czwartą Rewolucję Przemysłową, rozpoczął niemiecki projekt *Industrie 4.0*, zaprezentowany w 2011 roku. Pomysł ten zakładał unowocześnienie niemieckiego przemysłu i opierał się o wykorzystanie najnowszych technologii, takich jak m.in. systemy cyberfizyczne, Internet Rzeczy (ale też Internet Usług, Internet Ludzi, Internet Wszystkiego), przetwarzanie brzegowe oraz sztuczna inteligencja, które zostaną opisane bardziej szczegółowo w kolejnych częściach tego rozdziału.

W 2015 roku Klaus Schwab, założyciel Światowego Forum Ekonomicznego (ang. *World Economic Forum, WEF*), w artykule opublikowanym przez amerykańskie czasopismo *Foreign Affairs* przedstawił ideę *Przemysłu 4.0* większej liczbie odbiorców. Rok później temat czwartej rewolucji przemysłowej był motywem przewodnim corocznym spotkaniu Światowego Forum Ekonomicznego w Davos. Od tego czasu zagadnienie to jest przedmiotem wielu publikacji naukowych oraz projektów komercyjnych. Na poniższym *Rys. 2* zaprezentowane zostały najważniejsze informacje dotyczące poszczególnych rewolucji.



Rys. 2 Cztery rewolucje przemysłowe

Wśród najważniejszych pojęć *Przemysłu 4.0*, które należy wziąć pod uwagę przy projektowaniu systemów, wymienia się [48], [49]:

- Interoperacyjność/współoperacyjność (ang. *Interoperability*), czyli zdolność wielu czujników, urządzeń, maszyn i ludzi do wzajemnej wymiany informacji przez Internet. Kluczową rolę odgrywa tutaj bezprzewodowe połączenie internetowe, dzięki niemu zapewniona jest łączność z każdym komponentem systemu. Literatura opisuje w tym zakresie trzy koncepcje – Internet Rzeczy (IoT), Internet Ludzi (ang. *Internet of People, IoP*) oraz Internet Wszystkiego (ang. *Internet of Everything, IoE*), opisujące sieci połączonych ze sobą elementów systemu;
- transparentność informacji (ang. *Information transparency*) przekazywanych między komponentami – informacje powinny być dostępne dla każdego elementu systemu. Dzięki temu system będzie mógł podejmować pożądane w danym momencie decyzje, a jego operatorom umożliwi rozwijanie systemu poprzez wdrażanie usprawnień oraz nowych funkcjonalności;
- decentralizację decyzji (ang. *Decentralized decisions*) – decyzje dotyczące systemu mogą być podejmowane zarówno wewnątrz, jak i na zewnątrz fabryki/zakładu produkcyjnego. Transparentność informacji oraz współoperacyjność zapewniają operatorom rzetelne dane dotyczące działania systemu w konkretnym momencie, dzięki czemu proces podejmowania decyzji nie jest ograniczony do czasu i miejsca, a do aktualnych potrzeb;
- wsparcie techniczne (ang. *Technical assistance*) – system powinien wspierać operatora w przypadku awarii, dostarczając mu potrzebnych informacji, które umożliwią rozwiązanie problemu w krótkim czasie;
- zdolność do działania w czasie rzeczywistym (ang. *real-time capability*) – ciągła analiza danych umożliwi wykrycie (lub nawet zapobieżenie) awarii systemu i szybką reakcję na jej wystąpienie. Takie podejście skraca do minimum czas przestoju w procesie produkcji;
- architektura oparta na usługach (ang. *Service-Oriented Architecture, SOA*) – wzorzec projektowy, w którym jedne komponenty aplikacji dostarczają usługi innym jej częściom poprzez wykorzystanie protokołów komunikacyjnych, najczęściej połączenia sieciowego.

Zastosowanie powyższych wytycznych na etapie tworzenia systemu zaowocuje większą niż w przypadku tradycyjnego podejścia, elastycznością w realizacji potrzeb klienta oraz szybszym dostosowaniem się do wymaganych przez niego zmian, efektywnym skalowaniem zasobów produkcyjnych, a także niższymi kosztami produkcji [49]. Wbrew

niektórym sceptycznym głosem, *Przemysł 4.0*. nie ma na celu pozbawienia ludzi pracy w fabrykach i zastąpienia ich robotami, a jedynie usprawnienie procesów produkcyjnych poprzez wykorzystanie zaawansowanych technologii. Operatorzy maszyn i systemów wciąż będą niezbędnymi pracownikami, ich praca natomiast będzie wspierana przez inteligentne oprogramowanie.

Mimo wielu niewątpliwych zalet, idea *Przemysłu 4.0* nie pozostaje wolna od wad oraz trudności, które firmy mogą napotkać w czasie transformacji zakładów produkcyjnych na tę ścieżkę. Wśród najczęściej wymienianych znajdują się kwestie związane z cyberbezpieczeństwem – zwiększenie liczby połączeń pomiędzy urządzeniami i wymiany danych między nimi jest związane z koniecznością większej ochrony danych wrażliwych organizacji. Wprowadzenie dodatkowych zabezpieczeń, takich jak szyfrowanie przesyłanych informacji może wiązać się z większymi kosztami poszczególnych komponentów oraz regularnymi aktualizacjami sterowników. Co więcej, może okazać się, że mimo podjętych działań, mających zapewnić bezpieczeństwo danych i urządzeń, dojdzie do cyberataku [50]. Inną kwestią jest konieczność doszkolenia pracowników bądź konieczność zatrudnienia inżynierów pracujących z wykorzystaniem nowych technologii, którzy będą potrafili obsługiwać systemy. W związku ze zmianą procesu produkcji konieczna będzie również zmiana sposobu zarządzania [51]. Nie bez znaczenia pozostają też zagadnienia związane z przechowywaniem i zarządzaniem danymi – utworzenie spójnych baz danych, z łatwym dostępem do potrzebnych danych może stanowić wyzwanie. Trudności te, w połączeniu z niedoborem specjalistów na rynku pracy sprawiają, że niektóre firmy mają obawy związane z przekształceniem procesu produkcji zgodnie z ideą *Przemysłu 4.0*.

2.2.2. Internet Rzeczy

Internet Rzeczy (ang. *Internet of Things*, IoT) jednym z filarów *Przemysłu 4.0*. Zakłada on połączenie ze sobą wielu czujników oraz urządzeń i wymianę informacji między nimi za pomocą sieci Internet, bez konieczności zaangażowania w nią człowieka. Nazwa Internet Rzeczy została użyta po raz pierwszy w 1999 roku przez brytyjskiego inżyniera Kevina Ashtona [52] i opisywała sieć łączącą obiekty świata fizycznego z Internetem. Koncepcja komunikujących się ze sobą maszyn posiada dłuższą historię – już pod koniec lat 70 XX wieku dostępne były systemy zdalnego odczytu wartości na licznikach

energetycznych, w których dane przesyłane były za pomocą sieci telefonicznej. W latach 90 technologia bezprzewodowa umożliwiła rozwój systemów typu M2M (ang. *Machine To Machine*) na większą skalę. W tego typu systemach stosowano zazwyczaj dedykowane, zamknięte sieci (w systemach przemysłowych dodatkowo oparte o odpowiednie standardy), a nie sieci oparte o protokoły IP. Wraz z rozwojem technologii i spadkiem cen urządzeń elektronicznych nastąpił rozwój urządzeń Internetu Rzeczy.

Obecnie, z urządzeń *Internetu rzeczy* korzystają odbiorcy indywidualni, przedsiębiorstwa, a nawet władze (np. sterowanie ruchem drogowym w miastach) [53]. Urządzenia te pomagają w zapewnieniu bezpieczeństwa i zapobieganiu nieszczęśliwym wypadkom na drogach, terenach objętych budowami, w halach przemysłowych. Są szeroko wykorzystywane w medycynie i ratownictwie medycznym – umożliwiają m.in. zdalne monitorowanie stanu zdrowia pacjentów przewlekle chorych oraz wspierają osoby z niepełnosprawnościami. Firmy korzystają z urządzeń IoT do sprawnego i ekonomicznego zarządzania flotą i łańcuchem dostaw. Systemy M2M są obecne w górnictwie – umożliwiają efektywne wydobycie zasobów i ich transport; w rolnictwie – do zarządzania procesem nawadniania oraz jako wsparcie hodowli bydła. W końcu, ułatwiają życie milionom użytkowników indywidualnym, pomagając im planować najkrótszą trasę powrotu do domu, podpowiadając listy zakupowe, przypominając o nadchodzących terminach płatności za rachunki, przysyłając do dostawców mediów stany liczników i zapewniając rozrywkę na wielu płaszczyznach.

Uwzględniając użytkownika końcowego wśród urządzeń IoT można wyróżnić kategorię *Przemysłowego Internetu Rzeczy* (ang. *Industrial Internet of Things*, IIoT). Jej głównym celem jest połączenie komponentów przemysłowych, takich jak urządzenia i systemy kontroli z systemami informacyjnymi i procesami biznesowymi [54]. Dzięki temu możliwe jest pozyskiwanie i przetwarzanie dużych liczb danych z wielu urządzeń, a w konsekwencji optymalizacja procesów produkcyjnych, która umożliwi szybką adaptację do zmieniających się wymagań klienta. Tym, co odróżnia standardowe urządzenia Internetu Rzeczy dedykowane dla indywidualnego odbiorcy od urządzeń przemysłowego IoT jest przede wszystkim liczba przetwarzanych danych oraz krytyczność działania systemu. W przypadku IIoT liczba danych gromadzonych i przesyłanych w obrębie systemu są ogromne, a awaria jednego z węzłów może wiązać się z opóźnieniami w produkcji i dużymi stratami finansowymi. Jeśli uszkodzeniu ulegnie komponent w systemie wspierającym medycynę, bezpieczeństwo ruchu drogowego lub fabryki, zagrożone może być zdrowie lub nawet życie człowieka. Ścisłej kontroli podlegają też urządzenia, które mogą zostać włączone

do sieci, co nie ma miejsca w przypadku urządzeń IoT dedykowanych odbiorom indywidualnym. Główne podobieństwa i różnice zostały przedstawione w Tab. 1.

Tab. 1 Porównanie urządzeń IoT i IIoT

	IOT	IIOT
UŻYTKOWNIK KOŃCOWY	Indywidualny	Firma/przedsiębiorstwo produkcyjne/fabryka/podmioty medyczne
RODZAJ USŁUG	Zorientowane na człowieka, interakcje człowiek - urządzenie	Zorientowane na urządzenia, interakcje urządzenie - urządzenie
KRYTYCZNOŚĆ SYSTEMU	Niska	Wysoka
LICZBA PRZETWARZANYCH DANYCH	Średnia / duża	Duża / bardzo duża
POŁĄCZENIE MIĘDZY WĘZŁAMI	Nieustrukturalizowane, węzły są mobilne	Zachowana struktura, węzły są stałe i zarządzane
DANE WRAŻLIWE	Dotyczą osoby prywatnej	Dotyczą przedsiębiorstwa, uwzględniają szczegóły procesów produkcyjnych, mogą obejmować dane medyczne
POZIOM SKOMPIKOWANIA	Niewielki, kilka/kilkanaście urządzeń lub czujników współpracujących ze sobą	Duży, wiele komponentów wymieniających ze sobą dane
KOSZTY	Relatywnie niskie	Wysokie

Wykorzystanie idei *Internetu rzeczy* (włączając w to *Przemysłowy Internet Rzeczy*) w tak wielu dziedzinach niesie za sobą pewne ograniczenia i wymagania. Jednym z najważniejszych jest kwestia bezpieczeństwa [55]. W przypadku systemów IoT wiele czujników i urządzeń wymienia ze sobą informacje, ponadto poszczególne komponenty

posiadają różne środowiska operacyjne i często ograniczone moce obliczeniowe. Taka sytuacja może prowadzić do zagrożenia w postaci wycieku danych do nieuprawnionego lub niepożądanego odbiorcy. Aby temu zapobiec, projektując system IoT należy uwzględnić następujące elementy [56]:

- prywatność danych (ang. *data privacy*) – użytkownik powinien mieć kontrolę nad tym jakie dane, kiedy i gdzie są wykorzystywane, i w jakim zakresie są przekazywane innym urządzeniom lub podmiotom,
- zaufanie (ang. *trust*) – zastosowanie odpowiednich polityk bezpieczeństwa i certyfikatów ma na celu weryfikację czy urządzenie, z którym następuje komunikacja jest zaufanym odbiorcą,
- identyfikacja i uwierzytelnianie (ang. *identification and authentication*) – urządzenia i połączenia powinny być oznaczone identyfikatorem, a wraz z posiadaniem odpowiedniego klucza (hasła, kodu PIN itd.) mogą otrzymać dostęp do danych,
- kontrola dostępu (ang. *access control*) – ma za zadanie określenie, który komponent posiada dostęp do określonych danych i w jakim stopniu może z nich korzystać.

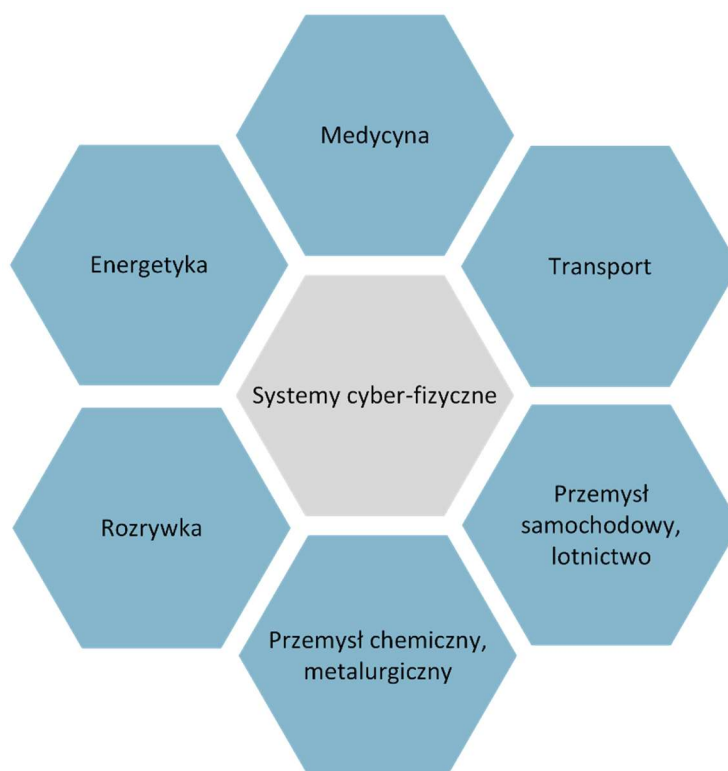
Innym, powiązaniem z bezpieczeństwem, wyzwaniem przy wdrażaniu systemów IoT jest konieczność aktualizacji oprogramowania. Nieaktualne oprogramowanie zwiększa podatność urządzenia na cyberataki. Wśród wymagań technicznych, które urządzenia IoT powinny spełniać ważna jest wydajność energetyczna i długość życia baterii zasilającej. Ograniczenie zużycia energii, konieczności częstego ładowania wraz z perspektywą długiego użytkowania urządzenia zachęci potencjalnego klienta do jego zakupu [53]. Większość urządzeń IoT powinna działać w czasie rzeczywistym (systemy medyczne, przemysłowe, sterowania ruchem drogowym) lub w czasie zbliżonym do rzeczywistego (np. urządzenia domowe). Aby to osiągnąć i nie spowodować przeciążenia sieci należy dobrze zarządzać przetwarzanymi danymi i unikać przesyłania danych nadmiarowych. Kwestie te, chociaż wymagają nakładów pracy na etapie projektowania, wpływają na jakość produktu końcowego oraz poziom zadowolenia użytkownika i powinny być uwzględniane w procesie tworzenia nowych urządzeń Internetu Rzeczy.

2.2.3. Systemy cyber-fizyczne

Systemy cyber-fizyczne (ang. *Cyber – Physical Systems, CPS*) stanowią, obok urządzeń typu IoT, podstawę koncepcji Przemysłu 4.0. Są one połączeniem obiektów fizycznych, występujących w świecie rzeczywistym z systemami cyfrowymi, które pełnią funkcję ich opisu [53]. Część cyfrowa jest odpowiedzialna za sterowanie systemem, umożliwienie połączenia sieciowego i wykonywanie obliczeń potrzebnych do funkcjonowania całego systemu przemysłowego. Ponadto, często stanowi wirtualną kopię fizycznego systemu wraz z jego aktualnymi danymi (tzw. cyfrowy bliźniak, ang. *digital twin*). Fizyczna część ma na celu przeprowadzanie procesów wytwórczych i automatyzację produkcji [57]. Obie części wymieniają między sobą dane, konieczne do zapewnienia ciągłości pracy systemu. Systemy cyber-fizyczne cechuje dostęp każdego komponentu systemu do sieci, wykonywanie obliczeń w czasie rzeczywistym oraz rozproszenie poszczególnych komponentów. Ponadto, systemy te powinny być niezawodne, łatwe do ponownej konfiguracji, a w niektórych obszarach (np. branża automotive, medyczna, energetyczna) posiadać stosowne certyfikaty [58].

Systemy cyber-fizyczne stanowią wsparcie w wielu dziedzinach – m.in. w przemyśle samochodowym, lotniczym, chemicznym, a nawet rozrywce. Bardzo duże zapotrzebowanie na systemy cyber-fizyczne jest także w medycynie – inteligentne sale operacyjne i zabiegowe, systemy wspierające oddziały intensywnej terapii, chirurgia obrazowa, protezy kończyn, neurobiologia, to tylko niektóre przykłady zastosowania najnowszych technologii w tej dziedzinie [58]. Główne branże, w których wykorzystywane są systemy CPS przedstawione zostały na Rys. 3.

Poza typowymi dla systemów cyber-fizycznych cechami, systemy CPS dedykowane branży medycznej powinny zapewniać maksymalne bezpieczeństwo danych (informacje o stanie zdrowia należą do danych szczególnie wrażliwych), przy jednoczesnej współpracy z innymi urządzeniami i czujnikami [59]. Ponadto, działanie tego typu systemów powinno być osadzone w kontekście (ang. *context awareness*) – system musi poprawnie identyfikować odczyty z czujników i analizować je z uwzględnieniem innych danych dotyczących danej osoby w taki sposób, by w razie zagrożenia życia pacjenta poprawnie poinformować o pogorszeniu stanu zdrowia.



Rys. 3 Zastosowanie systemów cyber-fizycznych

Niezależnie od przeznaczenia, systemy CPS muszą spełniać wymagania dotyczące bezpieczeństwa. Operując na różnych rodzajach danych wrażliwych (medycznych, energetycznych, związanych z procesami produkcyjnymi, ruchem drogowym czy lotniczym), są narażone na ataki hackerskie [50], co jest niewątpliwie jednym z największych zagrożeń dla systemów CPS. Konieczność współpracy oraz wymiany informacji z wieloma urządzeniami i czujnikami w ramach jednego systemu zawęża możliwości odpowiedniego zabezpieczenia kanałów komunikacyjnych, przez co rośnie podatność na różnego rodzaju wycieki danych. Częstą praktyką stosowaną przez hackerów jest wstrzykiwanie do systemu złośliwego oprogramowania w postaci wirusów i tzw. robaków. Dzięki temu możliwe jest przejęcie przez atakującego kontroli nad działaniem systemu i np. zatrzymanie linii produkcyjnej (co spowoduje bardzo duże straty finansowe) czy uzyskanie poufnych informacji dotyczących procesu produkcji. Tego typu ataki zdarzają się nawet w dużych firmach, są jednak niechętnie ujawniane opinii publicznej. Jednym z najbardziej znanych ataków hackerskich na system cyber-fizyczny dotyczył irańskiego programu nuklearnego. W ramach tego ataku, robak *Stuxnet* został wstrzyknięty do systemu SCADA (ang. *Supervisory Control And Data Acquisition*) i zmienił ustawienia sterowników PLC (ang. *Programmable Logic Controller*), powodując problemy z wirówkami używanymi do

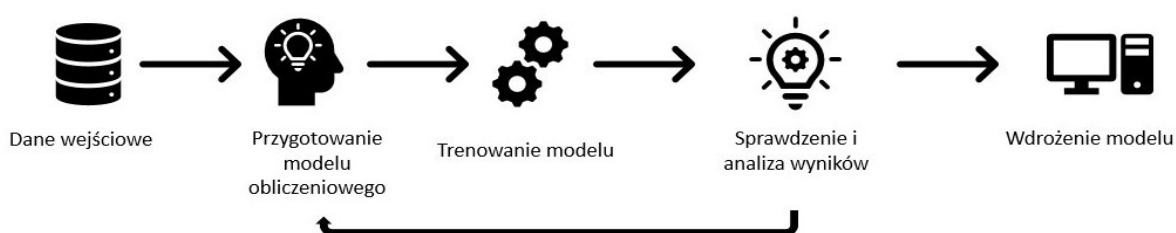
wzbogacania uranu [60]. Aby zmniejszyć ryzyko wystąpienia ataku należy regularnie dokonywać aktualizacji sterowników, uwzględnić kwestie związane z bezpieczeństwem już na etapie projektowania systemu, edukować pracowników o możliwych zagrożeniach, a także przeprowadzać kontrole, mające na celu jak najszybszą identyfikację i eliminację złośliwego oprogramowania, jeśli takie zostanie wykryte w systemie.

2.2.4. Sztuczna inteligencja, uczenie maszynowe, uczenie głębokie

Termin *sztuczna inteligencja* (ang. *Artificial Intelligence, AI*) nie posiada jednoznacznej definicji [61] i w rozważaniach technicznych dotyczy wielu zagadnień związanych przetwarzaniem danych przez komputery w sposób zbliżony do ludzkiej inteligencji. Na potrzeby niniejszej rozprawy można przyjąć, że sztuczna inteligencja stanowi dział nauki w dziedzinie informatyki i kogniistyki, którego badania koncentrują się na możliwie jak najlepszym odwzorowaniu poszczególnych funkcji umysłu i zmysłów na maszynach i w programach komputerowych. Wśród problemów podejmowanych w ramach tej dziedziny naukowej wyróżnia się między innymi:

- podejmowanie decyzji oraz rozwiązywanie problemów w warunkach niewystarczającej lub niekompletnej ilości informacji (ang. *reasoning, problem-solving*),
- przetwarzanie języka naturalnego (ang. *natural language processing, NLP*) – rozpoznawanie i analiza języka ludzkiego w formie mówionej i pisanej,
- planowanie i prognozowanie, w jaki sposób podjęte przez system działania wpłyną na świat rzeczywisty,
- uczenie maszynowe (ang. *machine learning, ML*),
- rozpoznawanie obiektów i dźwięków na podstawie danych zgromadzonych z różnych czujników i wnioskowanie o rzeczywistości na tej podstawie (np. zbieranie danych dotyczących położenia samochodu i obiektów wokół niego i informowanie kierowcy o potencjalnych zagrożeniach),
- komputerowe gry logiczne,
- inteligentne roboty, stosowane np. w fabrykach czy medycynie,
- inteligencja społeczna – rozpoznawanie i interpretacja emocji i samopoczucia ludzi.

Uczenie maszynowe stanowi część sztucznej inteligencji, która obejmuje algorytmy poprawiające swoje działanie („uczące się”) w oparciu o wcześniejsze wyniki. Tego typu algorytmy działają w oparciu o następujący schemat – najpierw budowany jest odpowiedni model obliczeniowy; następnie jest on walidowany z wykorzystaniem testowego zbioru danych; na podstawie wyników są wyciągane odpowiednie wnioski, które pomagają ulepszyć model obliczeniowy; obliczenia i analiza wyników wraz z ulepszaniem modelu są powtarzane aż do osiągnięcia przez model zakładanej jakości. Proces uczenia maszynowego został pokazany na Rys. 4.



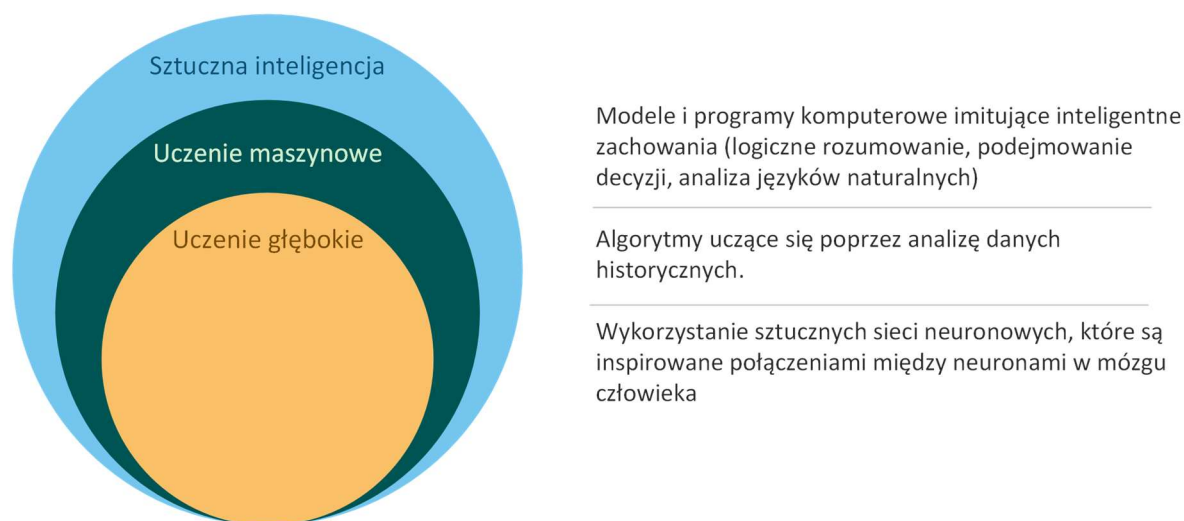
Rys. 4 Proces uczenia maszynowego

Wśród metod uczenia maszynowego można wyróżnić cztery podejścia, wymienione w kolejności od najczęściej do najrzadziej wykorzystywanych [62]:

- uczenie z nadzorem (ang. *supervised*) – w zbiorze danych wejściowych, służącym do wytrenowania modelu znajdują się obiekty z przyporządkowanymi klasami wyjściowymi, dzięki czemu algorytm ma przykłady poprawnie sklasyfikowanych danych (np. historyczne) i na tych przykładach uczy się przyporządkowywać obiekty do konkretnych klas;
- bez nadzoru (ang. *unsupervised*) – w zbiorze danych wejściowych nie znajdują się poprawnie opisane dane, a algorytm musi rozpoznać i sklasyfikować obiekty na podstawie znalezionych między nimi podobieństw;
- z częściowym nadzorem (ang. *semi-supervised*) – stanowi połączenie dwóch powyższych metod, w zbiorze danych wejściowych znajdują się zarówno poprawnie opisane obiekty, jak również takie, które nie mają przypisanych etykiet. Podejście to stosowane jest np. w sytuacji, gdy wykorzystanie w zbiorze danych wejściowych wyłącznie opisanych obiektów jest bardzo kosztowne;
- uczenie przez wzmacnianie (ang. *reinforcement learning*) – algorytm, stosując metodę prób i błędów, podejmuje różne działania mające na celu uzyskanie nagrody

w określonym czasie. Celem działania algorytmu jest wypracowanie odpowiedniej polityki podejmowania decyzji. W uczeniu przez wzmacnianie wyróżnia się następujące elementy – agenta, który podejmuje decyzje, środowisko, z którym agent wchodzi w interakcję i które przydziela agentowi nagrodę, oraz bufor, gdzie przechowywane są zgromadzone przez agenta w czasie podejmowanych interakcji informacje. Zgromadzone dane są następnie wykorzystywane w procesie uczenia się agenta. Podejście to jest stosowane w grach komputerowych, nawigacji i robotyce.

Z tematyką uczenia maszynowego związane jest także uczenie głębokie (ang. *deep learning*). Koncepcja uczenia głębokiego stanowi podzbiór uczenia maszynowego, a relację pomiędzy uczeniem głębokim, uczeniem maszynowym i sztuczną inteligencją obrazuje Rys. 5.



Rys. 5 Sztuczna inteligencja i jej dziedziny

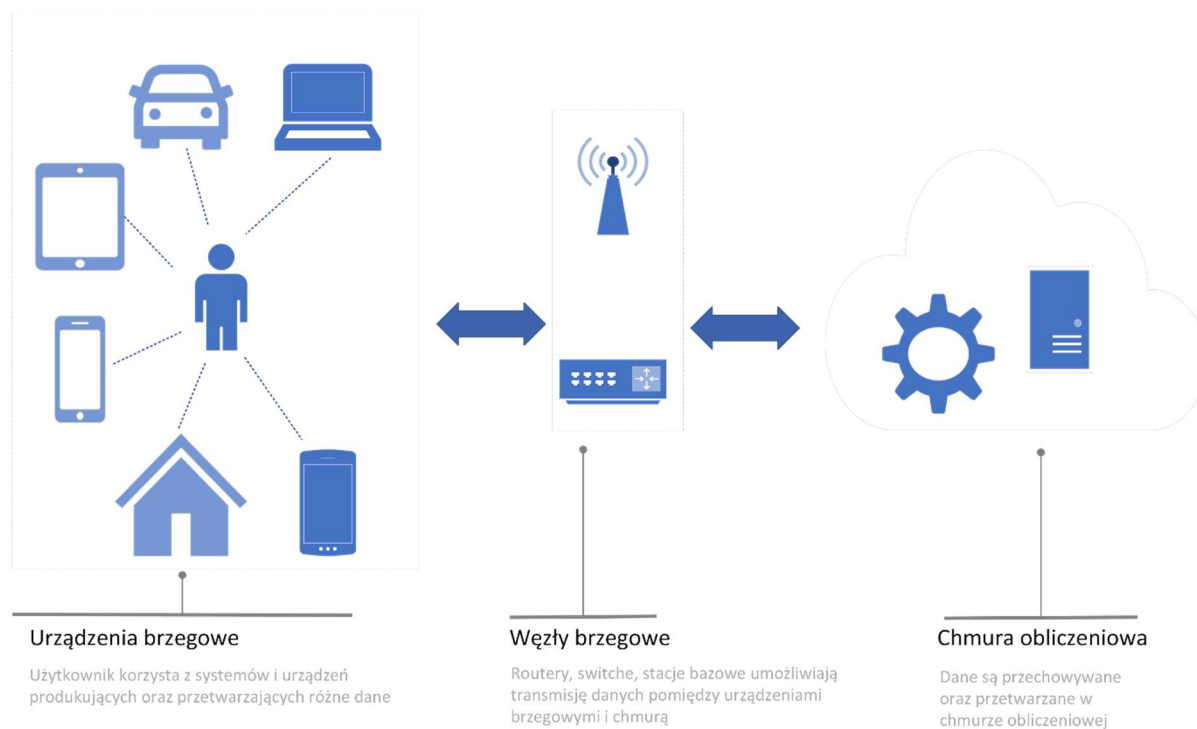
Uczenie głębokie do przetwarzania danych wykorzystuje sztuczne sieci neuronowe, które są wzorowane (częściowo) na rzeczywistych, biologicznych połączeniach między neuronami. Każda z sieci posiada wiele danych wejściowych, wyjściowych i ukrytych warstw. Dane uzyskane w wyniku wykonywania obliczeń na jednej warstwie, stanowią dane wejściowe dla kolejnej warstwy obliczeniowej. Dzięki takiej strukturze, algorytm uczy się wykorzystując swoje własne dane, a z każdą kolejną warstwą uzyskuje się nowe, bardziej szczegółowe cechy analizowanych obiektów. Algorytmy uczenia głębokiego potrzebują dużych zestawów danych wejściowych oraz mocnych jednostek obliczeniowych. Zastosowanie procesorów graficznych (ang. *Graphics Processing Unit*, GPU) zamiast tradycyjnych jednostek obliczeniowych umożliwia efektywne wykonywanie obliczeń. Procesy uczenia głębokiego, podobnie jak w uczeniu maszynowym, mogą być nadzorowane

częściowo, całkowicie lub nie posiadać zestawu danych do weryfikacji działania modelu. Algorytmy głębokiego uczenia mają zastosowanie zarówno w celach naukowych – m.in. do analizy obrazów (również w medycynie), rozpoznawania dźwięków, przetwarzania języków naturalnych, w bioinformatyce, jak i komercyjnych – do zarządzania relacjami z klientami, dopasowywania treści reklam na urządzeniach mobilnych czy treści w aplikacjach oferujących usługi rozrywkowe (serwisy z filmami, muzyką itp.) do preferencji użytkownika.

2.2.5. Przetwarzanie brzegowe

Używane niemal w każdej dziedzinie życia urządzenia IoT i systemy cyber-fizyczne w każdej sekundzie produkują gigabajty nowych danych. Operacje wykonywane w chmurze obliczeniowej (ang. *cloud computing*), chociaż bardzo wydajne, w przypadku tak dużych ilości danych posiadają pewne ograniczenia. Przede wszystkim jest to związane z przepustowością sieci i czasem przesyłania danych [63]. Przykładowo, samochód autonomiczny produkuje 1 GB danych na sekundę, a na podstawie tych danych podejmuje decyzje o swoim ruchu na drodze. Potrzebuje więc odpowiedzi w czasie rzeczywistym, a od jego „zachowania” zależy bezpieczeństwo innych uczestników ruchu drogowego. Przesyłanie informacji takich rozmiarów i oczekiwanie na odpowiedź serwera umieszczonego w chmurze nie sprawdzi się w tym przypadku. W odpowiedzi na tę oraz podobne do niej sytuacje powstał paradygmat przetwarzania brzegowego (ang. *edge computing*). Zgodnie z jego założeniami, dane pobierane z czujników, zamiast być przesyłane i przetwarzane w chmurze, są poddawane analizie na urządzeniach zlokalizowanych na końcu (brzegu) sieci, „bliżej” czujników – najczęściej są to urządzenia, z którymi te czujniki są bezpośrednio połączone [64]. Przykładowo, urządzeniem brzegowym dla czujnika monitorującego pracę serca będzie smartfon, podobnie jak dla kamery, której użytkownik używa podczas korzystania z nawigacji w telefonie. W przeciwieństwie do modelu obliczeń chmurowych, przy przetwarzaniu brzegowym do chmury przesyłane są dane już po wykonaniu potrzebnych obliczeń. Część danych zostaje zachowana w pamięci urządzenia, a dalej są przekazywane tylko niektóre, często przefiltrowane według odpowiednich kryteriów informacji. W chmurze dane są przechowywane i analizowane, a uzyskane rezultaty przekazywane odbiorcy, jeśli na nie czeka. Takie podejście znacznie skraca czas oczekiwania na wyniki przetwarzania danych, nie powoduje zwiększonego ruchu sieciowego, ogranicza zużycie energii i ma wpływ na efektywność działania systemu [65]. Ponadto, zmniejszenie ilości informacji przesyłanych do chmury, ogranicza ryzyko awarii związanej z przeciążeniem sieci, dzięki czemu system jest

bardziej niezawodny. Zwiększa się też bezpieczeństwo systemu – część danych wrażliwych jest przechowywana na urządzeniu, które, jeśli będzie odpowiednio zabezpieczone, ograniczy ryzyko udostępnienia ich niepożądanym odbiorcom. Paradygmat brzegowego przetwarzania danych ilustruje Rys. 6.



Rys. 6 Przetwarzanie brzegowe

Do aktualnych i potencjalnych, przykładowych zastosowań przetwarzania brzegowego można zaliczyć praktycznie wszystkie nowoczesne systemy bazujące na przetwarzaniu obrazu – czujniki samochodowe, aplikacje oferujące usługi nawigacji, systemy miejskie służące do zarządzania ruchem drogowym i mające na celu zapewnienie bezpieczeństwa mieszkańcom poprzez całodobowy monitoring. Przetwarzanie brzegowe sprawdzi się także w systemach generujące duże liczby danych w krótkim czasie – systemy lotnicze, stacje ładowania samochodów elektrycznych, systemy utrzymania i monitorowania procesów w fabrykach czy platformy sprzedażowe. Urządzenia brzegowe mogą też być używane w energetyce – np. do sterowania ustawieniem paneli słonecznych w oparciu o dane pogodowe pozyskane z czujników. Mniej oczywistym przykładem będzie zastosowanie tej technologii w rolnictwie – rolnicy pracujący w modelu rolnictwa precyzyjnego (ang. *Precision Agriculture*) [66] mogą korzystać z przetwarzania brzegowego do śledzenia poziomu nawodnienia upraw, występowania składników odżywczych w glebie, poziomu

nasłonecznienia terenu czy temperatury i w oparciu o te dane lepiej zarządzać procesem produkcji plonów.

Mimo niewątpliwych zalet przetwarzania brzegowego, zdarzają się też trudności, dlatego projektując systemy i urządzenia wykorzystujące ten model należy uwzględnić następujące kwestie [63]:

- heterogeniczność platform – w przeciwieństwie do przetwarzania chmurowego, w którym kod jest kompilowany i uruchamiany w jednym, określonym środowisku, przetwarzanie brzegowe opiera się o różne urządzenia. Tworząc oprogramowanie należy zapewnić jego prawidłowe działanie w różnych systemach operacyjnych, co wymaga od programisty większych umiejętności i może wydłużyć czas powstawania oprogramowania;
- poziom abstrakcji danych – konieczne jest określenie, które dane z czujników i w jakiej formie powinny być przesyłane do chmury, które filtrowane i przetwarzane na urządzeniu brzegowym. Należy też ustalić odpowiedni sposób i format przechowywania danych tak, aby były one łatwo dostępne dla urządzenia brzegowego, przy jednoczesnym odpowiednim poziomie szczegółowości danych;
- zarządzanie usługami w sieci urządzeń brzegowych – aby zagwarantować niezawodność systemu należy zadbać przede wszystkim o priorytety poszczególnych operacji, umożliwić łatwą wymianę elementów systemu lub dodanie nowych (np. w przypadku zużycia czujnika lub dołączenia do sieci nowego urządzenia brzegowego), izolację procesów w taki sposób, by w przypadku awarii jednego minimalizować ryzyko awarii całego systemu, a także izolację danych od aplikacji zewnętrznych;
- zarządzanie zużyciem energii poprzez dopasowanie liczby i częstotliwości interakcji z chmurą obliczeniową i operacjami wykonywanymi na urządzeniu brzegowym, tak aby zapewnić jak najdłuższy czas pracy baterii urządzenia przy zachowaniu wymaganej jakości przetwarzania danych;
- zminimalizowanie czasu oczekiwania na odpowiedź z chmury obliczeniowej oraz zadbanie o odpowiednią przepustowość sieci, tak aby zminimalizować liczbę sytuacji przeciążeń sieci.

Uwzględnienie powyższych aspektów na początkowych etapach projektowania pomoże zmniejszyć koszty przy jednoczesnym zapewnieniu dużej efektywności systemu.

3. Problematyka pracy na tle światowej literatury naukowej

Podobnie jak w przypadku innych nauk przyrodniczych, zagadnienia związane z pszczelarstwem są opisywane w literaturze naukowej na całym świecie. Tematyka monitorowania parametrów dotyczących uli oraz rodzin pszczelich pojawia się w naukowych opracowaniach od ponad wieku, a najnowsze prace opisują rozwiązania oparte o nowoczesne urządzenia i algorytmy. Celem niniejszego rozdziału jest przedstawienie, w jaki sposób zmieniały się sposoby monitorowania pszczół na przestrzeni lat oraz jakie tendencje można zaobserwować wśród najnowszych systemów dla pszczelarzy rozwijanych przez naukowców.

3.1. Monitorowanie pszczół w przeszłości

Historia pszczelarstwa sięga starożytności – malowidła w jaskiniach dotyczące zbierania przez ludzi miodu od dzikich pszczół powstały ok 10 000 lat temu, w III wieku p.n.e. o pszczołach pisał Arystoteles, a w I wieku p.n.e. Wergiliusz przedstawił ówczesne pszczelarstwo. W XVII wieku został wydany pierwszy polski podręcznik opisujący zasady hodowli pszczół zatytułowany „Nauka koło pasiek” autorstwa Walentego Kąckiego. W XVIII wieku szwedzki naukowiec Karol Linneusz dokonał klasyfikacji pszczół, wyodrębniając m.in. gatunek pszczoły miodnej *Apis Mellifera*, która jest hodowana w celu pozyskiwania miodu. Rozwój współczesnego pszczelarstwa nastąpił w XIX wieku, kiedy zaczęto konstruować ule z ruchomymi ramkami, co ułatwiło prowadzenie obserwacji pszczół.

Pierwsze próby obserwacji rodzin pszczelich miały miejsce już w XVIII wieku, natomiast jedna z pierwszych opisanych metod monitorowania warunków panujących w ulu pochodzi z 1914 roku i dotyczy zbierania pomiarów temperatury i wagi ula [67]. Celem obserwacji było ustalenie, w jaki sposób zwiększyć szansę pszczelej rodziny na przetrwanie okresu zimowego. Burton Gates przez 11 miesięcy dokonywał pomiarów co godzinę,

zapisując odczyty z siedmiu termometrów umieszczonych w różnych miejscach ula oraz jego wagę. W latach 30 XX wieku W. Dunham [68] wykorzystał metodę termopary elektrycznej do wykonywania pomiarów wewnątrz ula. Pomiarów również były dokonywane co godzinę, jednak czas trwania badań był dużo krótszy. W tym okresie podejmowano także pierwsze próby automatyzacji pomiarów – poprzez nagrywanie impulsów elektrycznych [69] oraz użycie sygnałów fotoelektrycznych do zliczania pszczół wlatujących i opuszających ul [70]. W kolejnych latach podejmowano próby ulepszania metody zliczania pszczół zaproponowanej przez W. Brittaina [71], [72]. Wraz z rozwojem elektroniki do monitorowania parametrów panujących w ulu oraz zliczania pszczół wykorzystywano coraz bardziej zaawansowane metody i urządzenia. Przykładowo, w artykule [73] został opisany system umożliwiający zliczanie pszczół i określanie kierunku ich poruszania się, oraz monitorowanie warunków atmosferycznych takich jak temperatura wewnątrz i na zewnątrz ula, poziom nasłonecznienia, wilgotność powietrza, prędkość wiatru, a także wagę ula. System działał w oparciu o mikroprocesor Intel 8088 i oprócz czujników do mierzenia powyższych parametrów, wykorzystywał promieniowanie podczerwone w celu zliczania pszczół przechodzących przez odpowiednio skonstruowane wejście. Promieniowanie podczerwone, wraz z użyciem mikroprocesora zostało wykorzystane do monitorowania aktywności pszczół również w innym systemie [74], z tą różnicą, że zwiększony został kanał wejścia - wyjścia dla pszczół oraz dokładność pomiarów. Było to możliwe dzięki zastosowaniu tylko jednego źródła promieniowania podczerwonego, a nie – jak w rozwiązaniu poprzednim – dwóch.

Za kolejny kamień milowy w historii monitorowania pszczół można uznać wykorzystanie kamer nagrywania filmów wideo wewnątrz i na zewnątrz ula. Jednym z pierwszych tego typu rozwiązań były systemy [75], [76] do śledzenia tzw. *tańca pszczół*, czyli określonej sekwencji ruchów, za pomocą których pszczoły przekazują sobie różne informacje. W 2008 roku zaprezentowany został system umożliwiający monitorowanie i zliczanie pszczół znajdujących się przed wejściem do ula [77]. Powstawanie rozwiązań tego typu doprowadziło do formalnego zdefiniowania terminu precyzyjnego pszczelarstwa (ang. *Precision Beekeeping, Precision Apiculture*) [78]. Rozwój metod i technologii wykorzystywanych do monitorowania pszczół w czasie obrazuje Rys. 7.

Zautomatyzowane monitorowanie uli może pomóc pszczelarzom lepiej zarządzać swoimi pasiekami, jednocześnie czas poświęcany na manualne inspekcje uli i zmniejszając liczbę ingerencji w naturalny rytm życia rodziny pszczelej. Dobrze zaprojektowany i zaimplementowany system umożliwi pszczelarzom wczesne zauważenie zjawisk,

świadczących o występowaniu choroby, osłabieniu rodziny pszczelej spowodowanej atakiem drapieżnika czy małą ilością pożywienia w ulu, czyli rozpoznanie stanu przed rójką, czyli procesem rozmnażania się pszczół polegającym na opuszczeniu ula przez dużą liczbę dorosłych osobników wraz z nową matką i oddaleniu się od dotychczasowego miejsca. Chociaż proces rojenia się jest zupełnie naturalną częścią cyklu życia pszczelej rodziny, w pasiekach jest on niepożądany, ze względu na możliwość utraty dużej liczby pszczół. Odpowiednio wczesne zauważenie sygnałów wskazujących na rójkę w najbliższym czasie, pozwoli pszczelarzom na stłumienie nastroju rojowego pszczół i zapobiegnie utracie wielu osobników, albo da czas na przygotowanie nowego ula i udomowienie nowo wyrojonej rodziny.



Rys. 7 Historia monitorowania pszczół

3.2. Nowoczesne systemy wspierające pszczelarzy

Wykorzystanie nowoczesnych technologii w pszczelarstwie, chociaż nie jest aż tak powszechne jak np. w medycynie, z roku na rok zyskuje coraz większą popularność.

Precyzyjne pszczelarstwo oparte jest przede wszystkim na analizie danych różnych warunków atmosferycznych (najczęściej dotyczy temperatury i wilgotności, ale literatura opisuje też systemy, w których mierzone są parametry takie jak prędkość wiatru czy ciśnienie), wagi ula, obrazu sprzed ula oraz z jego środka i dźwięków wydawanych przez pszczelą rodzinę. Zbierane informacje są poddawane analizie w celu m.in. określenia momentu rojenia się pszczół, kondycji zdrowotnej rodziny, występowania pasożytów, momentu zbioru miodu, a zimą do monitorowania poziomu zapasu pożywienia. Systemy i prototypy systemów dedykowanych dla pszczelarstwa można podzielić według dwóch podejść – ze względu na rodzaj pozyskiwanych i przetwarzanych danych oraz sposób ich przetwarzania.

3.2.1. Klasyfikacja systemów ze względu na rodzaj przetwarzanych danych

Wśród systemów pszczelarstwa precyzyjnego opisywanych w literaturze jednym z najczęściej przedstawianych podejść jest monitorowanie temperatury. Choć wykorzystywane w pszczelarstwie od bardzo dawna, nie traci na popularności – jest relatywnie tanie, a przy tym umożliwia rozpoznanie wielu sytuacji dotyczących rodziny pszczelej. Obserwacja temperatury panującej w ulu pomoże zauważyć niższą niż zazwyczaj aktywność pszczół (związaną np. z chorobą lub niewystarczającą ilością pożywienia), lub przeciwnie – wyższą, która może wskazywać na okres przed tzw. rójką (podziałem rodziny i opuszczeniem ula) czy wychów czerwiu (młodych pszczół). Przykładowo, w systemie opartym o platformę RaspberryPi zaproponowanym przez [79] monitorowano temperaturę w ulach, w celu rozpoznania okresu tuż przed momentem rojenia się pszczół (ang. *pre-swarming*) i odróżnienia go od okresu wychowu pszczół (ang. *brood rearing*). Równocześnie, w celu walidacji uzyskanych wyników, w ciągu dnia ule były obserwowane przez pszczelarza. W czasie czteromiesięcznych obserwacji wykryto dziewięć momentów rojenia się, z czego jeden miał miejsce w godzinach nocnych (ok. godziny przed wschodem słońca) i nie mógł zostać potwierdzony obserwacjami na miejscu, natomiast pozostałe momenty wskazane przez algorytm pokrywały się z rzeczywistymi obserwacjami pszczelarza.

Często, wraz z pomiarem temperatury, systemy korzystają z innych czujników, które odczytują parametry, takie jak waga ula, wilgotność, ciśnienie, stężenie różnych gazów w powietrzu, a bardziej zaawansowane rejestrują także dźwięk i obraz. Na przykład, w 2016 roku [80] zaproponowali rozwiązanie umożliwiające wykonywanie pomiarów temperatury, wilgotności i wagi ula. Czujniki były umieszczone w trzech miejscach ula, dane były

przesyłane bezprzewodowo do komputera przemysłowego, a badania objęły 20 uli w obrębie jednej pasieki. Takie podejście gwarantowało dostęp do danych w czasie rzeczywistym. Prototyp systemu [81] opiera się o wagę zaprojektowaną w celu ważenia ula, ponadto wykorzystuje czujnik temperatury i moduł GSM, za pomocą którego wartości pomiarów mogą być przesyłane na serwer. W systemie Internetu Rzeczy opisanym przez Braga et al. [82] wykorzystano pomiary temperatury wewnątrz i na zewnątrz ula, wilgotności w ulu, wagi ula oraz ciśnienia atmosferycznego, prędkości wiatru i opadów deszczu w celu wczesnego rozpoznawania złej kondycji zdrowotnej rodziny pszczoł; natomiast S. Kontogiannis [83] zaproponował nową architekturę całego ula, z uwzględnieniem czujników temperatury i wilgotności, a także zapewnieniem bezpieczeństwa ula, np. kradzieżą. W przypadku wystąpienia nieodpowiednich warunków termicznych system umożliwia ich regulację za pomocą dedykowanej aplikacji. Temperatura i wilgotność w ulu, wraz z rejestrowaniem aktywności pszczoł są monitorowane za pomocą innego systemu Internetu Rzeczy [84], który umożliwia pszczelarzom dostęp do danych zbieranych przez system poprzez aplikację mobilną i webową w czasie rzeczywistym. Pomiary warunków pogodowych wraz z monitorowaniem stężenia gazów w powietrzu mogą być wykonywane w celu określenia kondycji rodziny pszczoł i konieczności wizyty pszczelarza w pasiece w przypadku wykrycia niepokojących sytuacji. Takie podejście zostało opisane w artykule [85], w którym określono 10 różnych stanów ula i zdefiniowano, w którym przypadku potrzebna jest interwencja pszczelarza. System zaprojektowany został w oparciu o mikrokontroler ATmega1281, który wraz z czujnikami umożliwiał gromadzenie danych. Następnie, dane te zostały poddane analizie, a do klasyfikacji zostały wykorzystane drzewa decyzyjne. Podobny system został zaproponowany przez [86] i umożliwiał zbieranie parametrów takich jak temperatura, wilgotność powietrza, ciśnienie atmosferyczne, poziom jasności otoczenia i stężenie gazów w powietrzu. W odróżnieniu od poprzedniego podejścia, autorzy zaproponowali użycie mikrokontrolera LoPy, a w swojej pracy skoncentrowali się na optymalizacji procesu przesyłania danych do chmury obliczeniowej, przechowywania oraz przetwarzania danych różnego typu w czasie rzeczywistym. Wśród systemów, które oprócz wykorzystywania czujników temperatury, gromadzą też obrazy i filmy wideo można wymienić np. system pomagający wykrywanie momentu rojenia się pszczoł [87]. Obraz pozyskiwany z kamer umieszczonych wewnątrz i na zewnątrz ula ma umożliwić pszczelarzowi weryfikację sytuacji, która, na podstawie pomiarów temperatury i wilgotności, została sklasyfikowana jako rojenie. Kamery są uruchamiane, gdy odczyty z czujników wskazują moment rojenia, co znacznie zmniejsza zużycie energii. Inne, niskobudżetowe

rozwiązanie IoT wykorzystujące pomiary temperatury i wagi ula do rozpoznawania okresu wychowu pszczół zostało przedstawione w artykule [88]. Zastosowanie czujników mierzących parametry pogodowe, wagi oraz czujników ruchu zostało zaproponowane w artykule [89]. Jako mikrokontroler wykorzystany został komputer RaspberryPi, a autorzy prototypu zaimplementowali również wizualne narzędzie (stronę internetową) umożliwiające wykrywanie wśród monitorowanych pszczół stanów odbiegających od normy. Do analizy danych wykorzystane zostały algorytmy uczenia maszynowego, a w dalszych pracach nad systemem planowana jest implementacja modułu używającego czujnika gazu i określanie poziomu zarażenia rodziny pszczelej warrozą.

Analiza dźwięków pochodzących z ula umożliwia rozpoznawanie różnych stanów aktywności pszczół. Przykładowo, rozwiązanie zaproponowane przez [90] umożliwia gromadzenie i przechowywanie próbek dźwięku. Opisany system miał na celu zapewnienie odpowiedniej ilości danych, które na dalszych etapach prac miały posłużyć do analizy powiązań pomiędzy aktywnością pszczół określaną poprzez rodzaje wydawanych dźwięków a momentami rojenia się pszczół. Prototyp systemu zaproponowany kilka lat później przez A. Quandoura [91] również opierał się na analizie sygnałów akustycznych mierzonych wewnątrz ula za pomocą mikrofonu. Jako jednostkę przetwarzającą dane wykorzystano komputer jednopłytkowy Beagleboard, a analiza danych miała prowadzić do rozpoznawania pszczół zdrowych i zarażonych dręczeniem pszczelim. W ramach niemieckiego projektu IndusBee 4.0 [92] powstał prototyp systemu bazującego na analizie sygnałów akustycznych mierzonych wewnątrz ula i na tej podstawie rozpoznającego aktywności pszczół. Do jego budowy wykorzystano komputer RaspberryPi, mikrofon, a ponadto czujniki temperatury i tensometr. Autor, wśród planów rozwoju prototypu, wymienia rozbudowę rozwiązania o kamerę umożliwiającą nagrywanie obrazu sprzed wejścia do ula oraz wykorzystanie czujników gazu. Oba rozszerzenia miałyby umożliwić wykrywanie obecności dręcza pszczelego w rodzinie pszczelej. System, który zaproponował A. Zgank [93] jest rozwiązaniem typu IoT, opartym o gromadzenie i analizę danych akustycznych oraz warunków atmosferycznych (temperatura, wilgotność powietrza, ciśnienie atmosferyczne) w celu detekcji momentu rojenia się pszczół. Zbierane dane są przetwarzane za pomocą mikroprocesora ATmega i przesyłane poprzez sieć Wi-Fi do chmury obliczeniowej. Rozpoznawanie momentu rojenia umożliwia podobny [94] do poprzedniego system, w którym wraz z odczytami temperatury prowadzone są pomiary poziomu natężenia dźwięku w ulu. System oparty jest o mikrokontroler Arduino Uno, a dane są transmitowane za pomocą modułu Wi-Fi do chmury obliczeniowej. Równoległa analiza danych z obu źródeł umożliwiła

wykrycie początku rojenia się pszczół, a autorzy wymieniają funkcjonalność powiadamiania użytkownika o takiej sytuacji za pomocą platformy online jako kolejny krok w pracach nad systemem. Połączenie pozyskiwania oraz przetwarzania dźwięków i obrazów, wraz z danymi z czujników pogodowych przedstawione zostało w projekcie BeeMon [95]. System umożliwia monitorowanie zdrowia rodziny pszczelej poprzez udostępnianie pozyskanych danych zainteresowanym użytkownikom w formie strony internetowej, z dostępnymi narzędziami do wizualizacji danych. W planach rozwoju systemu jest między innymi funkcjonalność powiadamiania użytkowników o odbiegających od normy odczytach z czujników.

Kolejnym, często stosowanym przy systemach monitorujących zachowanie pszczół, podejściem jest pozyskiwanie oraz analiza obrazów i filmów wideo. Metoda ta jest nieinwazyjna dla rodzin pszczelich, a połączenie jej z algorytmami uczenia maszynowego daje dobre rezultaty. Jednym z pierwszych tego typu rozwiązań był opracowany przez J. Campbell et al. [77] system, który zliczał pszczoły wlatujące oraz wylatujące z ula, oraz rozpoznawał cztery rodzaje poruszania się pszczół. Kamera została umieszczona nad wejściem do ula, a w celu minimalizacji wpływu otoczenia na pozyskiwane obrazy, poniżej wejścia do ula umieszczono platformę, na której pszczoły mogły lądować. Zliczanie pszczół z wykorzystaniem analizy obrazów zostało również przedstawione w artykule [96]. Autorzy rozwiązania zaprojektowali specjalne przejście dla pszczół, nad którym umieszczona została kamera i projektor światła podczerwonego, służące do rejestrowania obrazów. W celu identyfikacji, każda pszczoła z ula została oznaczona papierowym znacznikiem z nadrukowanym numerem. Eksperyment, poza zliczaniem, pozwolił poznać godziny wzmożonej aktywności pszczół, niestety, po 15 dniach pszczoły całkowicie opuściły ul. Wykorzystanie filmów wideo w celu rozpoznawania i zliczania pszczół, które wracają do ula z pyłkiem kwiatowym zostało opracowane w artykule [97]. Detekcja pszczół niosących pyłek na dalszych etapach rozwoju systemu ma umożliwić monitorowanie efektywności procesu zapylania. Rozwiązanie oparte jest o komputer RaspberryPi z dedykowaną kamerą, umożliwiającą pozyskiwanie filmów i obrazów w rozdzielczości maksymalnie 1920 x 1020 pikseli i jakości HD (ang. *High Definition*). Inny system pozwalający na rozpoznawanie pszczół zbierających pyłek został opisany przez [98]. Jest on oparty o komputer jednopłytkowy RaspberryPi połączony z mikroprocesorem NVidia Jetson TX2, na którym wykonywane są operacje rozpoznawania obrazów (w tym rozpoznawanie pszczół niosących pyłek i ich zliczanie) z wykorzystaniem konwolucyjnych sieci neuronowych (CNN). Oprócz pozyskiwania obrazów, system wyposażony został w różne czujniki (m.in. temperatury,

wilgotności, opadów czy prędkości wiatru), dzięki czemu możliwe było określenie powiązań pomiędzy aktywnością pszczół a warunkami atmosferycznymi.

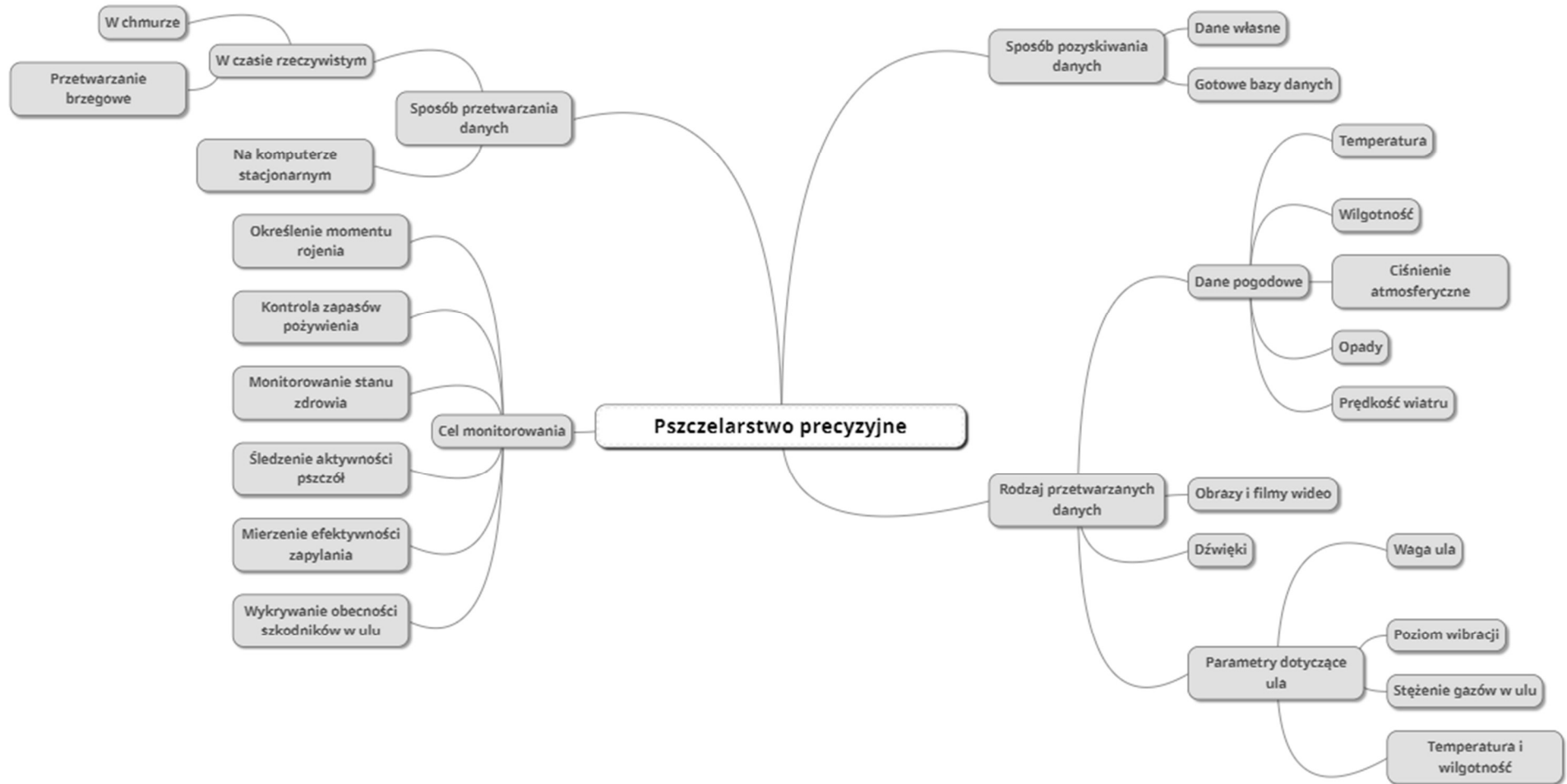
Rozwiązania wykorzystujące obrazy z kamery wideo często umożliwiają kilka funkcjonalności w ramach jednego systemu. Przykładowo, opracowany w Niemczech system DeepBees [99] posiada moduł do rozpoznawania rodzaju owadów znajdujących się przy wejściu do ula, co może pomóc przy wczesnym rozpoznawaniu zagrożeń; moduł detekcji pszczół niosących pyłek oraz moduł analizujący postawę pszczoły, który umożliwia wykrycie deformacji odwłoku, wskazujących na chorobę. Rozwiązanie oparte jest o komputer RaspberryPi, natomiast do klasyfikacji obrazów zostały wykorzystane konwolucyjne sieci neuronowe. Autorzy projektu wskazują na plany poszerzenia systemu m.in. o moduł umożliwiający detekcję pszczół chorujących na warrozę. Inny, zaproponowany przez naukowców z Tajwanu, prototyp systemu do monitorowania stanu zdrowia pszczół (w tym wykrywania zakażenia wirusem *nosema* oraz warrozą) posiada czujniki temperatury, wilgotności, mikrofon, GPS oraz kamerę termowizyjną. Chociaż w artykule [100] nie zostały opisane żadne badania, poza testowym pozyskiwaniem danych z czujników, autorzy chcą umożliwić m.in. rozpoznawanie aktywności pszczół, kontrolowanie ich stanu zdrowia oraz wykorzystywać sygnał GPS do zabezpieczania uli przed kradzieżą.

Literatura coraz częściej opisuje rozwiązania które wykrywają obecność dręcza pszczelego, bazujące na pozyskiwaniu i/lub analizie obrazów. W artykule [101] opisano algorytm umożliwiający wykrywanie obecności dręcza pszczelego w sześciokątnych komórkach plastrów pszczelich. Autorzy rozwiązania uzyskali dobre wyniki (dokładność na poziomie ok. 90% w przypadku wykrywania dręcza na odwłoku pszczoły), a do walidacji algorytmów wykorzystali obrazy z bazy Narodowego Uniwersytetu w Kostaryce. Moduł umożliwiający pozyskiwanie danych w celu rozpoznawania pszczół zarażonych warrozą został przedstawiony przez [102]. Celem badań było znalezienie optymalnego rozwiązania do zbierania i przetwarzania obrazów, które mogłoby zostać wykorzystane jako część kompleksowego systemu umożliwiającego monitorowanie pszczół. Cały system, który umożliwiał zbieranie i analizę danych został przedstawiony przez [103]. Aby uzyskać dobrej jakości dane, zbudowano specjalne przejście dla pszczół, w którym umieszczona była kamera, umożliwiająca jednoczesne pozyskiwanie obrazów w kolorze oraz w spektrum zbliżonym do podczerwieni, bez zagrożenia dla dalszej analizy danych w czasie rzeczywistym. Następnie, dane były poddawane kompresji i przesyłane poprzez dwa złącza Ethernet do dalszej analizy. Do wykrywania pszczół wykorzystano algorytm odejmowania tła, natomiast do wykrywania dręcza pszczelego – konwolucyjne sieci neuronowe. Pozostając w tematyce związanej

z wykrywaniem warrozy warto przywołać interesujące wyniki analizy stężenia różnych gazów, w tym dwutlenku węgla, w ulu. W oparciu o te dane autorzy publikacji [104], [105] określali poziom zarażenia rodziny pszczelej dręczem pszczelim. Mimo, że nie są to rozwiązania z grupy systemów Internetu Rzeczy, uzyskane wyniki wskazują, że implementacja czujników gazu w systemach IoT i odpowiednia analiza danych mogą być pomocą dla pszczelarzy zmagających się z problemem pszczół chorujących na warrozę.

Wśród rozwiązań związanych z tematem monitorowania pszczół i pszczelarstwa precyzyjnego opisywanych w literaturze znajdują się również prace, które nie wpisują się w powyższe kategorie. Przykładowo, [106] mierzyli poziom wibracji ula, a zebrane dane posłużyły autorom do opracowania powiązań między częstotliwością wibracji i momentem rojenia się pszczół. Belgijscy naukowcy [107] zaproponowali system IoT, który rozpoznaje rodzaje owadów zapylających rośliny uprawne na podstawie analizy dźwięków. Ponadto, system ma za zadanie mierzyć efektywność zapylania. Dzięki zastosowaniu wielu czujników umieszczonych w różnych miejscach na terenie upraw, rolnicy mają dostęp do danych z całego pola, które mogą w łatwy sposób monitorować za pomocą aplikacji mobilnej. Wykorzystanie głębokich sieci neuronowych [108] do analizy danych akustycznych pochodzących z ula umożliwiło rozpoznawanie okresu rojenia. Chociaż badania w tym artykule zostały wykonane w oparciu o ogólnodostępne dane z projektu Open Source Beehive Project, autor sugeruje, że rozwiązanie może zostać zastosowane w systemach Internetu Rzeczy.

Powyższe przykłady pokazują, że od kilkunastu lat tematyka pszczelarstwa precyzyjnego jest systematycznie rozwijana. Wciąż jednak pozostaje wiele zagadnień, które do tej pory nie zostały poruszone, lub były przedmiotem niewielu badań. Na Rys. 8 przedstawione zostały najpopularniejsze podejścia wykorzystywane w systemach umożliwiających monitorowanie pszczół.



Rys. 8 Systemy pszczelarstwa precyzyjnego

3.2.2. Klasyfikacja systemów ze względu na sposób przetwarzania danych

Przedstawione w poprzednim podrozdziale rozwiązania można również sklasyfikować ze względu na sposób przetwarzania danych. Dokonując tego rodzaju analizy, należy uwzględnić następujące składowe systemów: jednostkę wykonującą obliczenia, zastosowane algorytmy, protokoły komunikacyjne, a także sposób pozyskiwania oraz przechowywania danych. Aby uzyskać przejrzyste rezultaty, rozwiązania poddawane klasyfikacji powinny być możliwie kompleksowymi systemami, w przeciwnym wypadku niemożliwe będzie uwzględnienie większości powyższych składowych. Systemy, przedstawione w pierwszej części rozdziału i spełniające kryterium rozbudowanych i kompleksowych rozwiązań zostały zebrane i opisane w Tab. 2.

Analizując dane przedstawione w tabeli można zauważyć pewne zależności. Początkowo, systemy dedykowane pszczelarstwu precyzyjnemu oparte były głównie o komputery PC, które wraz z rozwojem technologii były zastępowane mikrokontrolerami. Najczęściej wykorzystywano (i trend ten pozostaje aktualny) płytki RaspberryPi, zdecydowanie rzadziej używano mikrokontrolerów ATmega i Arduino. Sposób pozyskiwania danych w większości rozwiązań nie zmienia się na przestrzeni lat – prawie wszystkie opisane systemy korzystają z danych pozyskiwanych w ramach własnych obserwacji; tylko kilka korzysta z gotowych baz danych. Zmienił się natomiast sposób ich przechowywania. Dawniej, dane były przechowywane w lokalnych bazach danych, obecnie często są przesyłane do chmury obliczeniowej oraz na serwery zdalne. Starsze rozwiązania do przesyłania danych między komponentami wykorzystywały połączenia fizyczne (kable), natomiast nowsze oparte są o transmisję bezprzewodową, głównie za pomocą protokołów Zigbee, LoRaWAN oraz WiFi. Analizując wykorzystywane algorytmy, można zauważyć dwie zależności – zarówno w nowszych, jak i w starszych systemach często stosowana jest maszyna wektorów nośnych (ang. *Support Vector Machine, SVM*), ponadto w nowszych rozwiązaniach wykorzystuje się sieci neuronowe.

Tab. 2 Zestawienie rozwiązań opisanych w literaturze

Autorzy	Rodzaj danych	Cel przetwarzania danych	Pozyskiwanie danych	Przechowywanie danych	Jednostka obliczeniowa	Sposób transmisji danych	Algorytmy
[77]	Obrazy	Śledzenie aktywności i toru lotu pszczół	Kamera	Brak danych	Brak danych	Magistrala IEEE 1394	Odejmowanie tła, modele oparte o rozkład Gaussa i progowanie obrazu
[87]	Obrazy, temperatura i wilgotność	Wykrywanie rojenia się pszczół	Kamera, czujnik	Dysk twardy	Komputer PC	Kabel połączony do portu COM ¹	Brak danych
[90]	Dźwięki, temperatura wewnątrz i na zewnątrz ula	Wykrywanie rojenia się pszczół	Mikrofon, czujnik	Lokalna baza danych	Komputer PC	Protokół Zigbee	Brak danych
[106]	Wibracje ula	Określenie poziomu zarażenia pszczół warrozą	Wzmacniacz magnetyczny	Dysk twardy	Komputer PC	Brak	Analiza głównych składowych
[96]	Obrazy	Śledzenie lotu pszczół	kamera	Dysk twardy	Komputer PC	Kabel USB	Maszyna wektorów nośnych, analiza głównych składowych

¹ CINAT – *The Centre of Investigation in to Tropical Apiculture of Costa Rica*, centrum badań nad pszczołami w Kostaryce

Autorzy	Rodzaj danych	Cel przetwarzania danych	Pozyskiwanie danych	Przechowywanie danych	Jednostka obliczeniowa	Sposób transmisji danych	Algorytmy
[101]	Filmy wideo	Identyfikacja obecności dręcza pszczelego w komórkach pszczół	Baza danych CINAT ¹	Nie dotyczy	Brak danych	Nie dotyczy	Algorytm odejmowania tła,
[91]	Dźwięki	Analiza pod kątem chorób pszczół	mikrofon	Lokalnie na urządzeniu	Beagleboard	Protokół Zigbee	Analiza głównych składowych, Maszyna wektorów nośnych, Liniowa analiza dyskryminacyjna
[79]	Temperatura	Wykrywanie rojenia się pszczół	Czujnik	Zdalny serwer	RaspberryPi	Brak danych	Autorski algorytm umożliwiający wykrycie momentu rojenia
[85]	Temperatura i wilgotność powietrza, stężenie gazów	Śledzenie aktywności pszczół	Czujniki	chmura	ATmega1281	Protokół Zigbee	Algorytmy oparte na progowaniu obrazu, Drzewa decyzyjne

¹ COM – ang. *communication port*, port szeregowy urządzenia

Autorzy	Rodzaj danych	Cel przetwarzania danych	Pozyskiwanie danych	Przechowywanie danych	Jednostka obliczeniowa	Sposób transmisji danych	Algorytmy
[80]	Temperatura i wilgotność powietrza, waga ula	Analiza produktywności pszczół, zbieranie informacji o ilości wyprodukowanego miodu	czujniki	Lokalny serwer, chmura	komputer przemysłowy, Atmega1281	Protokół Zigbee	Brak danych
[84]	Temperatura i wilgotność powietrza	Śledzenie aktywności pszczół	Czujniki	Lokalna baza danych	Moduł ESP8266	Wi-Fi	Brak danych
[81]	Waga ula	Brak danych	Waga	Zdalny serwer	Brak danych	Sieć GSM	Nie dotyczy
[86]	Temperatura i wilgotność powietrza, temperatura w ulu	Brak danych	Czujniki	Rozproszony system plików HDFS ¹	LoPy	Protokół LoRa/LoRaWAN	Brak danych

¹ HDFS – ang. *The Hadoop Distributed File System*, system rozproszonych plików

Autorzy	Rodzaj danych	Cel przetwarzania danych	Pozyskiwanie danych	Przechowywanie danych	Jednostka obliczeniowa	Sposób transmisji danych	Algorytmy
[102]	Obrazy	Wykrywanie dręcza pszczelego	Kamera	Brak danych	Brak danych	Brak danych	Algorytmy odejmowania tła, naiwny klasyfikator Bayesa, k najbliższych sąsiadów (k-NN), sieci neuronowe, losowy las decyzyjny
[82]	Waga ula, temperatura i wilgotność w ulu, temperatura powietrza, punkt rosy, ciśnienie atmosferyczne, prędkość wiatru	Analiza kondycji zdrowotnej pszczół	Czujniki, waga	lokalnie	Nie dotyczy	Nie dotyczy	Naiwny klasyfikator Bayesa, k najbliższych sąsiadów (k-NN), sieci neuronowe, losowy las decyzyjny

Autorzy	Rodzaj danych	Cel przetwarzania danych	Pozyskiwanie danych	Przechowywanie danych	Jednostka obliczeniowa	Sposób transmisji danych	Algorytmy
[83]	Temperatura i wilgotność powietrza (wewnątrz i na zewnątrz ula), strumień wideo, pozycja ula	Regulacja warunków panujących w ulu, zmniejszenie poziomu śmiertelności pszczół przy rojeniu się	Czujniki, kamera, żyroskop	Chmura	Atmega328, Komputer 32 - bitowy z architekturą ARM	LoRaWAN, MQTT ¹	Brak danych
[88]	Temperatura i wilgotność, waga	Analiza produktywności pszczół	czujniki	chmura	Moduł ESP8266	WiFi, MQTT	Nie dotyczy
[92]	Temperatura i wilgotność powietrza, dźwięki	Monitorowanie aktywności pszczół	Czujniki, mikrofon	Brak danych	Raspberry Pi	Brak danych	K najbliższych sąsiadów
[99]	Zdjęcia	Śledzenie pszczół	Zewnętrzne zbiory danych	Chmura	Brak danych	Brak danych	Konwolucyjne sieci neuronowe

¹ MQTT – ang. *MQ Telemetry Transport*, protokół transmisji danych oparty o wzorzec publikacja – subskrypcja.

Autorzy	Rodzaj danych	Cel przetwarzania danych	Pozyskiwanie danych	Przechowywanie danych	Jednostka obliczeniowa	Sposób transmisji danych	Algorytmy
[103]	Filmy wideo	Określenie poziomu zarażenia warrozą	kamera	Brak danych	Brak danych	GigE	Liniowa analiza dyskryminacyjna, Algorytmy odejmowania tła, Konwolucyjne sieci neuronowe
[107]	Dźwięki	Analiza produktywności pszczół	Mikrofon	Chmura	Komputer jednopłytkowy	LTE	Maszyna wektorów nośnych
[89]	Temperatura, wilgotność powietrza, ciśnienie atmosferyczne, ruch pszczół	Wykrywanie wandalizmu, detekcja chorób	Baza danych Hobos	Nie dotyczy	RaspberryPi	LoRaWAN	Maszyna wektorów wspierających
[93]	Dźwięki	Wykrywanie rojenia się pszczół	Dane własne oraz wygenerowane w ramach projektu Open Source Beehives (OSBH)	Lokalnie, zdalny serwer	Brak danych	WiFi	Model mieszaniny Gaussa, ukryty model Markowa
[94]	Dźwięki, temperatura i wilgotność	Wykrywanie rojenia się pszczół	Mikrofon, czujniki	chmura	Arduino UNO, moduł ESP8266	WiFi	Brak danych

Autorzy	Rodzaj danych	Cel przetwarzania danych	Pozyskiwanie danych	Przechowywanie danych	Jednostka obliczeniowa	Sposób transmisji danych	Algorytmy
[100]	Temperatura, wilgotność, dźwięki, obrazy termiczne, sygnał GPS	Brak danych (prototyp systemu)	czujniki	chmura	GrovePi	WiFi	Brak danych
[95]	Obrazy, dźwięki, temperatura i wilgotność	Zdalne monitorowanie pasieki	Kamera, mikrofon, czujniki	Serwer	RaspberryPi	MQTT, FTP	Algorytmy uczenia maszynowego i głębokiego (w przyszłości)
[98]	Obrazy, temperatura, wilgotność, poziom opadów	Monitorowanie pszczół zbierających nektar	Kamera, czujniki	Zdalny serwer	RaspberryPi	WiFi	Konwolucyjne sieci neuronowe
[108]	Dźwięki	Wykrywanie rojenia się pszczół	Dane wygenerowane w ramach projektu Open Source Beehives (OSBH)	Nie dotyczy	Brak danych	Brak danych	Głębokie sieci neuronowe

4. Przyjęte rozwiązania architektoniczne i algorytmiczne

W celu realizacji założeń zdefiniowanych we wstępie niniejszej rozprawy (rozdział 1.2) został zaprojektowany, a następnie zaimplementowany system Internetu Rzeczy, który wykorzystuje algorytmy głębokiego uczenia maszynowego i przetwarzania obrazów do wykrywania dręcza pszczelego na ciele pszczoły. W poniższych podrozdziałach zostały opisane kolejno: wczesne wersje algorytmów detekcji (ang. *object detection*) pszczół i dręcza pszczelego, architektura i sposób działania proponowanego systemu, zaimplementowane algorytmy i modele uczenia maszynowego, a także definicja i zasady działania algorytmu adaptacyjnego wyznaczania okna zagęszczenia wykonywanego przy uruchomieniu systemu i powtarzanego kilkakrotnie w czasie działania systemu, w celu dopasowania wielkości obrazu oraz liczby pobieranych i przetwarzanych klatek filmowych do aktualnych aktywności pszczół. W niniejszym rozdziale opisane zostały również wykorzystane miary oceny jakości identyfikacji pszczół i detekcji dręcza pszczelego.

4.1. Wczesne wersje algorytmów identyfikacji pszczół i detekcji dręcza pszczelego

Na wczesnym etapie badań nad wykrywaniem obecności dręcza pszczelego grupa badawcza prowadzona przez dr hab. inż. Dariusza Mrozka, prof. Politechniki Śląskiej, do której należała również autorka niniejszej rozprawy, opracowała urządzenia IoT oraz algorytmy umożliwiające identyfikację pszczół na klatkach filmowych oraz detekcję dręcza pszczelego. W ramach przeprowadzonych prac opracowane zostały urządzenia oparte o komputer jednopłytkowy RaspberryPi z akcelerorem Google Coral [109] oraz

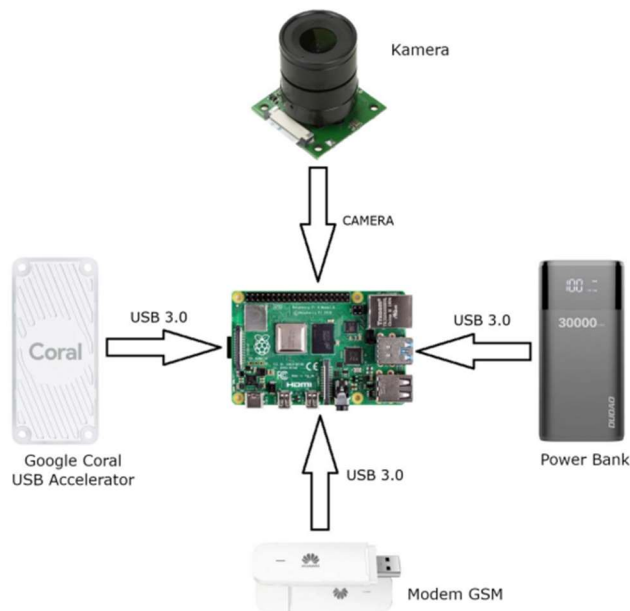
o komputer jednopłytkowy NVidia Jetson Nano 2GB [110]. Zastosowane algorytmy zostały szczegółowo opisane w rozdziałach (kolejno) 4.1.1 oraz 4.1.2, natomiast wyniki uzyskane w trakcie eksperymentów badawczych zostały zestawione z rezultatami osiągniętymi z wykorzystaniem algorytmów będącym przedmiotem niniejszej rozprawy w rozdziale 5.6.

4.1.1. Algorytmy działające na platformie RaspberryPi z akceleratorem USB Google Coral

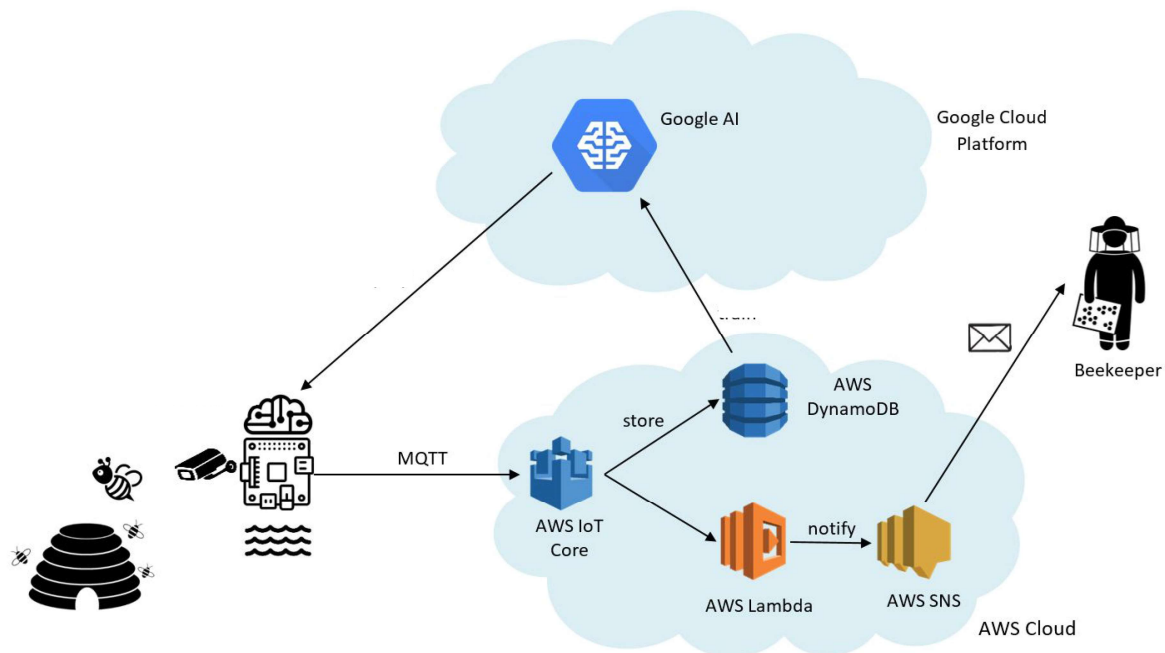
Pierwsze z urządzeń IoT umożliwiających wykrywanie pszczół i ich klasyfikację ze względu na występowanie (bądź nie) dręcza pszczelego, przedstawione na *Rys. 9*, działało w oparciu o komputer jednopłytkowy RaspberryPi w wersji 4, z czterordzeniowym procesorem Broadcom BCM2711 o 64-bitowej architekturze ARM-8 Cortex-A72, taktowaniu procesora o częstotliwości 1,5 GHz i pamięci operacyjnej RAM 4 GB. Do wykonywania procesów detekcji wykorzystano akcelerator USB Google Coral, który, dzięki wbudowanemu koprocesorowi TPU Edge (ang. *Tensor Processing Unit*), umożliwia szybkie wnioskowanie oparte o modele uczenia maszynowego. Aby zapewnić połączenie z wybraną chmurą obliczeniową Amazon AWS (Amazon Web Services) urządzenie zostało wyposażone w modem GSM, a przesyłanie danych odbywało się za pomocą protokołu MQTT. Pozyskiwanie obrazów sprzed ula realizowane było z wykorzystaniem kamery ArduCam OV5647, której maksymalna rozdzielczości wynosi 5Mpx. Urządzenie było zasilane z baterii typu powerbank o pojemności 30000 mAh.

Ogólna architektura urządzenia systemu IoT wykorzystywanymi usługami została przedstawiona na rysunku *Rys. 10*. Działanie systemu rozpoczyna się od nawiązania połączenia z chmurą AWS. Następnie pobierane są kolejne klatki nagrania. Urządzenie brzegowe, przygotowuje pobraną klatkę obrazu do rozmiaru 512 x 512 px, który jest oczekiwany na wejściu modelu sieci neuronowej wykorzystywanego w procesie detekcji pszczół. Po dokonaniu segmentacji pszczół, wyodrębnione obiekty pszczół są zapisywane. Kolejnym krokiem jest przekształcenie wyodrębnionych zdjęć do rozmiaru 224 x 224 px. Następnie wykonywana jest klasyfikacja obiektów pod kątem obecności dręcza pszczelego. W przypadku sklasyfikowania pszczoły jako chorej, jej zdjęcie przesyłane jest do chmury Amazon AWS. Usługa AWS IoT Core wywołuje usługę AWS Lambda oraz zapisuje otrzymane zdjęcie do bazy danych DynamoDB. Funkcja AWS Lambda

uruchamia natomiast usługę powiadamiania SNS (Simple Notification Service), która wysyła wiadomość e-mail do użytkownika.



Rys. 9. Urządzenie IoT oparte o komputer RaspberryPi z akceleratorem USB Google Coral ([109])



Rys. 10 Ogólna architektura systemu IoT wykorzystującego komputer RaspberryPi oraz chmurę AWS i Google Cloud Platform ([109])

W procesie wnioskowania wykorzystane zostały modele uczenia maszynowego oparte na usługach Google AutoML Vision Edge, kompatybilne z akceleratorem USB Google Coral. Modele zostały przygotowane na platformie chmurowej Google Cloud Platform za pomocą usługi Google AI, w sposób umożliwiający „dotrenowanie” modeli na danych otrzymanych z urządzenia IoT i zapisanych w bazie danych chmury AWS, a następnie przesłanie nowego modelu do urządzenia IoT. Architektura modeli oparta była o konwolucyjne sieci neuronowe (CNN) wykorzystujące tzw. uczenie przez wzmocnienie (ang. *reinforcement learning*), natomiast szczegółowe elementy architektury modeli nie zostały udostępnione przez dostawcę usługi. Przeprowadzone eksperymenty badawcze obejmowały sprawdzenie jakości procesów detekcji pszczoł oraz dręcza pszczelego w oparciu o miary precyzji, czułości oraz średniej harmonicznej obu wartości – miary F1. Ponadto zmierzone zostały czasy przetwarzania na poszczególnych etapach pracy urządzenia oraz dla różnych rozdzielczości obrazów wejściowych.

4.1.2. Algorytmy działające na platformie NVidia Jetson Nano w wersji 2GB

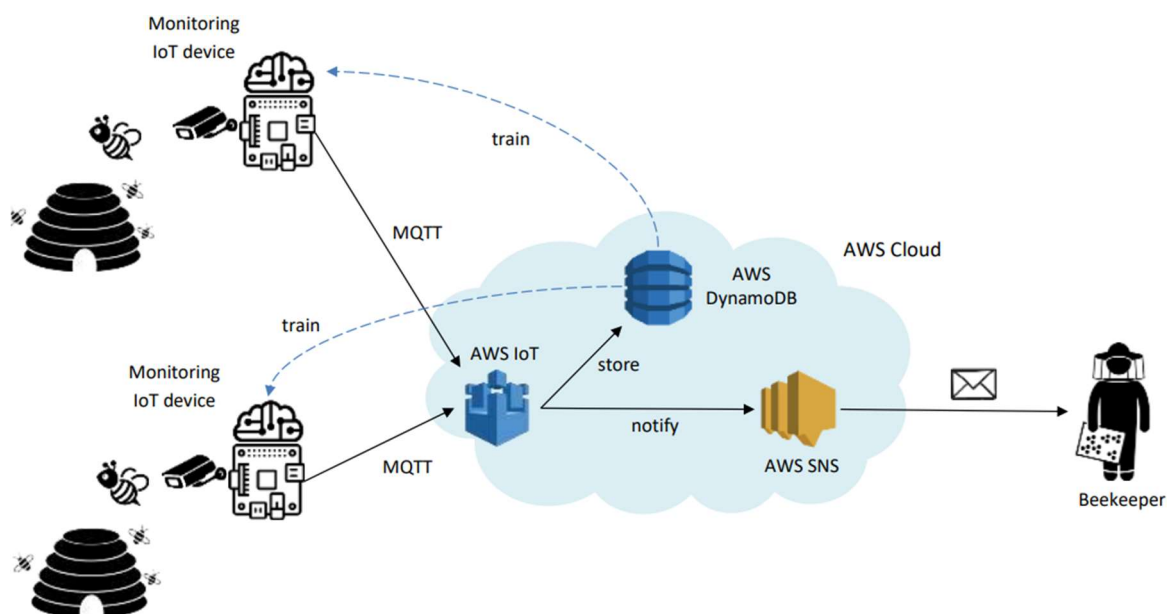
W kolejnym podejściu w trakcie wczesnych prac nad budową urządzeń IoT umożliwiających detekcję pszczoł, a następnie ich klasyfikację, wykorzystano komputer NVIDIA Jetson Nano w wersji z pamięcią operacyjną o rozmiarze 2GB, wyposażony w czterordzeniowy procesor ARM Cortex A57 o taktowaniu zegara 1.43 GHz, oraz układ graficzny NVidia Maxwell z 128 rdzeniami CUDA. Urządzenie przedstawiono na *Rys. 11*.

Zasada działania całego systemu, który jako urządzenie brzegowe wykorzystuje powyższe urządzenie IoT, zbliżona była do systemu opartego o minikomputer RaspberryPi. Urządzenie brzegowe, wyposażone w kamerę Sony IMX219-77, umożliwiała pozyskiwanie obrazów, następnie wykorzystywało modele konwolucyjnych sieci neuronowych w procesach detekcji i klasyfikacji pszczoł. W przypadku detekcji dręcza pszczelego, dzięki wyposażeniu urządzenia w modem GSM, nawiązywane było połączenie z chmurą obliczeniową Amazon AWS, do której, wykorzystując protokół MQTT, przesyłano informację o wykryciu chorej pszczoły. Ponadto przesłane zdjęcie chorej pszczoły przechowywane było w usłudze Amazon AWS DynamoDB, dzięki czemu gromadzone były dane, które mogłyby zostać wykorzystane w procesie ponownego wytrenowania sieci neuronowej umożliwiającej klasyfikację pszczoł. Część systemu wykorzystująca usługi chmury Amazon AWS została przygotowana w taki sposób, że

możliwe jest zastosowanie większej liczby urządzeń brzegowych i tym samym jednoczesne monitorowanie wielu uli jednocześnie. Architektura całego systemu została przedstawiona na Rys. 12.



Rys. 11 Urządzenie IoT działające w oparciu o komputer NVidia Jetson Nano w wersji 2GB [110]



Rys. 12 Ogólna architektura systemu IoT wykorzystującego komputer NVidia Jetson Nano w wersji 2GB oraz chmurę AWS [110]

Badania przeprowadzone na tym urządzeniu obejmowały sprawdzenie działania następujących rozwiązań:

- algorytmy odejmowania tła umożliwiające wykrywanie obiektów pszczół,
- model głębokiej sieci neuronowej umożliwiający detekcję pszczół,
- model głębokiej sieci neuronowej umożliwiający klasyfikację pszczół ze względu na stan zdrowia – obecność dręcza pszczelego na ciele pszczoły.

Wśród algorytmów odejmowania tła porównano dwie metody MOG2 [111] oraz KNN [112]. Metoda MOG2 umożliwia zamodelowanie każdego z pikseli reprezentujących tło za pomocą gęstości prawdopodobieństwa mieszaniny Gaussa. Wagi mieszaniny Gaussa reprezentują ilość czasu, jaką piksele były opisane danym kolorem – im większa wartość wagi tym większe prawdopodobieństwo, że dany piksel stanowi tło analizowanego obrazu. Aktualizacja parametrów (wag) oraz wartości składowych piksela odbywa się z wykorzystaniem równań rekurencyjnych. Algorytm odejmowania tła wykorzystujący metodę KNN (ang. *k-nearest neighbors*, k-najbliższych sąsiadów) oblicza, a następnie przechowuje wszystkie możliwe wartości składowych pikseli oraz dokonuje klasyfikacji nowych próbek w oparciu o podobieństwo do sąsiednich pikseli. Do porównania wyników zwracanych przez każdą z metod wykorzystano taki sam zestaw danych – klatki z dwóch filmów o różnych rozdzielczościach – 960×540 px oraz 480×270 px. Następnie porównane zostały uzyskane wyniki segmentacji pszczół w oparciu o miary jakości – precyzję, czułość oraz średnią harmoniczną obu wartości – miarę F1.

Modele konwolucyjnych sieci neuronowych zostały wytrenowane z wykorzystaniem środowiska uczenia głębokiego *Caffe* [113], a następnie wykorzystane do przygotowania silników wnioskowania, uruchamianych w środowisku wykonawczym *TensorRT* [114], które są kompatybilne z architekturą komputera NVidia Jetson Nano. Detekcja pszczół z wykorzystaniem konwolucyjnych sieci neuronowych była oparta o model sieci neuronowej typu DetectNet, ze strukturą warstw zbliżoną do 22 warstwowej sieci neuronowej GoogLeNet. Zastosowanie wstępnie wytrenowanej sieci neuronowej opartej na modelu GoogLeNet umożliwiło redukcję czasu potrzebnego do wytrenowania modelu klasyfikacji do 131 epok oraz uzyskanie dobrych wyników. Dla sieci neuronowej odpowiedzialnej za detekcję pszczół przeprowadzono eksperymenty badawcze na filmach w dwóch rozdzielczościach – 960×540 px, oraz 1980×1080 px. Dla analizowanych filmów obliczone zostały miary jakości takie, jak w przypadku algorytmów odejmowania

tła – precyzja, czułość oraz średnia harmoniczna obu wartości. Ponadto porównano wydajność czasową algorytmów uczenia maszynowego oraz algorytmów odejmowania tła.

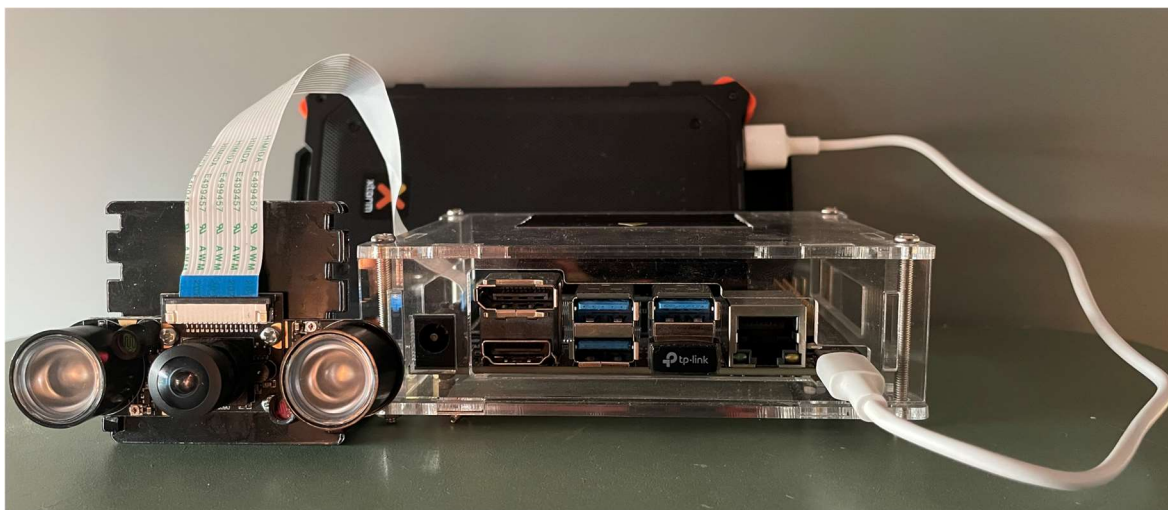
Do procesu detekcji dręcza pszczelego zastosowano model konwolucyjnej sieci neuronowej o architekturze zbliżonej do architektury modelu wykorzystanego w procesie detekcji pszczół. Model ten został przygotowany w podobny sposób, jednak do jego wytrenowania konieczne było jedynie 13 epok.

Rezultaty otrzymane w badaniach z wykorzystaniem urządzeń i algorytmów wczesnych wersji systemów detekcji pszczół i dręcza pszczelego (opisane szerzej w rozdziale 5.6) były dość dobre ze względu na wymienione wcześniej miary jakości, ale ich głównymi wadami był czas przetwarzania – w przypadku urządzenia działającego z wykorzystaniem akceleratora Google Coral przeprowadzone analizy dotyczące czasu oparte były o losowo wybrane klatki z filmu, i brakowało analizy zapotrzebowania czasowego w ujęciu całej pracy urządzenia, natomiast w przypadku rozwiązania opartego na komputerze NVidia Jetson Nano konieczne było pomijanie klatek filmu w celu przetwarzania obrazów w czasie rzeczywistym. Opracowane na potrzeby niniejszej rozprawy urządzenie zaprojektowane zostało w sposób umożliwiający zmniejszenie czasu koniecznego do przetwarzania poszczególnych klatek, dzięki czemu ograniczona została liczba pomijanych klatek przy jednoczesnym zachowaniu porównywalnych wyników dotyczących detekcji i klasyfikacji pszczół.

4.2. Architektura i działanie urządzenia IoT umożliwiającego detekcję pszczół i ich klasyfikację pod kątem obecności dręcza pszczelego

Urządzenie brzegowe systemu umożliwiającego monitorowanie pszczół oparte jest o komputer NVidia Jetson Nano [115] w wersji 4GB, dedykowany do tworzenia urządzeń przetwarzających obrazy oraz wykorzystujących modele uczenia maszynowego i sieci neuronowe. Komputer jednopłytkowy NVidia Jetson Nano posiada czterordzeniowy procesor ARM[®] Cortex[®]-A57 MP Core o taktowaniu 1,43 GHz oraz procesor graficzny, który zbudowany jest w oparciu o architekturę NVIDIA Maxwell[™] ze 128 rdzeniami NVIDIA CUDA[®]. Posiada 4GB pamięci operacyjnej i 16 GB pamięci masowej Flash,

gigabitowe gniazdo Ethernet oraz złącza HDMI. Jetson Nano ma możliwość podłączenia kamery poprzez interfejs MIPI-CSI oraz USB. Układ wyposażony jest także w zestaw wyprowadzeń z portami wejścia/wyjścia ogólnego przeznaczenia (GPIO, ang. *general-purpose input/output*), co umożliwia podłączenie różnego rodzaju urządzeń zewnętrznych. Na Rys. 13 przedstawiono urządzenie brzegowe wraz z jego poszczególnymi składowymi (kamerą, kartą sieciową oraz baterią typu powerbank), opisanymi w dalszej części niniejszego rozdziału.



Rys. 13 Urządzenie brzegowe oparte o komputer NVidia Jetson Nano w wersji 4GB

Pozyskiwanie obrazu zostało zrealizowane z wykorzystaniem kamery IMX219 160IR Camera IC Test Board [116] opartej o moduł Sony IMX219. Jest to urządzenie dedykowane dla minikomputerów, m.in. NVidia Jetson Nano i RaspberryPi, polecane do współpracy z aplikacjami wykorzystującymi sztuczną inteligencję do rozpoznawania obiektów. Połączona z komputerem poprzez taśmę CSI kamera umożliwia pozyskiwanie obrazów w jakości 8 Mpx z maksymalną prędkością 30 klatek na sekundę, w maksymalnej rozdzielczości 3280×2464 px, natomiast jej maksymalny kąt widzenia wynosi 162 stopnie. Obrazy pochodzące z kamery mają czerwono-fioletowy kolor – jest to spowodowane brakiem filtra podczerwieni (ang. *infra-red filter*, IR) w układzie matrycy światłoczułej (ang. *charge-coupled device*, CCD). Brak filtra IR wraz z wykorzystaniem modułu podświetlenia wyposażonego w diody podczerwieni (IR LED) sprawia, że układ jest odporny na zmiany w oświetleniu otoczenia (wynikające ze zmiany nasłonecznienia stanowiska badawczego) i umożliwia pozyskiwanie dobrej jakości obrazu w szerokim zakresie warunków atmosferycznych. Co ważne, diody IR LED generują światło niewidoczne dla człowieka, ale, w przeciwieństwie do promieniowania ultrafioletowego

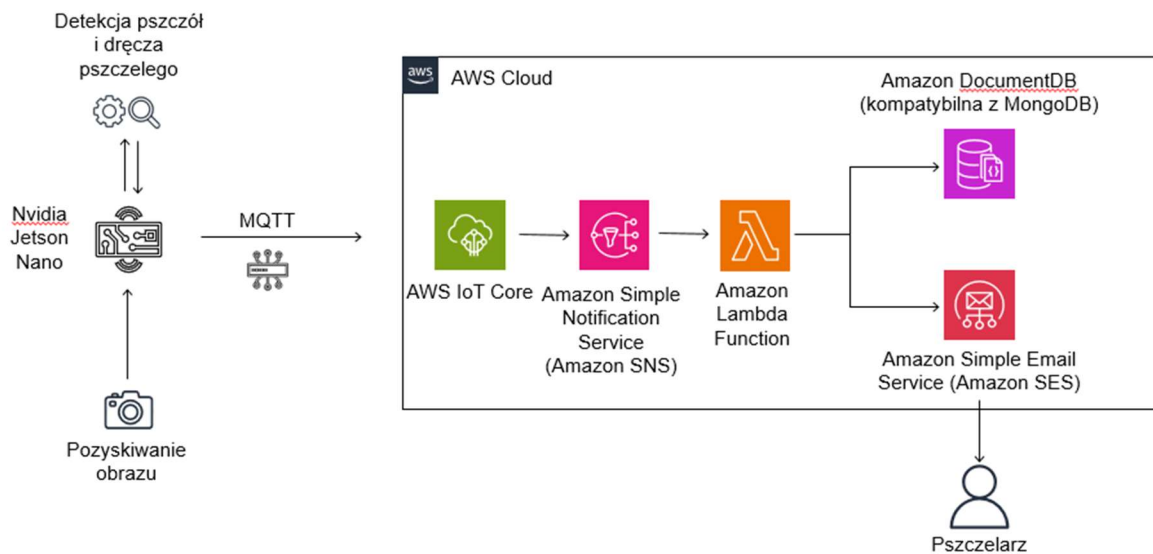
(ang. *ultraviolet*, UV), są bezpieczne dla zdrowia. Zastosowanie takiej kamery przyczynia się do realizacji celów zdefiniowanych w rozdziale 1.2 tej rozprawy, tj. umożliwia działanie systemu w różnych warunkach, jednocześnie nie zaburzając trybu życia i środowiska naturalnego pszczoł.

Połączenie z siecią Internet jest możliwe dzięki zastosowaniu karty sieciowej TP-Link TL-WN725N [117]. Zaletą tej karty sieciowej jest przede wszystkim jej niewielki rozmiar, który w połączeniu z dobrą przepustowością zapewnia stabilne połączenie z siecią i umożliwia przesyłanie obrazów do chmury obliczeniowej, w przypadku, gdy zostanie wykryta obecność dręcza pszczelego.

Oprogramowanie systemu zostało napisane w języku *Python* w wersji 3.8 [118], ze względu na kompatybilność z systemem operacyjnym Ubuntu zainstalowanym na urządzeniu NVIDIA; biblioteką *Tensorflow* [119] umożliwiającą realizację zagadnień uczenia maszynowego i głębokich sieci neuronowych; usługami Amazon AWS umożliwiającymi przechowywanie danych w chmurze. Modele uczenia maszynowego przygotowane zostały zapisane w formacie *.onnx*, (ang. *Open Neural Network Exchange*), który jest dedykowany do zapisu modeli sieci neuronowych i wykorzystywania ich w środowiskach wykonawczych. Następnie, modele zostały przygotowane do uruchomienia jako silniki wnioskowania (ang. *inference*) w środowisku wykonawczym NVidia *TensorRT*, domyślnie zainstalowanym na komputerze jednopłytkowym NVidia Jetson Nano. *TensorRT* umożliwia uruchamianie silników wnioskowania, w odpowiednio dopasowany do możliwości obliczeniowych sprzętu komputerowego sposób, zapewniający niskie opóźnienia oraz wysoką przepustowość aplikacji głębokiego uczenia.

Ogólny schemat działania systemu został przedstawiony na *Rys. 14*. Urządzenie ustawione w odpowiednim miejscu przed ułem wykonuje większość obliczeń na tzw. brzegu (ang. *on-edge*), połączenie z siecią Internet konieczne jest tylko na końcowym etapie przetwarzania danych w przypadku, gdy zostanie wykryte zagrożenie – dręcz pszczeli. W tej sytuacji urządzenie, za pomocą protokołu MQTT łączy się z usługą AWS IoT Core, która uruchamia kolejną usługę w chmurze AWS – SNS (ang. *Simple Notification Service*). Za pomocą tej usługi możliwe jest „powiadomianie” innych usług AWS oraz wysyłanie powiadomień (np. w formie wiadomości e-mail) do subskrybentów. Aby zapewnić użytkownikowi możliwość weryfikacji poprawności detekcji dręcza pszczelego wykonanej na urządzeniu brzegowym, konieczne jest przesłanie mu zdjęcia pszczoły sklasyfikowanej jako chora. Usługa SNS nie umożliwia przesyłania załączników

w powiadomieniach, dlatego w kolejnych etapach przetwarzania w chmurze wywoływana jest usługa AWS Lambda, w której przesyłane są wiadomości email wraz z załącznikiem (do tego kroku wywoływana jest usługa AWS SES (ang. *Simple Email Service*)) oraz zapisywane są dane w bazie danych DocumentDB. Przetwarzanie brzegowe jest jednym z wymagań zdefiniowanych w rozdziale 1.2 tej pracy i jest ono realizowane w ramach proponowanego systemu.



Rys. 14 Architektura systemu umożliwiającego wykrywanie sytuacji zagrożających życiu pszczoł

W ujęciu całościowym, system działa według **Algorytm 1**

Algorytm 1. Opisuje on poszczególne operacje, które są wykonywane przez system. Jako daną wejściową przyjmuje się zmienną $czestotliwoscPowtorzenAA$, która określa, z jaką częstotliwością (co ile przeanalizowanych klatek) ma być powtarzane wykonywanie algorytmu adaptacyjnego wyznaczania okna zagęszczenia, którego celem jest określenie obszaru, w którym odnotowano obecność pszczoł, zdefiniowanego i szczegółowo opisanego w rozdziale 4.4.1.

Algorytm 1 Działanie systemu umożliwiającego wykrywanie sytuacji zagrażających życiu pszczół

```
Wejście:  $czestotliwoscPowtorzenAA > 0, czestotliwoscPowtorzenAA \in N$   
WykonajAlgorytmAdaptacyjny ();  
licznik  $\leftarrow 0$ ;  
foreach  $f_i$  in  $F$  do  
    obraz  $\leftarrow$  WczytajObraz( $f_i$ );  
    pszczoły  $\leftarrow$  WykonajDetekcje(obraz);  
    WyznaczWspolrzedne(pszczoły);  
    obiekty  $\leftarrow$  WyobrebniejObiektyPszczol();  
    licznik ++  
end  
foreach  $o$  in obiekty do  
    czyZdrowa  $\leftarrow$  WykonajKlasyfikacje( $o$ );  
    if czyZdrowa  $\neq$  True then  
        PrzeslijDoChmury( $o$ );  
        PowiadomPszczelarza( $o$ );  
    end  
end  
if licznik =  $czestotliwoscPowtorzenAA$  then  
    WykonajAlgorytmAdaptacyjny ();  
    licznik  $\leftarrow 0$  ;  
end
```

Po pierwszym wykonaniu tego algorytmu rozpoczyna się proces pobierania obrazów – moduł kamery wczytuje obrazy, które są poddawane analizie pod kątem detekcji pszczół. W tym celu wykorzystywany jest odpowiednio wytrenowany model sieci neuronowej. Następnie, wykryte obiekty (pszczoły) są wyodrębniane i zapisywane do osobnych plików. Każdy obraz zawierający pszczołę podlega klasyfikacji pod kątem obecności dręcza pszczelego. Proces ten, podobnie jak w przypadku detekcji pszczół, realizowany jest za pomocą modelu sieci neuronowej. Jeśli obecność dręcza pszczelego zostanie wykryta, oznacza to, że pszczoła jest chora. W takiej sytuacji urządzenie łączy się z siecią Internet i wykonywane są dwie operacje – do użytkownika końcowego – pszczelarza – przesyłana jest informacja o wykryciu obecności dręcza pszczelego, natomiast do chmury obliczeniowej, gdzie przechowywane są obrazy umożliwiające ponowne wytrenowanie modelu, przesyłane jest zdjęcie chorej pszczoły. Każdorazowo, po

wykonaniu wszystkich operacji sprawdzany jest warunek dotyczący ponownego wykonania algorytmu adaptacyjnego wyznaczania okna zagęszczenia, którego cel i sposób działania został szczegółowo przedstawiony w rozdziale 4.4.1. Warunkiem zaprzestania działania powyższych operacji jest wyłączenie skryptu odpowiedzialnego za ich wykonywanie, które następuje po upływie określonego w parametrach systemu czasu.

4.3. Modele sieci neuronowych oparte na technice

Single Shot Detector

Jednym z wymagań zdefiniowanych we Wstępie niniejszej pracy (rozdział 1.2), jest działanie urządzenia IoT w czasie rzeczywistym. Aby je zrealizować, konieczna była analiza modeli sieci neuronowych, a następnie wybór takiego, który będzie najlepiej dostosowany do określonych wcześniej potrzeb. Literatura opisuje wiele różnych rozwiązań dedykowanych zagadnieniom rozpoznawania – detekcji obiektów (ang. *object detection*) na obrazach, wśród najbardziej popularnych wymienia się architektury modeli uczenia głębokiego YOLO (ang. *You Only Look Once*) [120], R-CNN (ang. *Regional Convolutional Neural Network*) oraz ich rozszerzenia – Fast R-CNN [121] i Faster R-CNN [122], SSD (ang. *Single Shot Detector*) [123]. Modele R-CNN przetwarzają obrazy dwuetapowo – w pierwszym kroku wyodrębniane są obszary (ang. *region proposals*, propozycje regionów), które mogą zawierać wyszukiwane obiekty. Następnie, podobne regiony są łączone w większe, by w kolejnych krokach wykonać ekstrakcję cech. W oparciu o te cechy wyodrębnione na początku propozycje regionów są poddawane klasyfikacji. Pomimo bardzo dobrej dokładności – we wstępnych badaniach [124] przeprowadzonych przez R. Girshick et al. na zbiorach danych VOC07 uzyskano wyniki czasowe detekcji jednego obrazu wynoszące 47 sekund (z wykorzystaniem procesora GPU). Badania te pokazały również, że czas potrzebny do wytrenowania tego typu sieci jest bardzo długi, a detekcja obrazów z wykorzystaniem algorytmu R-CNN nie jest możliwa w czasie rzeczywistym. W odpowiedzi na ten problem, ten sam zespół badawczy zaimplementował sieć Fast R-CNN [122], w której jako dane konwolucyjnej sieci neuronowej wyznaczającej cechy poszczególnych obiektów wczytywane są obrazy wejściowe, a nie regiony propozycji. Rozwiązanie to znacznie przyspieszyło czas potrzebny do wytrenowania modelu oraz skróciło średni czas detekcji obrazu z 49 do 0,32

sekund (na tym samym zbiorze testowym). Przełomem była jednak eliminacja wykorzystania algorytmu przeszukiwania selektywnego w celu wyznaczania regionów i wykorzystanie w tym celu osobnej sieci neuronowej, czego wynikiem było skrócenie czasu detekcji obrazu do ułamka sekundy. Tym samym autorzy udowodnili, że sieci oparte o architekturę Faster R-CNN mogą być wykorzystywane w systemach działających w czasie rzeczywistym.

Z kolei, sieci neuronowe bazujące na architekturze SSD opierają swoje działanie na wykonaniu obliczeń sieci neuronowej na obrazie wejściowym tylko raz i na tej podstawie wyliczeniu mapy cech. Natomiast w architekturze YOLO obraz wejściowy dzielony jest na siatki i dla każdej z nich wyznacza wiele predykowanych wartości ramek dla obiektów wraz z prawdopodobieństwami klas dla każdego z nich. Takie podejście skutkuje bardzo szybką i dość dokładną detekcją wielu obiektów jednocześnie. W Tab. 3 przedstawione zostały modele sieci neuronowych umożliwiające detekcję obiektów w czasie rzeczywistym. Wszystkie zostały zbudowane w oparciu o tę samą architekturę sieci neuronowej VGG16 [125], a podane w tabeli dane odnoszą się do detekcji w oparciu o dane ze zbioru PASCAL VOC2007 [126].

Tab. 3 Porównanie modeli sieci neuronowych SSD300, YOLO i Faster R-CNN dedykowanych detekcji obiektów na obrazach (na podstawie [123]).

Model sieci neuronowej	Średnia dokładność detekcji – mAP (ang. mean average precision)	Liczba klatek/s – FPS (ang. frames per second)
SSD300 (VGG16) [123]	72.4	59
YOLO (VGG16) [120]	63.4	45
Faster R-CNN (VGG16) [122]	73.2	7

Z zestawienia wynika, że największą dokładnością charakteryzuje się model sieci Faster R-CNN (przetwarzający obrazy wejściowe o minimalnym rozmiarze krótszego boku wynoszącym 600px [122]). Przetwarza on ponad 6 razy klatek na sekundę mniej w porównaniu do modelu YOLO (do porównania wykorzystano model, który jako dane wejściowe przyjmował obrazu o rozmiarze 448 x 448 px) i ponad 8 razy mniej

w porównaniu do modelu SSD300 (rozmiar obrazu wejściowego dla tego modelu wynosi 300×300 px). Model sieci SSD300 również charakteryzuje się bardzo dobrym poziomem dokładności. Z kolei, porównując modele YOLO i SSD (Rys. 15) [123], można zauważyć, że model SSD posiada dodatkowe warstwy, połączone z ostatnią warstwą modelu bazowego o architekturze VGG16. Sieć VGG16 realizuje proces klasyfikacji obrazów. Dodatkowe warstwy, które kolejno przyjmują mniejsze rozmiary, przewidują wartości przesunięć wyznaczonych (domyślnych) ramek ograniczających (ang. *default bounding box*) w stosunku do ich faktycznych wartości (ang. *ground truth*) o różnych skalach i proporcjach oraz związane z nimi pewności (ang. *confidence*). Powiązanie domyślnych ramek ograniczających z komórkami mapy cech (ang. *feature map*) pozwala na dyskretyzację przestrzeni możliwych ramek otrzymywanych na wyjściu modelu. Dzięki dodatkowym warstwom model SSD300 [123] uzyskuje lepszą precyzję dla mniejszych obrazów wejściowych niż model YOLO [120] przy zachowaniu porównywalnej liczby ramek przetwarzanych w czasie 1 sekundy.

Uwzględniając powyższe aspekty, przy projektowaniu systemu zdecydowano się na implementację modelu MobileNetV2 Lite opartego o metodę Single Shot Detector, który jest dedykowany dla urządzeń IoT przetwarzających obrazy pod kątem detekcji obiektów [127]. Algorytm ten, chociaż charakteryzuje się gorszą dokładnością, działa szybciej niż wersja podstawowa, natomiast sam model sieci jest głęboki na 53 warstwy splotowe i używa 4,2 milionów parametrów. Ponadto, dzięki zastosowaniu odwróconych resztkowych warstw wąskich gardeł (ang. *inverted residual bottleneck layers*) możliwa jest implementacja modelu zapewniająca dobrą wydajność pamięciową urządzenia, na którym wykonywane jest wnioskowanie. Odbywa się to poprzez zbudowanie hiper-grafu obliczeniowego G skierowanego acyklicznie, który składa się z krawędzi reprezentujących operacje i węzłów, które reprezentują tensory (obiekty będące uogólnieniem wektora) obliczeń pośrednich. Obliczenia są zaplanowane w taki sposób, aby zminimalizować liczbę tensorów przechowywanych w pamięci. Matematycznie, całkowitą ilość pamięci obliczeniowej M potrzebnej do wnioskowania na grafie obliczeniowym G można przedstawić wzorem [125]:

$$M(G) = \max_{op \in G} \left[\sum_{A \in op_{inp}} |A| + \sum_{B \in op_{out}} |B| + |op| \right], \quad (1)$$

gdzie:

op jest poszczególną operacją grafu G ,

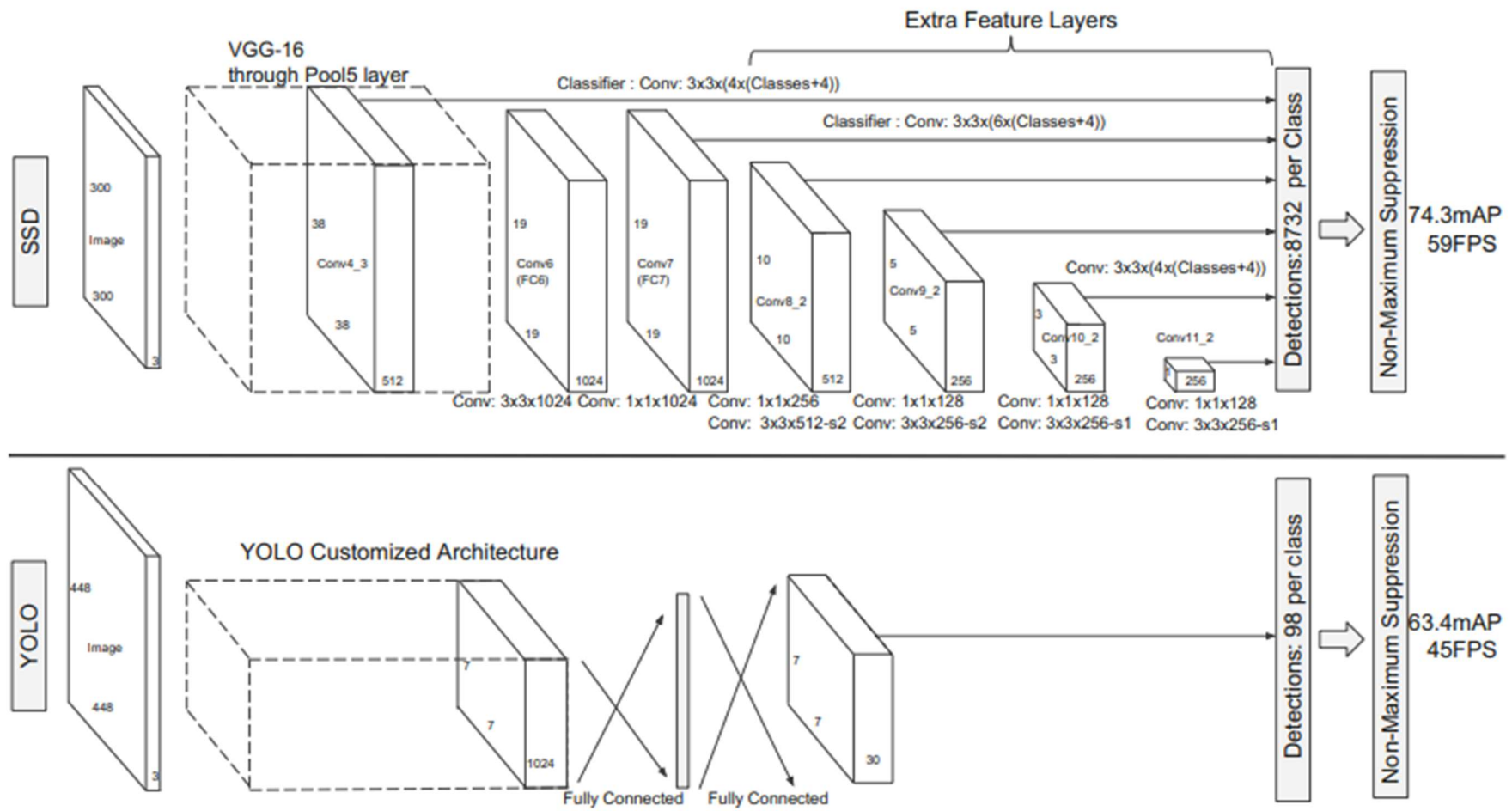
$|A|$ jest rozmiarem tensora wejściowego A ,

$|B|$ jest rozmiarem tensora wyjściowego B ,

$|op|$ ilość pamięci potrzebnej podczas wykonywania operacji op .

Ilość potrzebnej pamięci jest więc obliczana jako maksymalna wartość sumy rozmiarów połączonych danych wejściowych i wyjściowych ze wszystkich operacji.

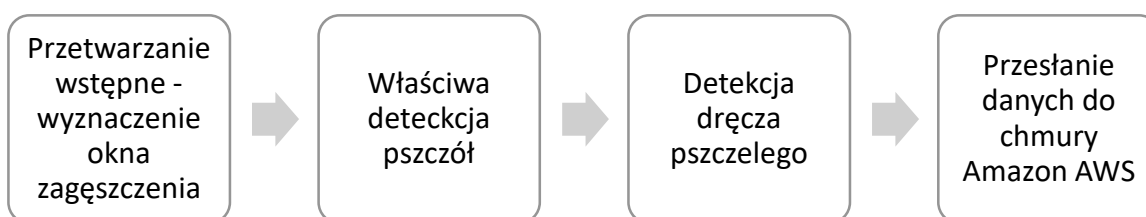
W niniejszej pracy przygotowane zostały modele sieci neuronowych oparte o implementację metody SSD w wersji MobileNetV2 Lite [127], przetwarzające dane wejściowe w rozmiarze 320×320 px zarówno dla problemu wykrywania obiektów pszczoł oraz detekcji dręcza pszczelego. Dla każdego z powyższych problemów detekcji modele zostały przygotowane w procesach uczenia obejmujących różną liczbę epok oraz w oparciu o oddzielne zestawy danych treningowych i testowych, opisanych w rozdziale 5.1 tej rozprawy.



Rys. 15 Porównanie architektury Single Shot Detector i YOLO (źródło: [123])

4.4. Algorytm zasadniczy realizacji procesu wykrywania sytuacji zagrażających życiu pszczół

Realizację procesu wykrywania sytuacji zagrażających życiu pszczół, wykonywanego na urządzeniu brzegowym, można podzielić na cztery główne etapy – przetwarzanie wstępne (ang. *preprocessing*), którego częścią jest wykonanie algorytmu adaptacyjnego wyznaczania okna zagęszczenia; właściwa detekcja pszczół na klatkach filmowych; detekcja dręcza pszczelego oraz przesłanie danych do chmury obliczeniowej i uruchomienie procesu powiadamiania pszczelarza. Schemat blokowy wykonania tego procesu w ramach powyższych etapów przedstawia Rys. 16, natomiast szczegółowy sposób realizacji każdego z nich prezentują kolejne rozdziały.



Rys. 16 Schemat działania urządzenia IoT umożliwiającego wykrywanie dręcza pszczelego

Zarówno proces przetwarzania wstępnego, jak również właściwej detekcji pszczół rozpoczyna się od pobrania odpowiedniej liczby klatek filmu. Film F o liczbie klatek $|F|$ można przedstawić w notacji matematycznej za pomocą poniższego wzoru:

$$F = \{f_i: i = 0 \dots |F| - 1, |F| \in \mathbb{N}^+\}, \quad (2)$$

gdzie:

f_i jest kolejną klatką (ang. *frame*) analizowanego filmu F , i jest indeksem klatki f oraz $i \in \langle 0, K \rangle$, a $K < |F|$ jest liczbą analizowanych klatek, natomiast $|F|$ jest liczbą klatek filmu F .

4.4.1. Algorytm adaptacyjnego wyznaczania okna zagęszczenia

Realizacja procesu wykrywania zagrożeń w kontekście rozmiaru przetwarzanych danych została oparta o dane uzyskiwane w wyniku działania *algorytmu adaptacyjnego wyznaczania okna zagęszczenia*, który opisuje Definicja 1.

Definicja 1 (Algorytm adaptacyjnego wyznaczania okna zagęszczenia). *W niniejszej rozprawie terminem **algorytmu adaptacyjnego wyznaczania okna zagęszczenia** określa się zestaw operacji na obrazach mający na celu wyznaczenie takiego obszaru, na którym w zadanym oknie czasowym pojawiło się najwięcej pszczoł, co pozwala przewidywać, że w kolejnych, następujących po sobie klatkach f_i filmu F na tym obszarze będą znajdowały się obiekty pszczoł.*

Zasada działania algorytmu adaptacyjnego wyznaczania okna zagęszczenia zbliżona jest to metod wykrywania kolizji opartych o haszowanie przestrzenne (ang. *spatial hashing*), wykorzystywanych przy tworzeniu gier komputerowych oraz tzw. map cieplnych (ang. *heat map*), których używa się w różnego rodzaju analizach, np. stron internetowych, w celu określenia obszarów, z którymi użytkownicy mają największą liczbę interakcji. Głównym założeniem działania algorytmu adaptacyjnego wyznaczania okna zagęszczenia jest stworzenie tzw. *siatki*, odwzorowującej zagęszczenie obiektów (pszczoł) na danej klatce filmowej f_i .

Definicja 2 (Siatka obrazu). *Siatka g (ang. *grid*) obrazu w niniejszej pracy stanowi dwuwymiarową reprezentację obrazu bazowego o rozmiarze $s \times w$, zawierającego obiekty (pszczoły), podzieloną na liczbę $c \times r$ pól o rozmiarze $h \times h$, odwzorowującą faktyczny rozmiar analizowanego obrazu wraz z zagęszczeniem obiektów przypadającym na dane pole, gdzie c określa liczbę kolumn (ang. *cols*), a r (ang. *rows*) liczbę wierszy siatki.*

Rozmiar siatki h powinien przyjmować wartości będące wspólnymi dzielnikami składowych s i w rozmiaru obrazu bazowego. Przykładowo, analizując obraz o rozmiarze 1280×720 pikseli ($s = 1280$, $w = 720$), możliwe jest wyznaczenie siatki składającej się z 16 kolumn i 9 wierszy, zawierającej 144 pola o rozmiarze $h = 80$ każde, a następnie przypisanie do każdego z pól siatki liczby obiektów pszczoł znajdujących się w analogicznym położeniu na obrazie bazowym. Sposób przypisania wartości do pól siatki jest następujący: dla zbioru wszystkich obiektów pszczoł P znajdujących się w przestrzeni dwuwymiarowej (X, Y) , należy wyznaczyć współrzędne $p(x, y)$, $x \in X$ i $y \in Y$, a następnie

określić przynależność każdego obiektu do odpowiedniego pola siatki g i zaktualizować liczbę obiektów znajdujących się w danym polu siatki. Wartość pola siatki można opisać:

$$g[j][k] = \sum p(x, y), \quad (3)$$

gdzie:

$p(x, y)$ jest obiektem pszczoły p , znajdującym się w miejscu opisanym współrzędnymi x i y , natomiast j i k reprezentują dane pole siatki:

$$j = \lfloor x/h \rfloor, j \in N, j \in \langle 0, c \rangle \text{ oraz } k = \lfloor y/h \rfloor, k \in N, k \in \langle 0, r \rangle.$$

Siatka g o rozmiarze 16×9 opisana w powyższym przykładzie może wyglądać jak przedstawiona na

Rys. 17. Można z niej odczytać następujące, odpowiadające zagęszczeniu siatki, wartości: dla współrzędnych siatki (6,1) i (5,3) zanotowano po 3 obiekty pszczoł, dla współrzędnych (8,2), (9,2) i (7,3) po 2 obiekty pszczoł, natomiast dla współrzędnych (6,0), (7,1), (6,2), i (7,4) po 1 pszczole. Łącznie w siatce zapisanych zostało 16 obiektów pszczoł. Puste pola w siatce odpowiadają wartościom 0.

id	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0							1									
1							3	1								
2							1		2	2						
3						3		2								
4								1								
5																
6																
7																
8																

Rys. 17 Przykładowa reprezentacja graficzna siatki g

Wyznaczenie siatki i jej zagęszczenia jest konieczne do określenia *okna zagęszczenia*, które na potrzeby niniejszej pracy zdefiniowano następująco:

Definicja 3 (Okno zagęszczenia). Terminem *okna zagęszczenia* określaną jest fragment obrazu bazowego, wyznaczony za pomocą siatki i jej zagęszczenia, odwzorowujący obszar

obrazu, na którym, po przeanalizowaniu zadanej liczby następujących po sobie klatek (obrazów) m , zaobserwowano liczbę obiektów (pszczoł) odpowiadającą założonej wartości progowej t .

Obszar okna zagęszczenia oz wyznaczany jest na podstawie współrzędnych pól siatki, w których zanotowano największą liczbę pszczoł w kolejnych klatkach analizowanego strumienia obrazów (filmu). Liczba analizowanych klatek m może (ale nie musi) być równa całkowitej liczbie klatek filmu (w rozdziale 5.5.1 przedstawiono, w jaki sposób dobór wartości m wpływa na rozmiar wyznaczonego okna zagęszczenia). Każdorazowo, po wyznaczeniu pola siatki z największą liczbą pszczoł, współrzędne okna zagęszczenia są aktualizowane. Okno zagęszczenia oz można opisać matematycznie za pomocą dwuelementowej tablicy współrzędnych (x_{min}, y_{min}) , (x_{max}, y_{max}) , które stanowią odpowiednio lewy górny i prawy dolny róg wyznaczonego obszaru okna zagęszczenia:

$$oz = [(x_{min}, y_{min}), (x_{max}, y_{max})], \quad (4)$$

gdzie:

$$x_{min}, x_{max} \in \langle 0, s \rangle; \quad y_{min}, y_{max} \in \langle 0, w \rangle; \quad x_{min}, x_{max}, y_{min}, y_{max} \in N.$$

Wartość progowa t określana jest za pomocą wzoru:

$$t = a \times sredniaLiczbaPszczol \times m \quad (5)$$

gdzie:

$a \in (0,1)$ i jest współczynnikiem regulacji wartości progowej t ,

$sredniaLiczbaPszczol$ opisuje średnią liczbę pszczoł przypadającą na daną klatkę, a wartość m odpowiada liczbie przetwarzanych w procesie wyznaczania okna zagęszczenia klatek.

Przykładowe okno zagęszczenia w odniesieniu do obrazu bazowego zostało przedstawione na Rys. 18 (czerwoną ramką zaznaczono obszar okna zagęszczenia).

Ze względu na naturę problemu – w zależności od wielu czynników (pory dnia, pogody, wielkości czy stanu zdrowia pszczelej rodziny) – na pozyskiwanych obrazach może pojawiać się zróżnicowana liczba pszczoł. Aby ograniczyć występowanie sytuacji, w której wyznaczone okno zagęszczenia przechodzi przez dużą część obiektu (pszczoły), a na pozyskiwanych obrazach pojawia się niewielka liczba pszczoł, zdefiniowany został kolejny parametr algorytmu adaptacyjnego – *liczba siatek*.



Rys. 18. Przykładowe okno zagęszczenia (zaznaczone na czerwono) w odniesieniu do obrazu bazowego.

Definicja 4 (Liczba siatek). *Liczba siatek jest parametrem algorytmu adaptacyjnego wyznaczania okna zagęszczenia, który określa sposób, w jaki wyliczane będą poszczególne wartości współrzędnych okna zagęszczenia i jest ściśle związana z wartością zmiennej **liczbaPszczol** opisującą liczbę obiektów znalezionych na analizowanej klatce.*

Wartość parametru *liczbaSiatek* jest określana w procesie przetwarzania wstępnego algorytmu adaptacyjnego wyznaczania okna zagęszczenia, którego działanie opisuje **Algorytm 2**.

Celem przetwarzania wstępnego jest określenie sposobu wyznaczania okna zagęszczenia oraz przygotowanie siatek (utworzenie obiektów siatek i ich inicjalizacja wartościami 0). W sytuacji, w której na obrazie widoczna jest duża liczba pszczoł (*liczbaPszczol* > $lp=10$, wartość stałej $lp = 10$ została uzasadniona w rozdziale 5.4), nie jest konieczne aż tak dokładne wyznaczenie siatki, jak w przypadku małej liczby obiektów. Wynika to z faktu, że mała liczba pszczoł przed ułem wskazuje na dużą ich rotację – pszczoły pracują i ich obecność w polu widzenia kamery ogranicza się do kilku/kilkunastu sekund. W przypadku dużej liczby pszczoł przed ułem mamy do czynienia m.in. z sytuacją, w której wychodzą one na ścianę ula z otworem wlotowym po to, by ochłodzić gniazdo i ich rotacja na kolejnych obrazach z kamery jest niewielka, a czas przebywania w polu widzenia kamery wydłuża się.

Algorytm 2 Przetwarzanie wstępne algorytmu adaptacyjnego wyznaczania okna zagęszczenia

```
Wejście:  $F[0]$  /* pierwsza klatka analizowanego filmu F
Wyjście: siatki
liczbaSiatek  $\leftarrow 1$ ;
obraz  $\leftarrow$  WczytajObraz( $F[0]$ );
pszczoly  $\leftarrow$  WykonajDetekcje(obraz);
liczbaPszczol  $\leftarrow$  len(pszczoly);
if liczbaPszczol  $\leq$  lp then
  | liczbaSiatek  $\leftarrow 4$ ;
end
siatki  $\leftarrow$  PrzygotujSiatki(liczbaSiatek);
```

Analizując liczbę pszczół pod kątem sposobu wyznaczania siatki, należy więc przyjąć, że w przypadku niewielkiej liczby pszczół należy wyznaczyć pozycję każdego obiektu bardzo dokładnie – dla każdej kombinacji wartości współrzędnych $xmin$, $ymin$, $xmax$, $ymax$ wyznaczane są siatki, następnie wartości okna zagęszczenia wyliczane są w oparciu o wartości skrajne ze wszystkich siatek. Natomiast w przypadku dużej liczby pszczół, pozycja obiektu może być reprezentowana przez uśrednione wartości jego współrzędnych. Wówczas obliczana jest tylko jedna siatka i na jej podstawie wyznaczane są wartości okna zagęszczenia.

Po wykonaniu przetwarzania wstępnego algorytmu adaptacyjnego i określeniu liczby siatek, które mają być tworzone do wyznaczenia okna zagęszczenia oraz ich inicjalizacji, wczytywany jest plik konfiguracyjny zawierający informację o początkowym rozmiarze obrazu $s \times w$ (1280×720 px), liczbie klatek m , która ma zostać użyta dla algorytmu adaptacyjnego, wartość a , będącą współczynnikiem regulacji wartości progowej t (ang. *threshold*) warunkującej zakończenie procesu rozszerzania powierzchni okna zagęszczenia, rozmiar siatki h , a także aktualne wartości okna zagęszczenia $xmin$, $xmax$, $ymin$, $ymax$. Dane te są jednocześnie danymi wejściowymi dla algorytmu adaptacyjnego wyznaczania okna zagęszczenia, którego działanie przedstawia **Algorytm 3**.

Algorytm 3 Działanie algorytmu adaptacyjnego wyznaczenia okna zagęszczenia

```

Wejście:  $m, a, h, w, s, F$ 
Wyjście:  $oz$ 
 $c \leftarrow \text{floor}(s/h)$ ; /* liczba kolumn siatki */
 $r \leftarrow \text{floor}(w/h)$ ; /* liczba wierszy siatki */

/* wyznaczenie liczby siatek i utworzenie obiektów siatek z wartościami
   początkowymi = 0 */
siatki  $\leftarrow$  WyznaczIPrzygotujSiatki( $F[0]$ );
liczbaSiatek  $\leftarrow$  len(siatki);
listy  $\leftarrow$   $\emptyset$ ;
oknaz  $\leftarrow$   $\emptyset$ ; /* Okna zagęszczenia */
i  $\leftarrow$  0;
liczbaPszczol  $\leftarrow$  0;

while  $i < m$  do
    obraz  $\leftarrow$  WczytajObraz( $F[i]$ );
    pszczoly  $\leftarrow$  WykonajDetekcje(obraz);
    WyznaczWspolrzedneOrazZaktualizujSiatki(pszczoly);
    liczbaPszczol  $\leftarrow$  liczbaPszczol + len(pszczoly);
    i ++;
end
sredniaLiczbaPszczol  $\leftarrow$  floor(liczbaPszczol/ $m$ );
 $t \leftarrow a * \text{sredniaLiczbaPszczol} * m$ ;

for  $id \leftarrow 0$  to liczbaSiatek do
    lista  $\leftarrow$  PrzygotujListeObiektow(siatki[id]);
    ostatniElementListy  $\leftarrow$  lista.pop();
    xmin  $\leftarrow$  ostatniElementListy[0][0];
    ymin  $\leftarrow$  ostatniElementListy[0][1];
    xmax  $\leftarrow$  xmin;
    ymax  $\leftarrow$  ymin;
    liczbaPszczolZSiatki  $\leftarrow$  lista[1];
    while liczbaPszczolZSiatki  $\leq$  liczbaPszczol  $\times t \wedge$  lista  $\neq \emptyset$  do
        aktualnePoleSiatki  $\leftarrow$  lista.pop();
        xmin  $\leftarrow$  min(xmin, aktualnePoleSiatki[0][0]);
        ymin  $\leftarrow$  min(ymin, aktualnePoleSiatki[0][1]);
        xmax  $\leftarrow$  max(xmax, aktualnePoleSiatki[0][0]);
        ymax  $\leftarrow$  max(ymax, aktualnePoleSiatki[0][1]);
        liczbaPszczolZSiatki  $\leftarrow$  ZaktualizujLiczbePszczol(xmin, ymax, siatka[id]);
    end
    oknaz.append([xmin, ymin, xmax, ymax])
end
oz[xmin]  $\leftarrow$  min(oknaz.xmin)  $\times h$ ;
oz[ymin]  $\leftarrow$  min(oknaz.ymin)  $\times h$ ;
oz[xmax]  $\leftarrow$  max(oknaz.xmax)  $\times h$ ;
oz[ymax]  $\leftarrow$  max(oknaz.ymax)  $\times h$ ;

```

W pierwszych krokach definiowane są potrzebne zmienne, następnie do tych zmiennych przypisywane są wartości początkowe – zera oraz puste tablice. Kolejnym etapem jest wykonanie detekcji pszczoł na zadanej liczbie klatek, wyznaczenie współrzędnych tych obiektów i przypisanie ich do odpowiednich pól w siatce bądź siatkach. Każdorazowo, po zakończeniu procesu detekcji pszczoł na danym obrazie

aktualizowana jest ich suma. Następnie dla każdej zdefiniowanej siatki wyznaczane są listy składające się z elementów e :

$$e = [[j, k], n],$$

gdzie:

j, k – są współrzędnymi pola siatki, natomiast wartość n – odpowiada liczbie pszczoł w danym polu.

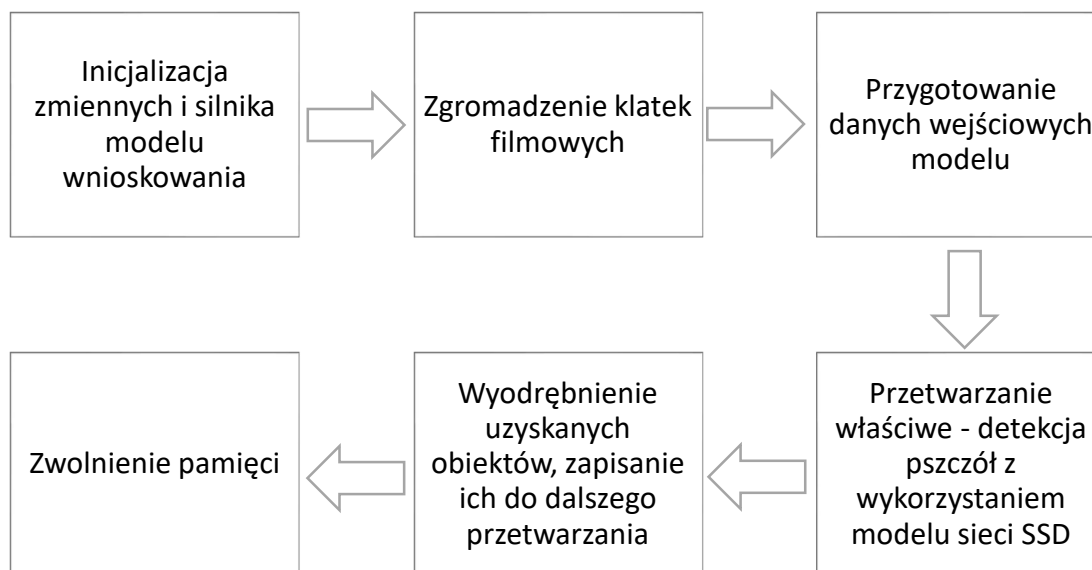
Każda lista posortowana jest rosnąco według liczby pszczoł znajdujących się w danym polu siatki. Po przygotowaniu listy wyznaczane są początkowe wartości współrzędnych $xmin, ymin, xmax, ymax$, które odpowiadają wartościom współrzędnych pola siatki o największej liczbie pszczoł. Następnie, współrzędne te są aktualizowane – obszar *okna* (lub okien, w przypadku wykonywania obliczeń dla czterech siatek) *zagęszczenia* rozszerza się o kolejne pola siatki, dopóki nie zostanie znaleziona pożądana liczba pszczoł, odpowiadająca wartości progowej t , a liczba elementów listy będzie większa od 0. Końcowym etapem działania algorytmu adaptacyjnego jest wyznaczenie współrzędnych *okna zagęszczenia*, które są wyznaczane odpowiednio jako wartości minimalne i maksymalne wszystkich *okien zagęszczenia* $okna_z$, pomnożone przez rozmiar siatki h . Po obliczeniu obszaru *okna zagęszczenia* oz uzyskane współrzędne są zapisywane do pliku konfiguracyjnego, aby skrypt odpowiedzialny za pobieranie i przetwarzanie obrazów mógł działać szybciej i przetwarzać odpowiednio dużą liczbę klatek.

4.4.2. Właściwa detekcja pszczoł na klatkach filmowych

Po wyznaczeniu *okna zagęszczenia* przez algorytm adaptacyjny i określeniu, liczby klatek, którą należy pomijać, urządzenie rozpoczyna proces właściwej detekcji obiektów pszczoł na klatkach filmowych, zilustrowany na *Rys. 19*.

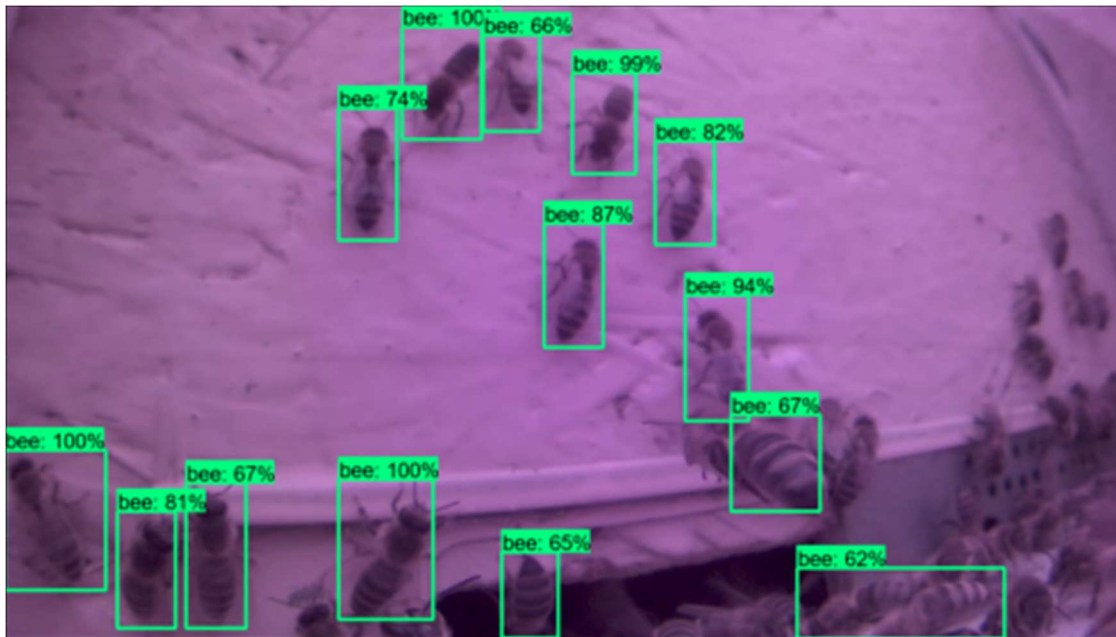
Pierwszym etapem działania algorytmu jest inicjalizacja zmiennych wykorzystywanych przez algorytm (między innymi dotyczących liczby przetwarzanych klatek czy obszaru *okna zagęszczenia*), oraz inicjalizacja silnika modelu detekcji pszczoł. Po zgromadzeniu odpowiednio dużej liczby klatek K (której wartość została opisana i uzasadniona w rozdziale 5.2), rozpoczyna się etap przygotowywania obrazów do przetwarzania przez model sieci neuronowej. W tym celu zebrane obrazy przekształcane są do rozmiaru oczekiwanego na wejściu modelu, tj. 320×320 px i przygotowywane są wymagane przez model obiekty – tensory, które będą wykorzystywane w procesie

wnioskowania. Następnie rozpoczyna się właściwa detekcja pszczół z wykorzystaniem modelu uczenia maszynowego SSD MobileNetV2 Lite. Zwrócone wyniki, z uwzględnieniem obszaru okna zagęszczenia, są zapisywane do listy obiektów, które na dalszym etapie działania systemu będą wykorzystywane w procesie detekcji obecności dręcza pszczelego. Ostatnim krokiem przetwarzania danych związanych z detekcją pszczół jest zwolnienie pamięci, w której przechowywane były analizowane klatki.



Rys. 19 Etapy właściwej detekcji pszczół na klatkach filmowych

Przykładowo, na *Rys. 20* przedstawiono klatkę z zaznaczonymi obiektami, które zostały zidentyfikowane jako pszczoły w procesie właściwej detekcji pszczół, wraz z pewnością klasyfikatora dotyczącą każdej z detekcji.



Rys. 20 Przykładowy obraz z zaznaczonymi obiektami wykrytymi w procesie detekcji pszczół

4.4.3. Detekcja dręcza pszczelego

Proces detekcji dręcza pszczelego na ciele pszczoły, opisany przez **Algorytm 4**, w swoim działaniu jest zbliżony do działania algorytmu umożliwiającego właściwą detekcję pszczół na klatkach filmowych.

Algorytm 4 Detekcja dręcza pszczelego

```

Wejście: wyodrebioneObiektyPszczol
PrzygotujSilnikDetekcji ();
PrzygotujDaneWejsciove (wyodrebioneObiektyPszczol);
foreach o in wyodrebioneObiektyPszczol do
    | czyZdrowa ← WykonajDetekcjeDrecza(o);
    | if czyZdrowa ≠ True then
    |     | NawiazPolaczeniezAWS ();
    |     | PrzeslijDane (o);
    | end
end
end

```

Pierwszym krokiem jest przygotowanie silnika detekcji dręcza pszczelego opartego na modelu uczenia maszynowego wykorzystującego technikę SSD. Następnie, wyodrębnione w trakcie procesu detekcji pszczół (opisanego w poprzednim rozdziale

4.4.2) obiekty owadów są przetwarzane w celu przygotowania ich do formatu danych wejściowych, których oczekuje na wejściu silnik wnioskowania umożliwiający detekcję dręcza pszczelego. Podobnie jak w przypadku silnika detekcji pszczół, jest to obraz przekształcony do rozmiaru 320 x 320 px. Każdy z obiektów (pszczół) podlega klasyfikacji ze względu na obecność dręcza pszczelego na ciele pszczoły. Jeśli w wyniku wnioskowania modelu wykryty zostanie dręcz pszczeli, uruchamiany jest algorytm odpowiedzialny za nawiązanie połączenia z usługami chmury Amazon AWS. Kiedy połączenie zostanie ustanowione, do chmury wysyłana jest wiadomość zawierająca informację o wykryciu chorej pszczoły wraz z datą i godziną jej wykrycia oraz zakodowanym załącznikiem – zdjęciem. W chmurze uruchamiane są usługi odpowiedzialne za zapisanie obrazu do bazy danych oraz przesłanie do pszczelarza wiadomości e-mail z załączonym zdjęciem pszczoły, na ciele której wykryta została obecność dręcza pszczelego.

4.5. Miary oceny jakości modeli

Do oceny jakości wyników zwracanych przez każdy z modeli, została określona macierz pomyłek/błędów (ang. *confusion matrix*) przedstawiona w Tab. 4. Zgodnie z tą tabelą wartości zwracane przez klasyfikator (np. wykrywający obecność dręcza pszczelego na ciele pszczoły) można ocenić jako:

- prawdziwie ujemne TN (ang. *True Negative*), kiedy pszczoła jest zdrowa i model nie wykrył dręcza pszczelego),
- prawdziwie dodatnie TP (ang. *True Positive*), kiedy na pszczole widoczny jest dręcz pszczeli i model sklasyfikował ją jako chorą,
- fałszywie ujemne FN (ang. *False Negative*), kiedy pszczoła jest chora, ale model nie wykrył obecności dręcza pszczelego i fałszywie określił ją jako zdrową,
- fałszywie dodatnie FP (ang. *False Positive*), kiedy pszczoła zdrowa została sklasyfikowana jako chora.

Tab. 4 Macierz pomyłek

		Wartości rzeczywiste	
		0	1
Wartości predykowane	0	prawdziwie ujemne (TN)	falszywie ujemne (FN)
	1	falszywie dodatnie (FP)	prawdziwie dodatnie (TP)

Na podstawie macierzy pomyłek obliczono następujące metryki, które posłużyły do oceny wyników uzyskanych w procesach detekcji obiektów pszczoł i dręcza pszczelego:

- a. dokładność (ACC – ang. *accuracy*), definiowana jako stosunek poprawnie sklasyfikowanych obiektów do wszystkich obiektów, wyrażona wzorem (6):

$$ACC = \frac{TP + TN}{TP + TN + FP + FN}, \quad (6)$$

- b. precyzja (PPV – ang. *positive predictive value*), określająca stosunek obiektów sklasyfikowanych jako prawdziwe do faktycznie prawdziwych (7):

$$PPV = \frac{TP}{TP + FP}, \quad (7)$$

- c. czułość (TPR – ang. *true positive rate*), obliczana jako iloraz wartości prawdziwie dodatnich i sumy wartości prawdziwie dodatnich i falszywie ujemnych (8):

$$TPR = \frac{TP}{TP + FN}, \quad (8)$$

- d. miara F1, będąca średnią harmoniczną precyzji i czułości (9):

$$F_1 = \frac{2 \times PPV \times TPR}{PPV + TPR} = \frac{2 \times TP}{2 \times TP + FP + FN}, \quad (9)$$

Wartości powyższych metryk zostały wyliczone dla modeli z różną wartością parametru *IoU* (ang. *Intersection over Union*, część wspólna do całości) określanej też jako współczynnik podobieństwa Jaccarda. Jest to stosunek części wspólnej ramki ograniczającej zwróconej przez klasyfikator oraz rzeczywistej ramki ograniczającej obiekt (określonej np. za pomocą współrzędnych w zestawie danych treningowych lub testowych)

do sumy pól obu ramek. W teorii zbiorów indeks Jaccarda wyraża się następującym wzorem (10):

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|}, \quad (10)$$

gdzie: J jest wartością indeksu Jaccarda, natomiast A, B są analizowanymi zbiorami.

Graficznie wartość parametru IoU można przedstawić następująco:

$$IoU = \frac{\text{Część wspólna pól ramek}}{\text{Suma pól obu ramek}} = \frac{\img alt="Diagram illustrating the Intersection over Union (IoU) metric. It shows two overlapping blue squares. The intersection of the two squares is shaded in a darker blue, representing the numerator of the IoU formula. The union of the two squares is outlined in a lighter blue, representing the denominator of the formula." data-bbox="760 250 820 332"/>$$

Im wyższa jest wartość parametru IoU , tym obiekty zwrócone przez klasyfikator dokładniej odwzorowują pozycje rzeczywistych obiektów na danym obrazie.

Aby określić jakość klasyfikacji pszczoły jako zdrową bądź chorą obliczone zostały dwie dodatkowe metryki:

- a. specyficzność, swoistość (TNR – ang. *true negative rate*), obliczana jako iloraz wartości prawdziwie ujemnych i sumy wartości prawdziwie ujemnych oraz fałszywie dodatnich (11):

$$TNR = \frac{TN}{TN + FP}, \quad (11)$$

oraz

- b. współczynnik korelacji Matthews'a MCC (ang. *Matthews Correlation Coefficient*), który w literaturze występuje także jako współczynnik ϕ , wyliczany według następującego wzoru (12):

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}. \quad (12)$$



5. Wyniki eksperymentów badawczych

W niniejszym rozdziale opisano dane, których użyto do realizacji eksperymentów badawczych, sposób wykonania badań oraz uzyskane wyniki. Przeprowadzone eksperymenty badawcze można podzielić ze względu na przedmiot badań na trzy grupy – detekcja obiektów pszczół na przetwarzanym obrazie, klasyfikacja pszczoły (jako chorą – z obecnym dręczeniem pszczelim, albo zdrową) oraz wyznaczenie okna zagęszczenia, które wyznacza obraz bazowy analizowany w kolejnych etapach działania systemu. Ponadto, dla każdego z etapów procesu wykrywania zagrożeń dokonano pomiarów czasu. Realizacja problemów badawczych dotyczących detekcji obiektów (zarówno pszczoły jak i dręcza pszczelego) została zrealizowana z wykorzystaniem modelu sieci neuronowej o architekturze SSD MobileNetV2 Lite.

5.1. Dane wykorzystane do badań

Dane wykorzystane do eksperymentów – filmy – były nagrywane w ciągu dnia, o różnych porach, w miesiącach letnich i jesiennych (od czerwca do początku października). Ze względu na specyfikę problemu – pszczoły rzadko opuszczają ule w dni deszczowe, zimne i przy bardzo dużym zachmurzeniu – warunki pogodowe panujące w czasie nagrywania filmów były zróżnicowane w umiarkowanym zakresie – było słonecznie, stopień zachmurzenia wahał się od niemal zerowego do średniego. Zmianom ulegał również poziom nasłonecznienia – w zależności od pory dnia (godziny przed i popołudniowe) ule były bezpośrednio oświetlone bądź też zacienione. Kamera, która została wykorzystana do pozyskiwania filmów ustawiana była w odległości ok. 20 cm od otworu wlotowego ula, w taki sposób, aby nie ingerować w tor lotu pszczoły oraz umożliwić im swobodne poruszanie się w sąsiedztwie ula i na jego powierzchni. Większe odległości powodowały, że obiekty pszczoły były zbyt małe, natomiast mniejsze ograniczały obszar rejestrowany przez kamerę. Filmy były zapisywane co 60 sekund, w formacie *.avi*, w rozdzielczości: 1280×720 px, częstotliwość pobierania klatek wynosiła 10 klatek na sekundę (wstępne badania, opisane w artykule [110] pokazały, że

analiza obrazów w wyższych rozdzielczościach mocno wpływa na czas przetwarzania klatki i aby utrzymać czas rzeczywisty obliczeń konieczne jest pomijanie klatek).

Z filmów zostały wyodrębnione zdjęcia, zapisane w formatach *.png*, Do przygotowania, a następnie przetestowania modelu detekcji pszczoł wybranych zostało 100 obrazów, na których zaznaczonych zostało 1463 obiektów pszczoł. 1399 z nich stanowiły osobniki zdrowe, pozostałe 64 - chore. Na analizowanych obrazach znajdowała się różna liczba pszczoł, wahająca się od 4 do 35 pełnych i wyraźnych sylwetek. Na potrzeby niniejszej pracy zostały określone następujące przedziały (dla których zasadność wyznaczonych wartości została potwierdzona w rozdziale 5.4) opisujące liczbę pszczoł na zdjęciach:

- *mało* – liczba pszczoł $\in (0, 10)$,
- *dużo* – liczba pszczoł > 10 .

W związku z brakiem nagrań sprzed ula zawierających zaatakowane dręczeniem pszczoły, dane zostały przygotowane w taki sposób, aby w jak najlepszym stopniu odwzorować rzeczywistość. W tym celu, przy pomocy pszczelarzy, zebrane zostały znalezione w ulu (martwe) osobniki dręcza pszczelego, następnie naniesiono je na zdrowe pszczoły. Tak przygotowanym pszczołom wykonano zdjęcia (na tle zbliżonym do koloru i faktury ula, z odległości ok. 20 cm). W tym celu wykorzystano kamerę, będącą elementem urządzenia IoT opisanego w rozdziale 4.2. Warto zaznaczyć, że proces przeprowadzony został przy zachowaniu szczególnej dbałości o pszczoły, a po zakończeniu wykonywania zdjęć usunięto dręcza pszczelego z pszczoł i umożliwiono im powrót do ula.

Następnie „chore” pszczoły zostały wklejone w losowe miejsca na wybranych obrazach, ponadto wycięto obiekty dręcza pszczelego i naniesiono je na niektórych zdrowych pszczołach znajdujących się w zestawie wybranych obrazów. Obiekty pszczoł – obszary zainteresowania (ang. *region of interests*, ROI) były zaznaczane z wykorzystaniem programu *LabelImg* i zapisywane w formacie PASCAL VOC (jest to rodzaj formatu *.xml*, wykorzystywany w zagadnieniach uczenia maszynowego do rozwiązywania problemów dotyczących detekcji obiektów), natomiast nanoszenie obiektów pszczoł i dręcza pszczelego na klatki wyodrębnione z filmów zostało zrealizowane za pomocą programu Adobe Photoshop®.

Do wytrenowania modelu umożliwiającego klasyfikację pszczół na zdrowe i chore (te z obecnym na pszczole dręczeniem pszczelim) posłużyły wyodrębnione w poprzednim kroku zdjęcia przedstawiające pojedyncze pszczoły. Ze względu na niewielką liczbę obrazów wykonano proces augmentacji danych, do którego posłużyła biblioteka *albumations* [128]. Zwielokrotniając liczbę obrazów, wykonano losowo następujące transformacje, uwzględniając przy tym położenie zaznaczonych wcześniej obiektów dręcza pszczelego:

- obrót o 90° , zastosowany z prawdopodobieństwem 0.5,
- odbicie lustrzane w pionie, zastosowany z prawdopodobieństwem 0.5,
- odbicie lustrzane w poziomie, zastosowany z prawdopodobieństwem 0.5,
- zmiana kontrastu obrazu, zastosowana z prawdopodobieństwem 0.2.

W wyniku wykonania powyższych operacji uzyskano 6050 zdjęć pszczół, z czego połowę stanowiły pszczoły zdrowe, a połowę chore.

5.2. Detekcja pszczół na klatkach filmowych

W procesie detekcji pszczół został wykorzystany model *SSD MobileNet V2* w wersji Lite [127], przetwarzający obrazy wejściowe zeskalowane do rozmiaru 320x320 px. Jest to rozmiar wymagany przez wybrany przygotowany model, i jest on wystarczający do realizacji zagadnień detekcji obiektów [123]. Należy przy tym pamiętać, że obiekty pszczół znalezione w tym procesie są wyodrębniane z obrazu o rozmiarze okna zagęszczenia. Takie podejście umożliwia realizację procesu detekcji dręcza pszczelego w oparciu o zdjęcia pszczół dobrej jakości i oryginalnym rozmiarze.

Do wytrenowania modelu posłużył zbiór 75 ze 100 obrazów, natomiast w procesie testowania wykorzystanych zostało 25 obrazów wyodrębnionych z różnych filmów, na których zaznaczono łącznie 360 obiektów pszczół. Wstępny proces uczenia został przeprowadzany dla 2000 epok (kroków), jednak uzyskany model nie spełniał założeń dotyczących jakości identyfikacji pszczół, tj. wyniki były gorsze niż uzyskane we wcześniejszym etapie badań z wykorzystaniem urządzeń opisanych w rozdziałach 4.1.1 oraz 4.1.2. Proces uczenia został więc przeprowadzony dwukrotnie, dla 3000 epok (kroków)

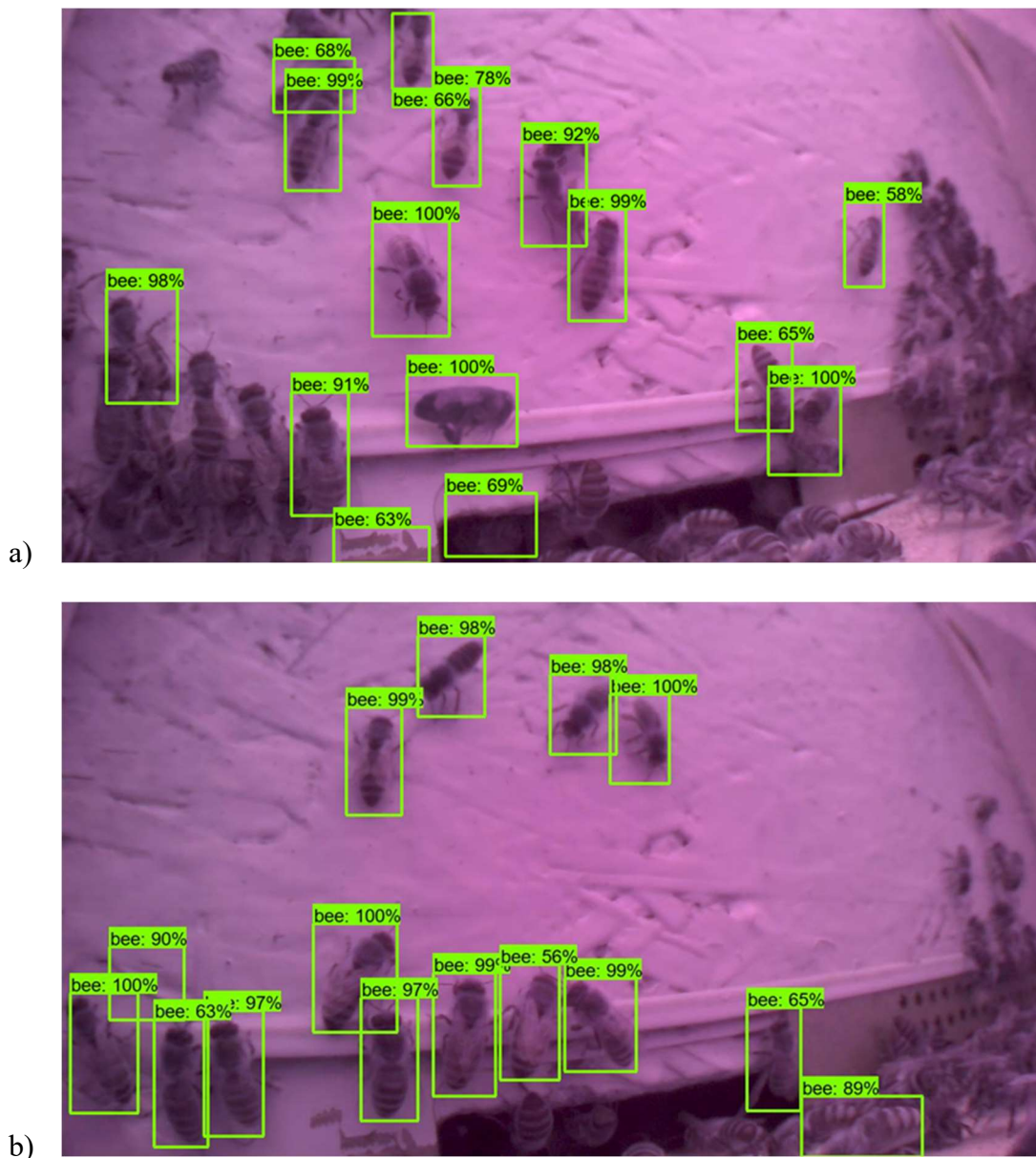
oraz dla 4000 epok (kroków), za każdym razem wykorzystywano ten sam zestaw danych oraz taki sam plik konfiguracyjny. W procesie walidacji modeli obliczano macierz pomyłek dla każdego z obrazów, dla dwóch wartości parametru IoU , tj. $IoU \geq 0.5$ i $IoU \geq 0.8$. Na podstawie zastosowanych parametrów zostały określone robocze nazwy modeli – *3k05* dla modelu trenowanego w trakcie 3000 epok i wartością parametru $IoU \geq 0.5$ i analogicznie dla kolejnych modeli: *3k08*, *4k05* i *4k08*, które są stosowane do opisu modeli w dalszej części rozprawy. Należy zaznaczyć, że w przypadku procesu detekcji wielu obiektów na jednym obrazie, wartości prawdziwie ujemne nie będą wyliczane – klasyfikator będzie bowiem zwracał tylko te obiekty, które określi (z określonym prawdopodobieństwem) jako zadaną klasę, pozostałe fragmenty obrazu będą traktowane jako tło. Wszystkie wartości macierzy pomyłek dla walidacji modeli z uwzględnieniem powyższych parametrów zostały przedstawione w Tab. 5. Na potrzeby dalszych analiz przyjmuje się następujące oznaczenia modeli (odpowiadające kolejnym kolumnom Tab. 5 – *model 3k05*, *model 3k08*, *model 4k05*, *model 4k08*. Liczbą obiektów określono faktyczne obiekty pszczoł występujące na obrazach i zaznaczone w plikach *.xml*, w odniesieniu do których porównywano wartości zwrócone przez modele (opisane etykietą *liczba detekcji*). Dla zachowania czytelności prezentowanych danych analizowane obrazy zostały oznaczone numerami z zakresu 1 – 25.

Z zadanej tabeli można odczytać, dla większości analizowanych obrazów liczba detekcji obiektów była większa, jeśli użyty został model zbudowany w procesie 4000 epok. Z kolei wraz ze zwiększaniem wartości IoU malała liczba obiektów fałszywie zidentyfikowanych jako pszczoły. Liczba rzeczywistych obiektów na zdjęciach nie miała bezpośredniego wpływu na poszczególne miary jakości – modele gorzej radziły sobie w przypadkach, gdy zagęszczenie pszczoł było duże, a ich sylwetki na siebie nachodziły. Taką sytuację przedstawiono na Rys. 21.

Dokładność, z jaką modele *4k05* i *4k08* znajdowały liczbę pszczoł była w większości przypadków wyższa niż odpowiednie wartości zwracane przez modele *3k05* i *3k08*, co obrazuje wykres pokazany na Rys. 22. Z wykresu można odczytać, że dokładność dla modelu *4k05* dla sporej liczby analizowanych obrazów oscylowała w okolicy 0.8, natomiast odpowiadającego mu pod kątem wartości parametru IoU *3k05* częściej osiągała wartości powyżej 0.7. Średnie wartości dokładności każdego modelu wynoszą odpowiednio: 0.77 dla modelu *3k05*, 0.70 dla modelu *3k08*, 0.83 dla modelu *4k05*, oraz 0.78 dla modelu *4k08*.

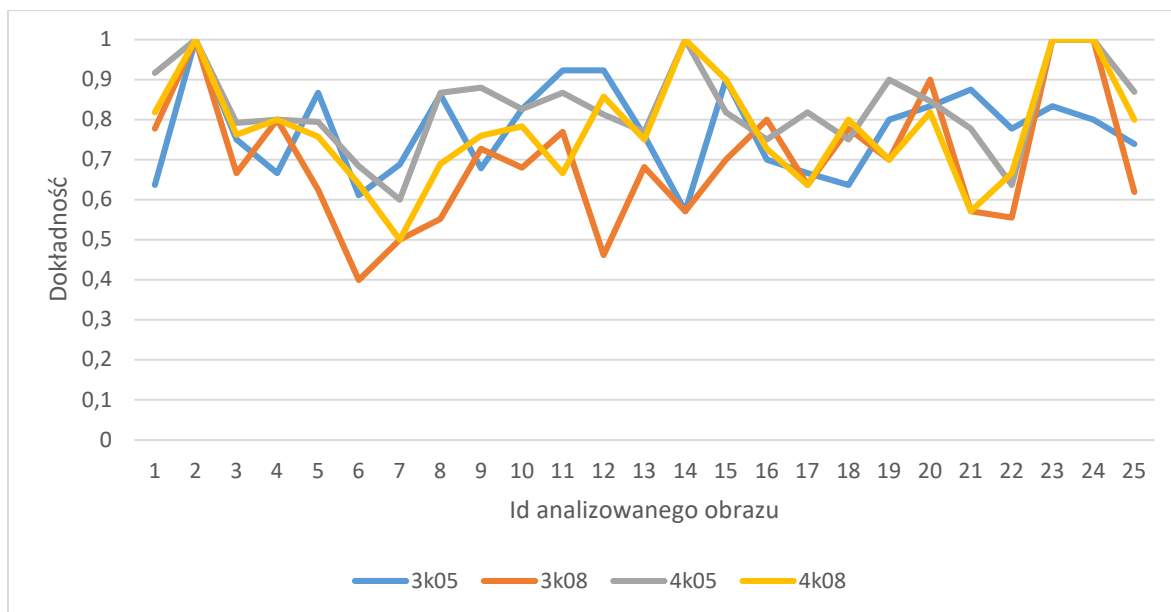
Tab. 5 Macierze pomyłek dla wytrenowanych modeli

id obrazu	rzeczywista liczba obiektów	3k05				3k08				4k05				4k08			
		liczba detekcji	TP	FP	FN	liczba detekcji	TP	FP	FN	liczba detekcji	TP	FP	FN	liczba detekcji	TP	FP	FN
1	10	7	7	0	4	7	7	0	2	12	11	1	0	9	9	0	2
2	6	6	6	0	0	6	6	0	0	6	6	0	0	6	6	0	0
3	19	21	18	3	3	14	14	0	7	22	19	3	2	16	16	0	5
4	9	8	6	2	1	9	8	1	1	8	8	0	2	8	8	0	2
5	30	26	26	0	4	20	20	0	12	29	27	2	5	26	25	1	7
6	33	23	22	1	13	14	14	0	21	29	26	3	9	24	23	1	12
7	16	11	11	0	5	8	8	0	8	15	12	3	5	8	8	0	8
8	27	25	25	0	4	16	16	0	13	27	26	1	3	20	20	0	9
9	22	26	19	7	2	16	16	0	6	24	22	2	1	21	19	2	4
10	22	21	19	2	2	20	17	3	5	20	19	1	3	19	18	1	4
11	13	12	12	0	1	10	10	0	3	15	13	2	0	12	10	2	3
12	13	12	12	0	1	6	6	0	7	16	13	3	0	12	12	0	2
13	23	23	19	4	2	15	15	0	7	23	20	3	3	20	18	2	4
14	7	7	4	3	0	7	4	3	0	7	7	0	0	7	7	0	0
15	10	9	9	0	1	7	7	0	3	10	9	1	1	9	9	0	1
16	9	9	7	2	1	8	8	0	2	12	9	3	0	10	8	2	1
17	11	10	8	2	2	7	7	0	4	9	9	0	2	8	7	1	3
18	9	10	7	3	1	7	7	0	2	12	9	3	0	8	8	0	2
19	9	10	8	2	0	7	7	0	3	10	9	1	0	9	7	2	1
20	12	12	10	2	0	9	9	0	1	13	11	2	0	9	9	0	2
21	8	8	7	1	0	4	4	0	3	8	7	1	1	4	4	0	3
22	10	7	7	0	2	5	5	0	4	8	7	1	3	6	6	0	3
23	6	6	5	1	0	6	6	0	0	6	6	0	0	6	6	0	0
24	5	5	4	1	0	5	5	0	0	5	5	0	0	5	5	0	0
25	21	20	17	3	3	14	13	1	7	22	20	2	1	16	16	0	4



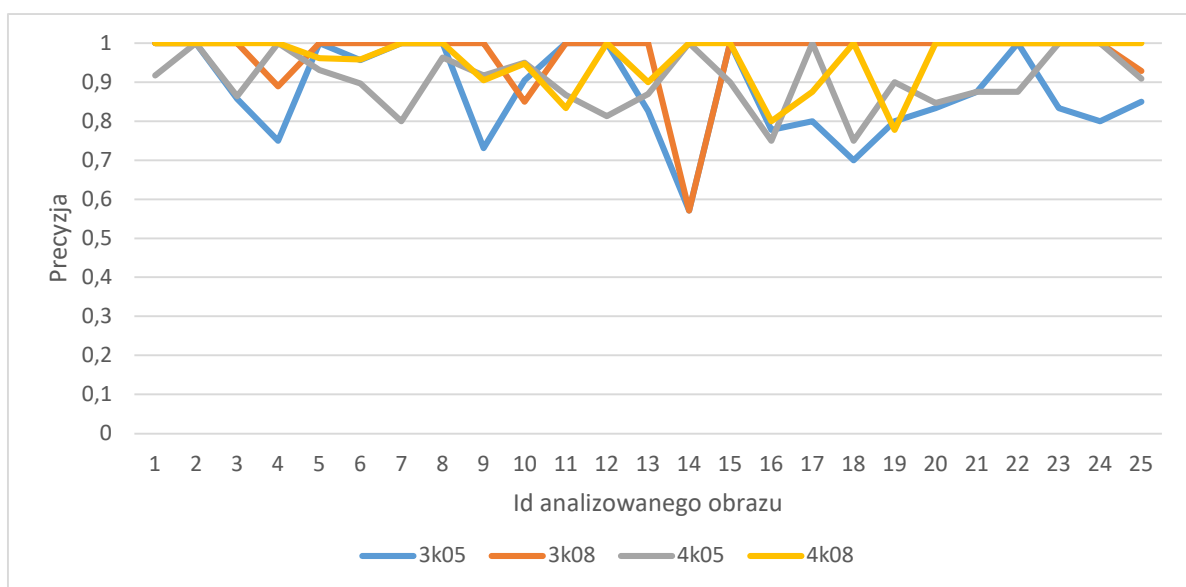
Rys. 21 Przykładowe wyniki detekcji pszczół zwrócone przez model 4k05 na obrazach z większym (a) i mniejszym (b) zagęszczeniem obiektów.

Dla każdego z modeli precyzja detekcji pszczół, której szczegółowe wartości przedstawione zostały na Rys. 23, była dosyć wysoka. Oznacza to, że zwracane obiekty w większości odpowiadają obszarom, na których znajdują się pszczoły. Można przy tym zauważyć następującą zależność – im wyższa jest wartość parametru *IoU* modelu, tym większa jest precyzja wyznaczona dla każdego z obrazów.



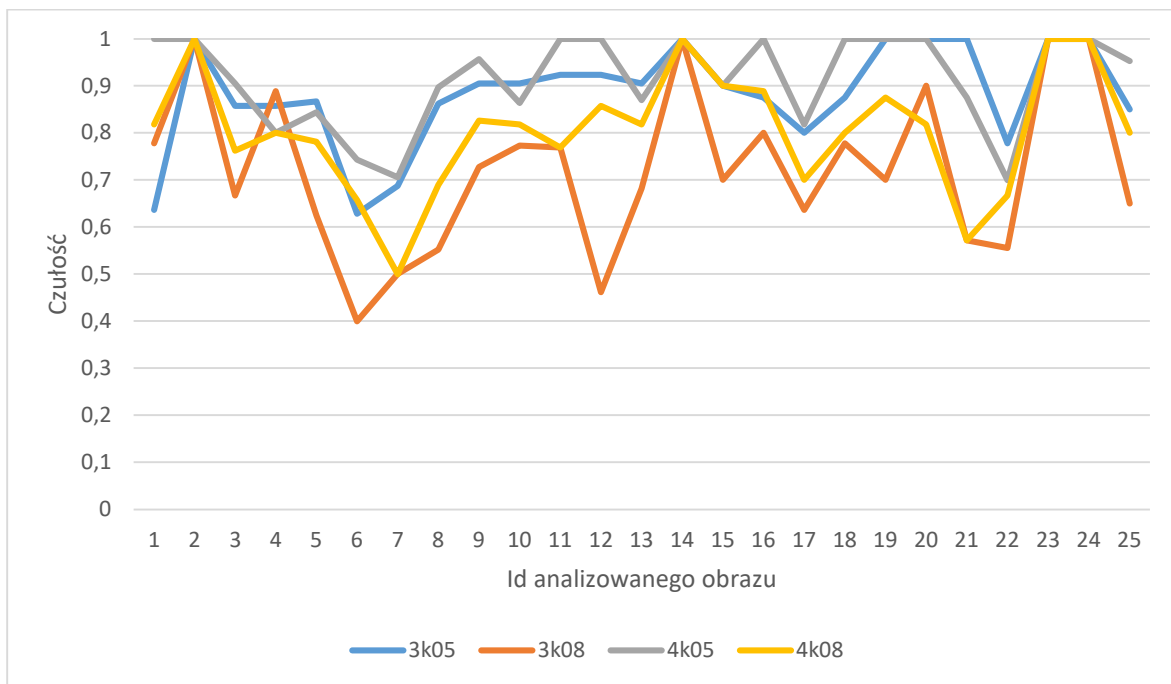
Rys. 22 Miara dokładności modeli wykrywających obiekty pszczoł

Żaden z modeli nie jest jednak nieomylny i każdemu z nich, nawet dla wartości IoU powyżej 0.8, zdarza się określić jako pozytywny obszar, na którym pszczoły nie ma. Najniższe wartości precyzji dotyczą detekcji wyznaczonych przez model 3k05, podobnie jest w przypadku najniższych wartości dokładności. Już na etapie analizy tych dwóch miar można zauważyć, że modele wytrenowane w trakcie 3000 epok radzą sobie gorzej z detekcją pszczoł niż modele, których proces uczenia trwał dłużej.



Rys. 23 Jakość klasyfikacji modeli wyrażona jako metryka precyzji

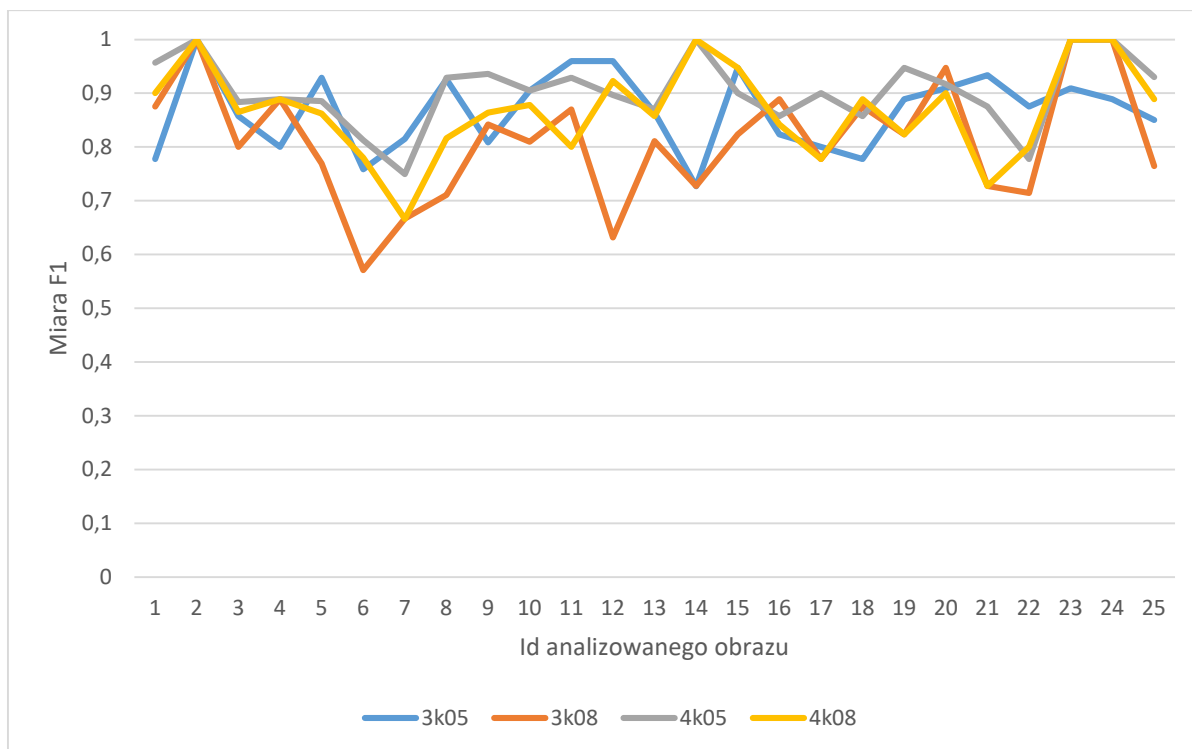
Czułość modeli, przedstawiona na Rys. 24 pokazuje, że lepsze rezultaty uzyskiwane są dla modeli 3k05 i 4k05. Mają one ustawioną niższą wartość parametru IoU , w związku z czym zwracają większą liczbę znalezionych (zazwyczaj poprawnie) obiektów. Wartości czułości dla tych modeli są do siebie zbliżone, chociaż, podobnie jak w przypadku dokładności i precyzji, lepsze wyniki uzyskuje się dla modelu 4k05.



Rys. 24 Czułość modeli detekcji pszczół

Wartości miary F1 dla poszczególnych modeli i obrazów, zilustrowane na Rys. 25, odzwierciedlają wyniki uzyskane dla precyzji i czułości – ponownie lepiej wypadają modele, dla których proces uczenia był dłuższy w porównaniu do modeli trenowanych w procesie 3000 epok.

Analizując powyższe wykresy, ciężko jest jednoznacznie określić, który z modeli zwraca najwyższe wyniki i będzie najlepiej realizował proces detekcji pszczół. Pomocne w wyborze mogą być uśrednione wartości policzonych metryk, przedstawione w Tab. 6. Dane te pokazują, zgodnie z wcześniejszymi założeniami, że modele, które uczone były dłużej, będą zwracały lepsze lub porównywalne wyniki dla analogicznych przedziałów wartości progowej IoU .



Rys. 25 Miara F1 modeli detekcji pszczół

Tab. 6 Uśrednione wartości metryk jakości modeli

Model \ Metryka	3k05	3k08	4k05	4k08
Dokładność	0,77	0,70	0,83	0,78
Precyzja	0,87	0,97	0,90	0,96
Czułość	0,88	0,72	0,91	0,80
Miara F1	0,87	0,81	0,90	0,87

Ważnym aspektem przy doborze odpowiednich parametrów modelu, na podstawie którego wykonywane będą dalsze operacje, jest liczba obiektów znalezionych w procesie detekcji. Do rozwiązania postawionego w niniejszej rozprawie problemu badawczego, tj. wykrywania obecności dręcza pszczelego, istotne jest, aby zminimalizować ryzyko pominięcia chorej pszczoły. W związku z tym, na etapie detekcji wszystkich osobników, dokładność i czułość klasyfikatora wskazują w większym stopniu na odpowiednie modele – modele 4k05 i 4k08. Zdarzało się, że zwracały one większą liczbę obiektów niż faktyczną liczbę pszczół. Taka sytuacja może wskazywać na delikatne *przetrenowanie* (ang. *overfitting*) modeli, natomiast warto mieć na uwadze, że często znalezione i sklasyfikowane jako pszczoły obiekty stanowiły po prostu niepełne sylwetki

owadów. Detekcja takich obiektów nie przeszkodzi na dalszych etapach prowadzonych badań, oraz, z dużym prawdopodobieństwem, nie wpłynie na jakość oceny stanu zdrowia pszczoły. Będzie natomiast korzystne, jeśli okaże się, że na danym fragmencie pszczoły (np. wchodzącej lub wychodzącej z ula), wykryta zostanie obecność dręcza pszczelego. Dla zadanych założeń będzie to model *4k05*, tj. model wytrenowany w trakcie 4000 epok z określonym parametrem $IoU \geq 0.5$.

5.3. Detekcja dręcza pszczelego – klasyfikacja pszczół

Kolejnym – kluczowym – etapem badań było przeprowadzenie analiz mających na celu wykrycie chorych osobników. Klasyfikacja pszczół pod kątem obecności dręcza pszczelego została sprawdzona w oparciu o przedstawiony w poprzednim rozdziale model *4k05*, który, w wyniku przeprowadzonych analiz, został określony jako najbardziej odpowiedni. Do przygotowania modelu detekcji dręcza pszczelego wykorzystany został bazowy zbiór 950 zdjęć pszczół wyodrębnionych w procesie detekcji opisanym w rozdziale 5.1. W skład tego zbioru weszły 64 zdjęcia przedstawiające chore pszczoły i 886 zdjęć przedstawiających pszczoły zdrowe. Zbiór obrazów zawierających osobniki chore został podzielony na dwa podzbiory (uczący – 49 obiektów, i testowy – 15), a następnie każdy z tych zbiorów został odpowiednio zwielokrotniony (tak jak opisano w rozdziale 5.1). Podziału dokonano przed procesami augmentacji, aby mieć pewność, że wśród danych testowych znajdują się obiekty różne w stosunku do zbioru danych treningowych. Następnie dokonano operacji augmentacji wszystkich zdjęć, w wyniku której uzyskano łącznie 6050 obrazów, z których 4950 (~ 80%) posłużyło uczeniu modelu, a 1100 jego testowaniu.

Podobnie jak w etapie dotyczącym detekcji pszczół, w procesie wykrywania dręcza pszczelego wykorzystano model sieci neuronowej SSD MobileNetV2 Lite. W tym przypadku do uzyskania dobrych wyników wystarczył proces uczenia wymagający 2000 epok. Rezultaty wszystkich detekcji, opisane jako macierz pomyłek, zostały przedstawione w Tab. 7. Dla zdecydowanej większości przypadków klasyfikator prawidłowo ocenił, czy na zdjęciu znajduje się pszczoła zdrowa czy chora. Dobry wynik uzyskany został również

dla wartości fałszywie ujemnych, czyli sytuacji, w których dręcz pszczeli nie został zauważony przez model. Miało to miejsce tylko dla 3 obiektów, co stanowi jedynie 0.003% wszystkich sprawdzonych obrazów. Zdecydowanie gorzej, bo aż dla 12% przypadków klasyfikator ocenił zdrowe pszczoły jako chore, co na dalszym etapie działania systemu może skutkować niepotrzebnymi powiadomieniami wysyłanymi do pszczelarza. Zdjęcia pszczół sklasyfikowanych jako chore są każdorazowo wysyłane w wiadomości email do pszczelarza, który ma możliwość weryfikacji czy model właściwie ocenił obecność dręcza pszczelego na ciele pszczoły.

Tab. 7 Macierz pomyłek detekcji dręcza pszczelego

		Wartości rzeczywiste	
		0	1
Wartości predykowane	0	404 (TN)	3 (FN)
	1	146 (FP)	547 (TP)

Wyniki poszczególnych metryk oceniających jakość wykrywania obecności dręcza pszczelego zostały przedstawione w Tab. 8. Uzyskane rezultaty potwierdzają, że jakość klasyfikacji pszczół przygotowanego modelu jest na wysokim poziomie. Dla równomiernie zbilansowanego zbioru dokładność na poziomie 86% jest dobrym wynikiem. Czułość na poziomie 99% potwierdza, że klasyfikator bardzo dobrze identyfikuje chore pszczoły, natomiast nieco niższe wartości precyzji (79%) oraz swoistości (73%) odzwierciedlają pomyłki w określaniu zdrowych pszczół jako chore. Ogólną jakość klasyfikatora określa wynik współczynnika MCC, będący na poziomie 77%.

Tab. 8 Miary oceny jakości detekcji dręcza pszczelego

Metryka	Wartość
Dokładność	0.86
Precyzja	0.79
Czułość	0.99
F1	0.88
Swoistość	0.73
Współczynnik korelacji Matthews'a	0.77

5.4. Działanie algorytmu adaptacyjnego wyznaczania okna zagęszczenia

Celem działania algorytmu adaptacyjnego wyznaczania okna zagęszczenia jest ograniczenie obszaru klatek filmowych w taki sposób, aby analizie podlegał taki fragment obrazu, na którym znajdują się pszczoły. Badania dotyczące działania tego algorytmu miały na celu sprawdzenie, dla których wartości parametrów wejściowych wyznaczone obszary analitycznego okna zagęszczenia są odpowiednie – realnie odzwierciedlają fragment klatki filmowej, na której zagęszczenie pszczół jest największe. Algorytm adaptacyjnego wyznaczania okna zagęszczenia został zweryfikowany na podstawie pięciu filmów, nagranych w różnych warunkach, takich, że na każdym z nich odnotowano inne zagęszczenie oraz inną średnią liczbę pszczół przypadających na klatkę. Z każdego filmu wyodrębniono 300 następujących po sobie klatek, rozmiar początkowy każdej z klatek wynosił 1280×720 pikseli. Dla każdego z nich sprawdzono dwa parametry wysokości siatki $h = 40$ i $h = 80$ i dwie wartości stałej $a = 0.7$ oraz $a = 0.8$, będącej współczynnikiem regulacji progowej t . Wartość współczynnika a ma bezpośredni wpływ na sumę pszczół znalezionych w kolejnych polach siatki, po uzyskaniu której proces aktualizacji rozmiaru okna zagęszczenia zostanie zakończony. Dla każdej z powyższych konfiguracji parametrów wykonano obliczenia dla uśrednionych współrzędnych pszczół (*okno zagęszczenia* wyznaczane za pomocą jednej siatki), oraz dla wartości wszystkich

współrzędnych pszczoł (*okno zagęszczenia* wyznaczane za pomocą czterech siatek). Działanie algorytmu zostało sprawdzone z wykorzystaniem modelu detekcji pszczoł, określonego w rozdziale 5.2 jako model *4k05*. Uzyskane wyniki zostały przedstawione w dwóch tabelach (Tab. 9 oraz Tab. 10) i pogrupowane według wartości parametru wysokości siatki h .

Tab. 9 Wyniki działania algorytmu adaptacyjnego dla rozmiaru siatki $h = 40$

Id filmu	suma znalezionych pszczoł	Śr. liczba pszczoł/klatka	liczba siatek=1, a=0.7	liczba siatek=1, a=0.8	liczba siatek=4, a=0.7	liczba siatek=4, a=0.8
1	353	3	$x_{\min} = 40, x_{\max} = 880$ $y_{\min} = 400, y_{\max} = 640$	$x_{\min} = 0, x_{\max} = 1240$ $y_{\min} = 0, y_{\max} = 680$	$x_{\min} = 0, x_{\max} = 960,$ $y_{\min} = 200, y_{\max} = 680$	$x_{\min} = 0, x_{\max} = 1240,$ $y_{\min} = 0, y_{\max} = 680$
2	504	5	$x_{\min} = 40, x_{\max} = 1200$ $y_{\min} = 40, y_{\max} = 640$	$x_{\min} = 0, x_{\max} = 1240$ $y_{\min} = 40, y_{\max} = 680$	$x_{\min} = 0, x_{\max} = 1240$ $y_{\min} = 0, y_{\max} = 680$	$x_{\min} = 0, x_{\max} = 1240$ $y_{\min} = 0, y_{\max} = 680$
3	2690	26	$x_{\min} = 80, x_{\max} = 880,$ $y_{\min} = 40, y_{\max} = 640$	$x_{\min} = 40, x_{\max} = 1000,$ $y_{\min} = 40, y_{\max} = 680$	$x_{\min} = 40, x_{\max} = 960,$ $y_{\min} = 0, y_{\max} = 680$	$x_{\min} = 0, x_{\max} = 1240,$ $y_{\min} = 0, y_{\max} = 680$
4	2312	23	$x_{\min} = 80, x_{\max} = 1000,$ $y_{\min} = 0, y_{\max} = 680$	$x_{\min} = 0, x_{\max} = 1240$ $y_{\min} = 0, y_{\max} = 680$	$x_{\min} = 0, x_{\max} = 1240$ $y_{\min} = 0, y_{\max} = 680$	$x_{\min} = 0, x_{\max} = 1240$ $y_{\min} = 0, y_{\max} = 680$
5	824	8	$x_{\min} = 0, x_{\max} = 1240$ $y_{\min} = 0, y_{\max} = 680$	$x_{\min} = 0, x_{\max} = 1240$ $y_{\min} = 0, y_{\max} = 680$	$x_{\min} = 0, x_{\max} = 1240$ $y_{\min} = 0, y_{\max} = 680$	$x_{\min} = 0, x_{\max} = 1240$ $y_{\min} = 0, y_{\max} = 680$

Tab. 10 Wyniki działania algorytmu adaptacyjnego dla rozmiaru siatki $h = 80$

Id filmu	suma znalezionych pszczoł	Śr. liczba pszczoł/klatka	liczba siatek=1, a=0.7	liczba siatek=1, a=0.8	liczba siatek=4, a=0.7	liczba siatek=4, a=0.8
1	353	3	$x_{\min} = 80, x_{\max} = 880$ $y_{\min} = 400, y_{\max} = 640$	$x_{\min} = 0, x_{\max} = 1200$ $y_{\min} = 0, y_{\max} = 640$	$x_{\min} = 0, x_{\max} = 1200,$ $y_{\min} = 160, y_{\max} = 640$	$x_{\min} = 0, x_{\max} = 1200$ $y_{\min} = 0, y_{\max} = 640$
2	504	5	$x_{\min} = 0, x_{\max} = 800$ $y_{\min} = 240, y_{\max} = 560$	$x_{\min} = 0, x_{\max} = 1200$ $y_{\min} = 0, y_{\max} = 640$	$x_{\min} = 0, x_{\max} = 1200$ $y_{\min} = 0, y_{\max} = 640$	$x_{\min} = 0, x_{\max} = 1200$ $y_{\min} = 0, y_{\max} = 640$
3	2690	26	$x_{\min} = 80, x_{\max} = 880,$ $y_{\min} = 0, y_{\max} = 640$	$x_{\min} = 80, x_{\max} = 880,$ $y_{\min} = 0, y_{\max} = 640$	$x_{\min} = 0, x_{\max} = 880,$ $y_{\min} = 0, y_{\max} = 640$	$x_{\min} = 0, x_{\max} = 1200$ $y_{\min} = 0, y_{\max} = 640$
4	2312	23	$x_{\min} = 80, x_{\max} = 800,$ $y_{\min} = 0, y_{\max} = 640$	$x_{\min} = 0, x_{\max} = 1200$ $y_{\min} = 0, y_{\max} = 640$	$x_{\min} = 0, x_{\max} = 1200$ $y_{\min} = 0, y_{\max} = 640$	$x_{\min} = 0, x_{\max} = 1200$ $y_{\min} = 0, y_{\max} = 640$
5	824	8	$x_{\min} = 0, x_{\max} = 1200$ $y_{\min} = 0, y_{\max} = 640$	$x_{\min} = 0, x_{\max} = 1200$ $y_{\min} = 0, y_{\max} = 640$	$x_{\min} = 0, x_{\max} = 1200$ $y_{\min} = 0, y_{\max} = 640$	$x_{\min} = 0, x_{\max} = 1200$ $y_{\min} = 0, y_{\max} = 640$

Porównując rezultaty działania algorytmu adaptacyjnego dla różnych parametrów można zauważyć kilka zależności. Dla odpowiedniego doboru parametrów, ważna jest, poza średnią liczbą pszczoł, gęstość ich rozmieszczenia w danej klatce – dla dużej liczby pszczoł ulokowanych dość blisko siebie dokładniej dopasowany obszar okna zagęszczenia uzyskuje się z zastosowaniem jednej siatki o mniejszych rozmiarach pól. Taki układ parametrów powoduje, że pole okna zagęszczenia jest dość duże, natomiast obszary na

których nie ma pszczoł (lub są bardzo niewyraźne) są poprawnie ograniczone przez algorytm, czego przykładem jest wyznaczone okno zagęszczenia dla filmu 3, przedstawione na *Rys. 26*, na którym obszar okna zagęszczenia jest zaznaczony czerwoną ramką i umiejscowiony na bazowym obrazie, będącym pierwszą klatką analizowanego filmu.



Rys. 26 Okno zagęszczenia obliczone dla filmu 3 dla następujących parametrów: $h=40$, $a=0.8$

Wynik zwrócony przez algorytm dla zadanych parametrów uwzględnia wspomniane wcześniej zagęszczenie i małą rotację obiektów. Zdjęcie przedstawia pszczoły „wentylujące” ul – w okolicy wejścia do ula znajduje się dużo obiektów które na kolejnych klatkach będą się przemieszczały o niewielkie odległości. Z dużym prawdopodobieństwem obiekty, które zostały wykryte przez algorytm „wejdą” w obszar wyznaczonego okna zagęszczenia w kolejnych klatkach. Jednocześnie obszar po prawej stronie oryginalnego obrazu został uznany za nieistotny ze względu na nieostre obiekty, które nie zostały znalezione w procesie detekcji pszczoł. Nieco inną sytuację przedstawia film 4, dla którego okno zagęszczenia wyznaczone przez algorytm, przedstawione na *Rys. 27*, jest bardziej dopasowane dla parametru $a=0.7$. Dla tego filmu również odnotowano dużą liczbę pszczoł. Na poszczególnych klatkach zagęszczenie pszczoł jest jednak mniejsze niż na filmie 3. Ponownie obszar po prawej stronie, zawierający niewyraźne obiekty został

określony przez algorytm jako nieistotny. W przypadku ustawienia parametru $a = 0.8$, jak można odczytać z Tab. 9, obszar okna zagęszczenia byłby tylko w niewielkim stopniu mniejszy od obrazu wejściowego, a obszar z niewyraźnymi obiektami pszczoł podlegałyby dalszym analizom, mimo, że prawdopodobieństwo wykrycia na nim obiektu byłoby bardzo niskie. Powyższe obserwacje wskazują, że wartość współczynnika a powinna być zatem ustalana z uwzględnieniem zagęszczenia obiektów – większe zagęszczenie obiektów (liczba pól z dużą liczbą pszczoł) powinno odpowiadać wyższej wartości parametru a .



Rys. 27 Okno zagęszczenia obliczone dla filmu 4 dla następujących parametrów $h=40$, $a=0.7$

Dla niewielkiej liczby pszczoł obliczenie okna zagęszczenia dla każdej kombinacji współrzędnych znalezionych obiektów zwraca większe okno zagęszczenia, dzięki czemu zminimalizowane jest ryzyko pominięcia pszczoły, która np. wlatuje do ula. W sytuacji niewielkiej liczby pszczoł wyznaczenie zbyt małego okna zagęszczenia zmniejsza liczbę klatek, w których pojawi się poruszająca się pszczoła. Powyższy przykład dobrze ilustrują przedstawione na Rys. 28 a) i b) okna zagęszczenia obliczone w wyniku działania algorytmu dla filmu 1, na którym pojawia się niewiele pszczoł na poszczególne klatki, co wskazuje na to, że są one w ruchu. Okno zagęszczenia na rysunku a) jest zbyt małe – jego obszar wynosi 840×240 px, co stanowi jedynie 22% rozmiaru początkowego. Co więcej, okno zagęszczenia uwzględnia jedynie niewielki obszar przed wejściem do ula,

co zmniejsza szansę na detekcję pszczoły, która wleci bezpośrednio do ula (bez zatrzymania się na jego ścianie).



Rys. 28 Okna zagęszczenia wyliczone za pomocą algorytmu adaptacyjnego dla filmu 1 z użyciem jednej (zdjęcie a.) i czterech (zdjęcie b.) siatek

Lepiej wyznaczone, choć nieco większe okno zagęszczenia zostało zaprezentowane na zdjęciu b). Obiekty znajdujące się w jego polu przedstawiają pełne sylwetki pszczoł, a obszar przed wejściem do ula jest na tyle szeroki, że możliwa będzie detekcja pszczoł, które wlatują do ula bezpośrednio, lub zatrzymają się obok wejścia do ula i do niego wejdą. Wyznaczenie okna zagęszczenia w odpowiednim rozmiarze, jest ważne, ponieważ niepożądaną sytuacją byłoby zbyt duże ograniczenie analizowanego obrazu skutkujące znacznym zmniejszeniem liczby wykrywanych obiektów.

Wyniki uzyskane dla filmu 2, zaprezentowane w tabelach Tab. 9 oraz Tab. 10 wraz z zdjęciem pierwszej klatki tego filmu, przedstawionej na *Rys. 29* mogą mylnie sugerować, że dla parametru liczby siatek = 4 zostały wyznaczone zbyt duże obszary okna zagęszczenia.



Rys. 29 Okno zagęszczenia zwrócone przez algorytm adaptacyjny dla filmu 2

Analizując kolejne klatki, widać, że powyższe założenie jest błędne. Na *Rys. 30*, przedstawiającym późniejsze klatki filmu 2 widać, dlaczego okna zagęszczenia zostały wyznaczone poprawnie. Na analizowanych obrazach znajduje się niewielka liczba pszczół, co, zgodnie z przyjętymi założeniami, wskazuje na ich dużą rotację w okolicy wejścia do ula. Potwierdzają to zidentyfikowane obiekty pszczół będących w locie. Warto przy tym zaznaczyć, że między poniższymi klatkami upłynęły ułamki sekund, co dodatkowo udowadnia poprawność założeń zdefiniowanych dla algorytmu adaptacyjnego wyznaczania okna zagęszczenia.



Rys. 30 Późniejsze klatki filmu

Z przedstawionych w tabelach Tab. 9 i Tab. 10 wyników można też wywnioskować, że dobór powyższych parametrów nie ma aż takiego znaczenia w sytuacji, w której pszczoły są w ruchu – na zdjęciach jest ich niewiele (ale nie bardzo mało), i dla większości znalezionych obiektów tor lotu i przestrzeń, po której się poruszają są podobne (czyli, np. kolejne pszczoły będąc w locie są lokalizowane w tych samych polach siatki, po czym lądują i wchodzą do ula również w tych samych lub sąsiednich polach siatki). Przykładem tego może być obszar okna zagęszczenia uzyskany dla filmu 5, zaprezentowany na Rys. 31.



Rys. 31 Okno zagęszczenia wyznaczone dla filmu 5.

Pomimo tego, że algorytm wyznaczył niewiele mniejszy, w stosunku do oryginalnego, obszar zdjęcia, uwzględnione zostało wejście do ula wraz z obszarem przed, dzięki czemu możliwa jest detekcja pszczół w locie, co ilustrują zdjęcia przedstawione na Rys. 32, będącymi jednymi z kolejnych, analizowanych w ramach zadanego filmu, klatek.

Na przykładzie klitek z filmu 5 widać, że dobrana wartość $lp = 10$ (wykorzystana w procesie przetwarzania wstępnego algorytmu adaptacyjnego opisanego przez **Algorytm 2**) określająca liczbę pszczoł na zdjęciu, w odniesieniu której należy wykonywać obliczenia okna zagęszczenia w oparciu o 4 siatki, została dobrana prawidłowo.



Rys. 32 Późniejsze klatki analizowanego filmu 5

Ostatnim z analizowanych parametrów był rozmiar siatki h . Dla większości analizowanych danych zastosowanie parametru $h=40$ zwracało obszary okna zagęszczenia lepiej odwzorowujące rzeczywiste pozycje pszczoł na obrazach.

Podsumowując, analizując uzyskane wyniki można wywnioskować, że parametry algorytmu adaptacyjnego wyznaczania okna zagęszczenia takie jak współczynnik a , rozmiar siatki h , oraz liczba siatek mają wpływ na rozmiar wyznaczonego okna zagęszczenia. Na potrzeby analizy wydajności czasowej działania algorytmu adaptacyjnego zostały przyjęte następujące wartości – rozmiar siatki $h = 40$ oraz wartość współczynnika regulacji $a = 0.8$. Takie wartości odwzorowują większą liczbę operacji konieczną do wyznaczenia okna zagęszczenia. Wpływ doboru wartości parametru liczby analizowanych klitek m oraz liczby siatek na wydajność czasową algorytmu adaptacyjnego został poddany analizie w rozdziale 5.5.1.

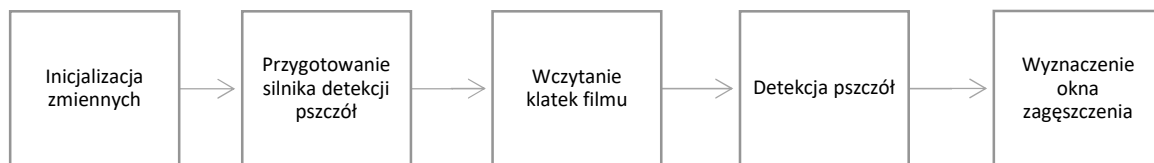
5.5. Wydajność czasowa działania całego systemu wykrywającego sytuacje zagrażające życiu pszczoł

Dla każdego z etapów przetwarzania brzegowego na urządzeniu IoT zostały przeprowadzone pomiary czasu działania, z uwzględnieniem każdego z pięciu filmów wykorzystanych w opisanych we wcześniejszych rozdziałach eksperymentach

badawczych. Czasy zostały zmierzone na urządzeniu docelowym, tj. na komputerze NVidia Jetson Nano 4GB.

5.5.1. Wydajność algorytmu wyznaczającego okno zagęszczenia

Dla każdego z filmów wykonano pomiary czasu działania algorytmu adaptacyjnego wyznaczania okna zagęszczenia dla przetwarzania wartości 100, 200, 300 klatek, z uwzględnieniem kolejnych kroków działania algorytmu, przedstawionych na Rys. 33.

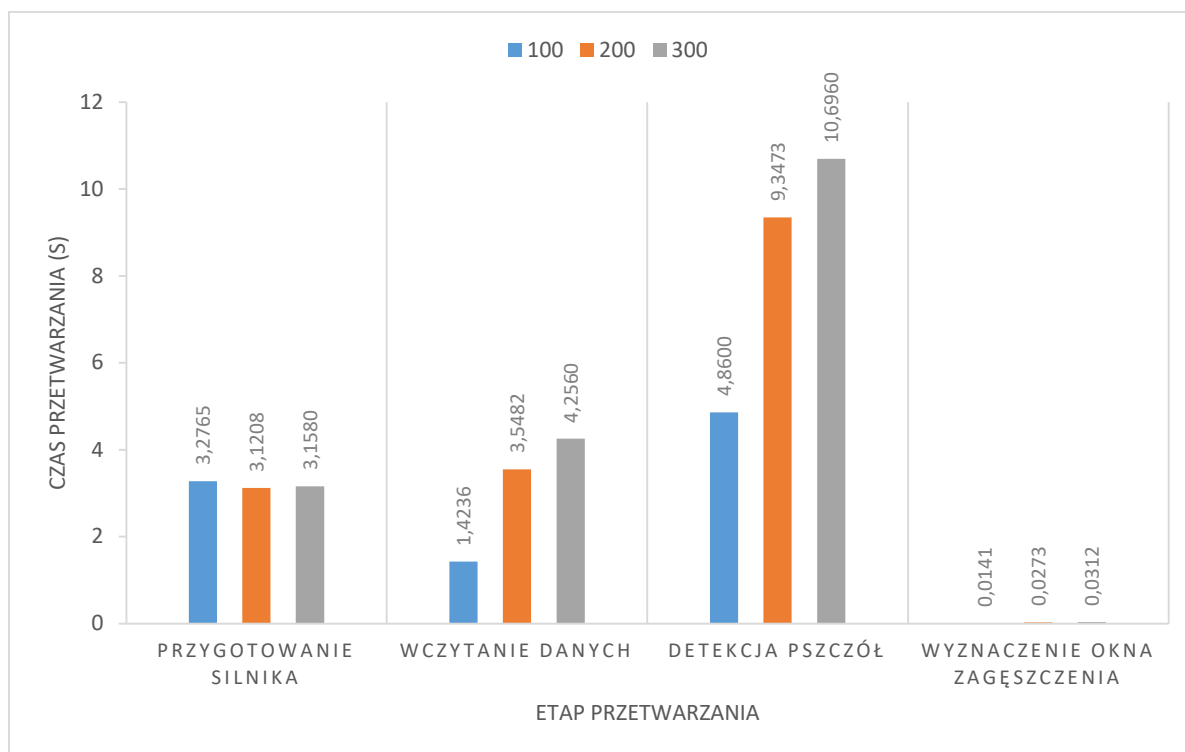


Rys. 33 Kolejne etapy działania algorytmu adaptacyjnego wyznaczania okna zagęszczenia

Analiza liczby klatek większej niż 300 nie przynosiła zmian w obszarach wyznaczonego okna zagęszczenia, zwiększała natomiast znacznie czas trwania całego algorytmu. Sytuacja ta wynika z faktu, że najwięcej czasu potrzebne jest do wykonania detekcji pszczoł na każdej z przetwarzanych klatek.

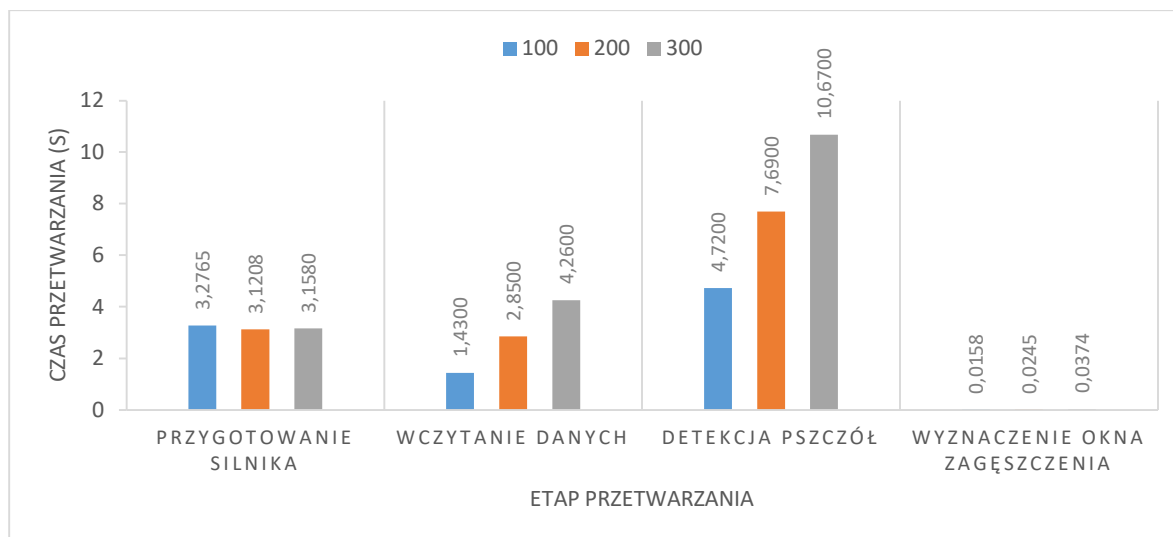
Dla każdego z analizowanych filmów czas trwania inicjalizacji zmiennych wynosił między 0.3 a 0.4 ms – jest to na tyle mała wartość, że nie została uwzględniona na Rys. 34, przedstawiającym średnie czasy wykonywania poszczególnych etapów algorytmu wyznaczającego okno zagęszczenia dla 100, 200 i 300 klatek filmu. Jak można odczytać z tego wykresu, czas wczytywania, a następnie przetwarzania kolejnych klatek filmu jest ściśle związane z liczbą przetwarzanych klatek, a czas potrzebny na wykonanie tego kroku rośnie proporcjonalnie do wzrostu liczby wczytywanych obrazów. Czas przygotowania silnika modelu detekcji pszczoł oscyluje w okolicy wartości 3,1 sekundy i takie wartości uzyskiwano również na innych etapach badań eksperymentalnych dotyczących pomiarów czasów działania przeprowadzonych na potrzeby niniejszej rozprawy, w których wykorzystywano wnioskowanie z wykorzystaniem modeli uczenia maszynowego (niezależnie od tego czy był to model detekcji pszczoł czy dręcza pszczelego).

Z wykresu przedstawionego na Rys. 34 można również zauważyć, że średni czas działania samego algorytmu wyznaczania okna zagęszczenia jest bardzo mały – wynosi od 14,1 do 31,2 ms, i rośnie wraz z liczbą przetwarzanych klatek. Całkowite, uśrednione dla pięciu analizowanych filmów, czasy działania algorytmu adaptacyjnego wyznaczania okna zagęszczenia wynoszą odpowiednio 9,75 sekund, 13,65 sekund oraz 18,08 sekund dla 100, 200 i 300 przetwarzanych klatek.



Rys. 34 Średnie czasy przetwarzania poszczególnych etapów algorytmu wyznaczającego okno zagęszczenia dla 1 siatki

Podobne zależności można odczytać z wykresu przedstawionego na Rys. 35, prezentującego średnie czasy przetwarzania poszczególnych etapów działania algorytmu wyznaczającego okna zagęszczenia. Czasy wyznaczania okna zagęszczenia minimalnie wzrosły dla parametru liczby siatek równej 4, jednak nadal pozostają niewielkie w stosunku do czasów działania wszystkich etapów algorytmu. Całkowite, uśrednione dla pięciu analizowanych filmów, czasy działania algorytmu adaptacyjnego umożliwiającego wyznaczania okna zagęszczenia wyniosły odpowiednio: 9,80 s, 13,67 s oraz 18,01 s dla 100, 200 i 300 przetwarzanych klatek.



Rys. 35 Średnie czasy przetwarzania poszczególnych etapów algorytmu wyznaczającego okno zagęszczenia dla 4 siatek

5.5.2. Wydajność czasowa algorytmu detekcji pszczoł oraz dręcza pszczelego

Detekcja pszczoł jest wykonywana za pomocą sieci neuronowej na procesorze GPU urządzenia Jetson Nano. Sam ten fakt pozwala oczekiwać, że czasy przetwarzania pojedynczej klatki będą niewielkie. Niemniej, zbyt długi czas analizy poszczególnych klatek filmowych może skutkować koniecznością cyklicznego pomijania niektórych z nich, aby móc realizować proces wykrywania obecności dręcza pszczelego w czasie rzeczywistym.

Zgodnie z dokumentacją środowiska *TensorRT* [114], w którym uruchamiane są modele wnioskowania, mierząc czasy wnioskowania należy wziąć pod uwagę fakt, że procesor „musi się rozgrzać” (ang. *warmup*), w związku z czym przetwarzanie pierwszej / kilku pierwszych klatek (w zależności od modelu) może zająć nawet około 3 sekund, podczas gdy analiza kolejnych zajmuje już tylko ułamki sekund. W związku z tym dane opisujące średnie czasy detekcji pszczoł, obliczone dla 600 klatkowego filmu (o długości 60 s) zostały policzone z pominięciem czasu przetwarzania pierwszej klatki (wynoszącego 3,03 s) i przedstawione w Tab. 11. Największy odnotowany czas wynosił 0,15 s i była to wartość uzyskana dla pierwszej analizowanej klatki. W związku z tym, nie należy oczekiwać, że takie wartości będą pojawiały się często.

Uśrednione wartości dla czasów przetworzenia jednego wyodrębnionego zdjęcia z pszczołą zostały natomiast przedstawione w Tab. 12. Wartości przedstawione w tej tabeli zostały wyliczone dla ponad 15 tysięcy zarejestrowanych obiektów pszczoł. W porównaniu do czasu detekcji pszczoł na obrazie, można zauważyć, że średni czas potrzebny do klasyfikacji pszczoły jest mniejszy. Ponownie, pierwsza klatka była najdłużej przetwarzana.

Tab. 11 Czasy detekcji pszczoł na 1 klatce filmu

Wartość średnia (s)	0,0124
Odchylenie standardowe (s)	0,0072
Największy odnotowany czas (s)	0,1570
Najmniejszy odnotowany czas (s)	0,0105

Tab. 12 Czasy klasyfikacji jednego obrazu z obiektem pszczoły pod kątem obecności dręcza pszczelego

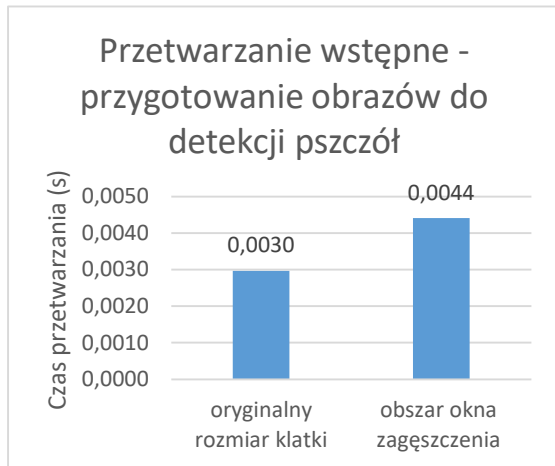
Wartość średnia (s)	0,0109
Odchylenie standardowe (s)	0,0008
Największy odnotowany czas (s)	0,0627
Najmniejszy odnotowany czas (s)	0,0099

Całkowite czasy realizacji poszczególnych etapów działania całego procesu przetwarzania danych na urządzeniu brzegowym dla filmów z różną liczbą pszczoł zostały przedstawione na Rys. 36 – 40.

Rys. 36 przedstawia czasy przetwarzania poszczególnych etapów działania urządzenia brzegowego dla filmu 1. Całkowity czas potrzebny do przetworzenia filmu składającego się z 600 klatek, trwającego 60 sekund wynosił 66,085 s dla oryginalnego obrazu oraz 65,377 s dla obrazu ograniczonego poprzez wyznaczone okno zagęszczenia o współrzędnych [(0,680), (0,1240)]. Na analizowanym filmie średnia liczba pszczoł wynosiła ok. 4 na klatkę, były one w ruchu, i, jak widać po wielkości okna zagęszczenia, nie były one bardzo blisko siebie. Zastosowanie okna zagęszczenia pozwoliło na przyspieszenie czasu przetwarzania całego filmu o niecałą sekundę. Jest to korzyść

niewielka, natomiast biorąc pod uwagę fakt, że procesy detekcji pszczoł i ich klasyfikacji mogą zostać zrównoleglone (tak, jak zostało to zrealizowane we wstępnych badaniach wykonanych na urządzeniu NVidia Jetson Nano w wersji 2 GB [110]), dla małej liczby pszczoł na zdjęciu nie będzie konieczne pomijanie klatek.

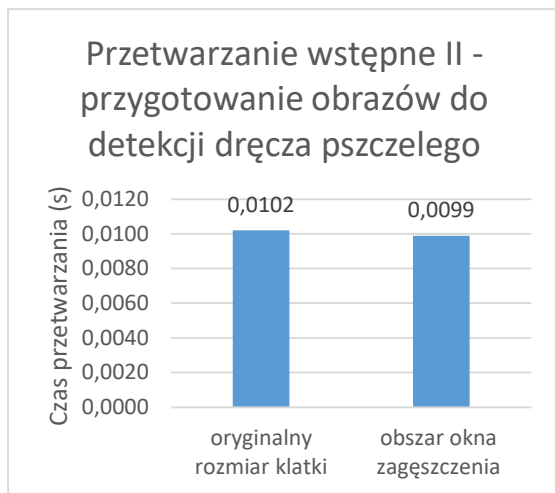
a)



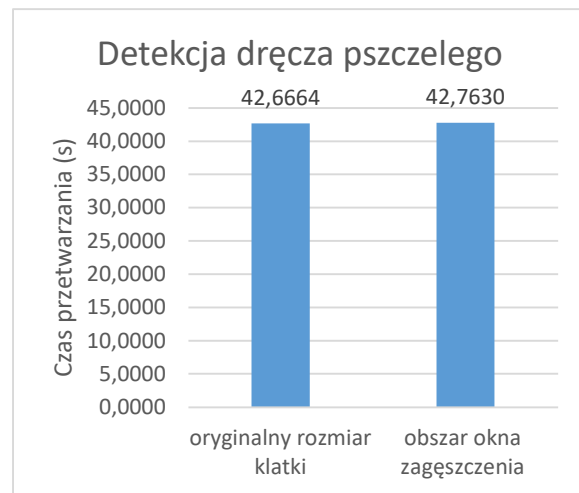
b)



c)



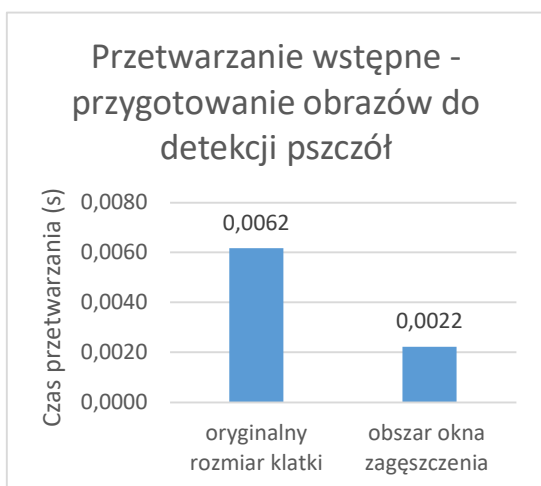
d)



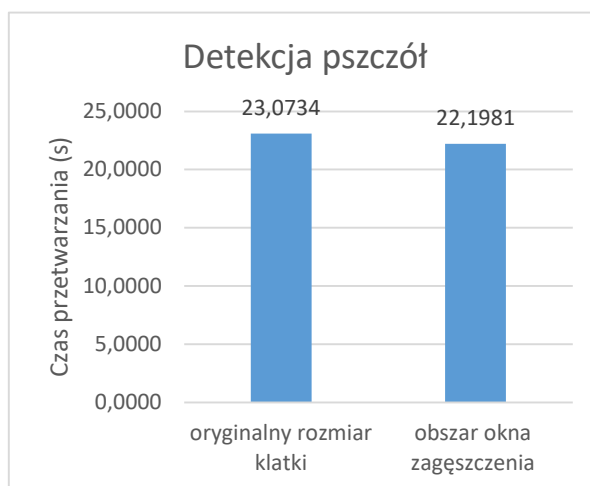
Rys. 36 Czasy przetwarzania filmu 1

Na Rys. 37 przedstawione zostały czasy przetwarzania dla filmu 2, na którym średnia liczba pszczoł na analizowanej klatce wynosiła 5. Całkowity czas przetwarzania filmu wyniósł 73,669 s dla oryginalnego obrazu i 72,482 s dla wyznaczonego okna zagęszczenia. Ponownie, dla okna zagęszczania, uzyskano o 1,2 s lepszy czas, natomiast zrównoleglenie operacji detekcji i klasyfikacji pszczoł umożliwi przetwarzanie filmu bez konieczności pomijania klatek.

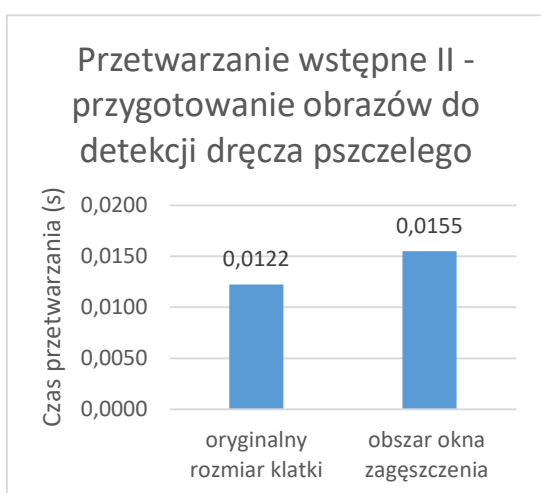
a)



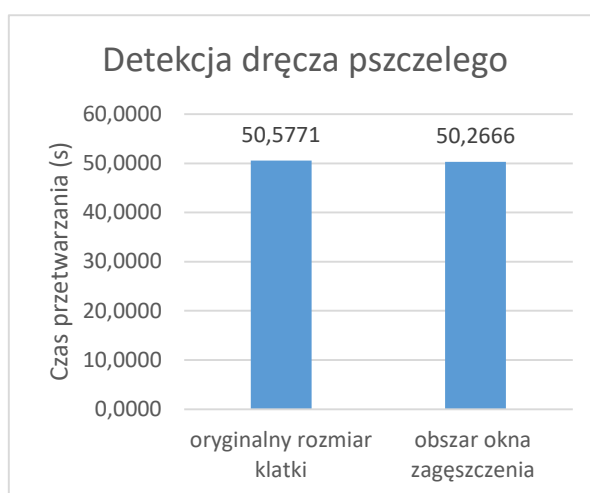
b)



c)



d)

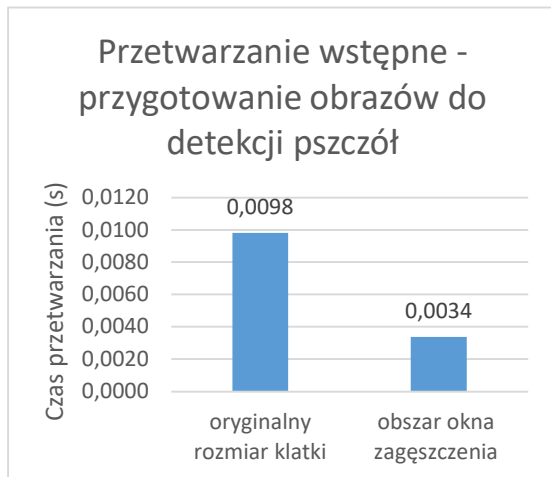


Rys. 37 Czasy przetwarzania filmu 2

Większe różnice w całkowitym czasie przetwarzania zostały odnotowane dla filmu 3. Na tym filmie odnotowano dużą średnią liczbę pszczół przypadających na 1 klatkę (powyżej 20). Całkowity czas przetwarzania filmu, po zastosowaniu okna zagęszczenia zmniejszył się z 279,282 s o blisko 33 sekundy do 246,33 s. Duży całkowity czas przetwarzania wynika z dużej liczby wyodrębnionych pszczół, co jest widoczne na wykresie zilustrowanym na Rys. 38. W tym przypadku zapewnienie przetwarzania w czasie rzeczywistym będzie możliwe przy odpowiednim ustawieniu liczby pomijanych klatek. Dla tego przypadku przetwarzanie co 4 klatki filmu zredukuje czas potrzebny do sklasyfikowania pszczół do ok. 53 sekund (przy uwzględnieniu średniego czasu klasyfikacji 1 pszczoły, przedstawionego w Tab. 12, założeniu zrównoleglenia procesów detekcji pszczół i ich klasyfikacji). Liczba pszczół podlegających klasyfikacji również

ulegnie zdecydowanemu zmniejszeniu natomiast należy mieć na uwadze, że sytuacja, w której pszczoły są blisko siebie, a ich liczba jest duża wskazuje na moment wentylowania ula, co z kolei powoduje niewielkie zmiany położenia pszczoł w kolejnych klatkach. Pominięcie kilku klatek pod rząd nie spowoduje zatem wzrostu prawdopodobieństwa pominięcia pszczoły w procesie detekcji i klasyfikacji.

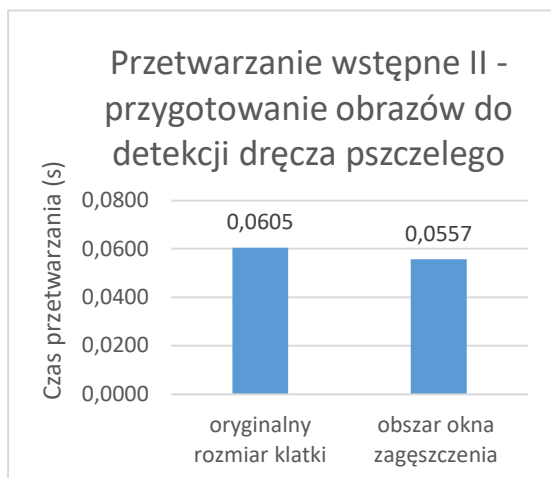
a)



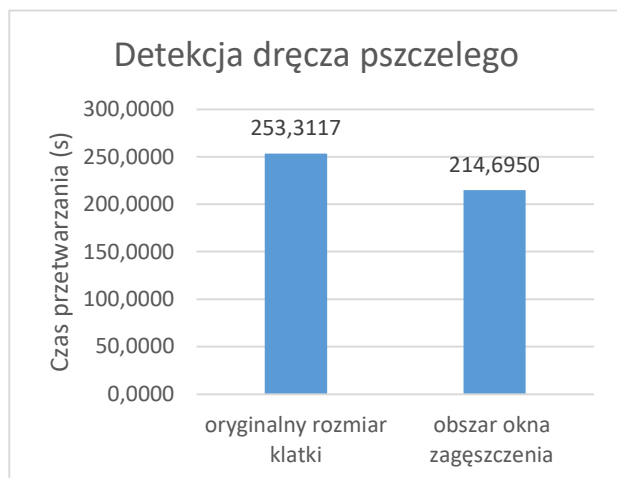
b)



c)



d)

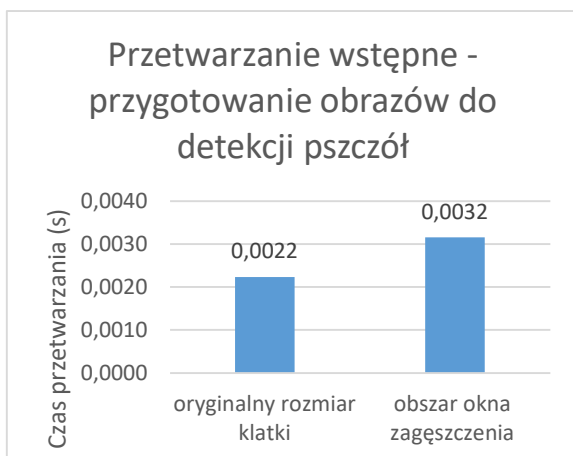


Rys. 38 Czasy przetwarzania dla filmu 3

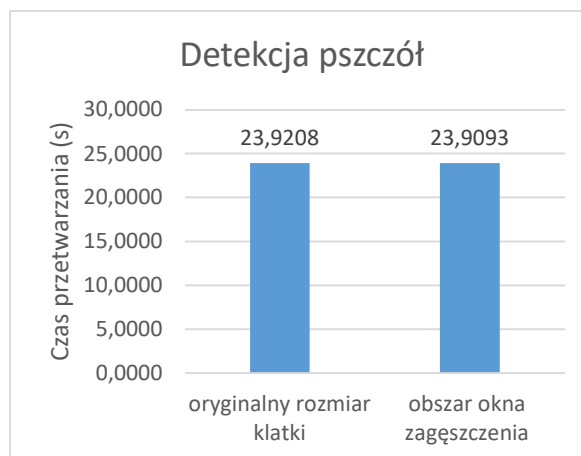
Zastosowanie obszaru okna zagęszczenia przy przetwarzaniu filmu 4, na którym średnia liczba pszczoł przypadających na klatkę wynosiła 23, pozwoliło na redukcję całkowitego czasu przetwarzania filmu ze 185,67 sekund do 133,52 czyli o 1/3 wartości początkowej. Zostało to zilustrowane na Rys. 39. Dla tego filmu przetwarzanie co 3 klatki filmu (przy uwzględnieniu średniego czasu klasyfikacji jednej pszczoły, przedstawionego w Tab. 12,

oraz założeniu zrównoleglenia procesów detekcji i klasyfikacji) pozwoli przetworzenie film w czasie poniżej 60 sekund.

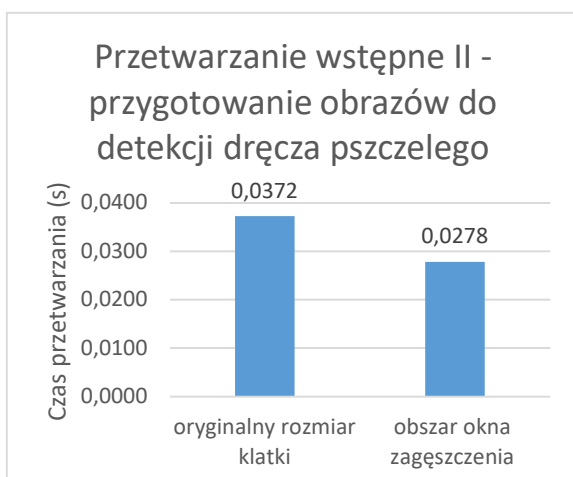
a)



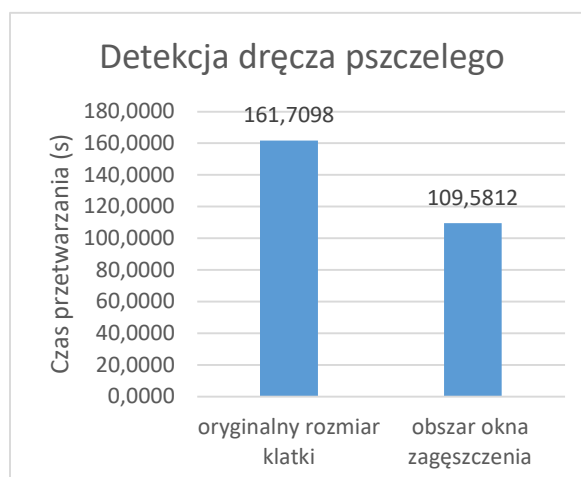
b)



c)



d)



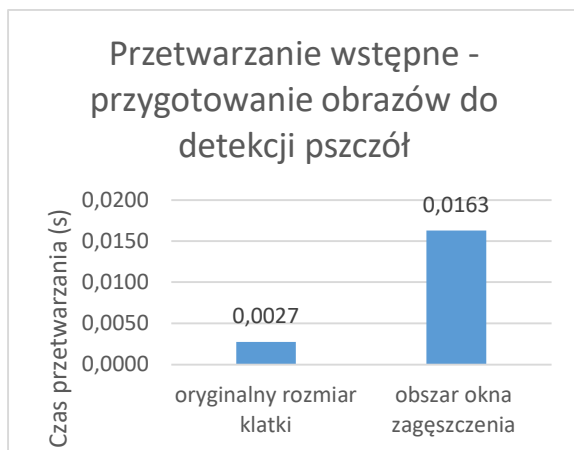
Rys. 39 Czasy przetwarzania dla filmu 4

Dla ostatniego analizowanego filmu 5 (wyniki przedstawiono na Rys. 40, dla którego średnia liczba pszczół przypadająca na jedną klatkę wynosiła ok. 8, zastosowanie okna zagęszczenia zmniejszyło czas klasyfikacji wszystkich wyodrębnionych obiektów pszczół o około 1 sekundę. Zrównoleglenie operacji detekcji pszczół i warrozy dla tego filmu nie zapewni przetworzenia filmu w czasie rzeczywistym, dlatego konieczne będzie pomijanie niewielkiej liczby klatek. Już dla pomijania co 10 klatki i zrównoleglenia wykonywanych operacji możliwe będzie przetworzenie filmu w czasie poniżej minuty.

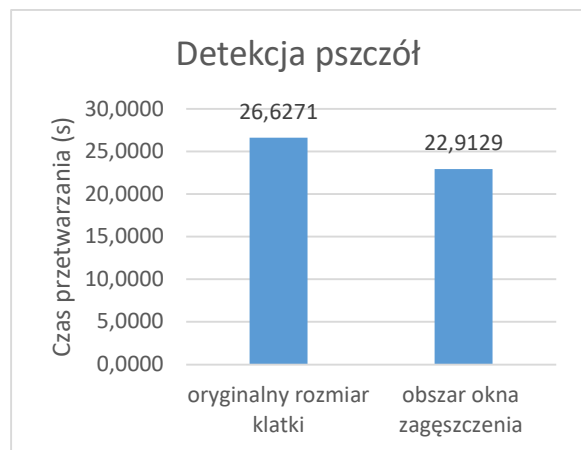
Podsumowując, wyniki badań przeprowadzonych na brzegowym urządzeniu IoT pokazują, że zastosowanie algorytmu adaptacyjnego wyznaczania okna zagęszczenia

pozwala (czasami znacząco) skrócić czas analizy obrazu, co przekłada się na możliwość analizy większej liczby klatek z każdej sekundy filmu i detekcję dręcza pszczelego w czasie rzeczywistym. W niektórych przypadkach niezbędne jest cykliczne pomijanie pewnej liczby klatek analizowanego filmu. Jednak im mniejsza liczba klatek zostanie pominięta, tym mniejsze prawdopodobieństwo utraty ujęć pszczół z widocznym pasożytem dręcza na ciele pszczoły.

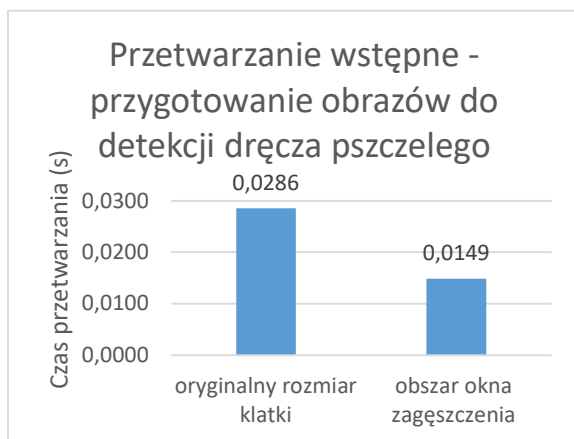
a)



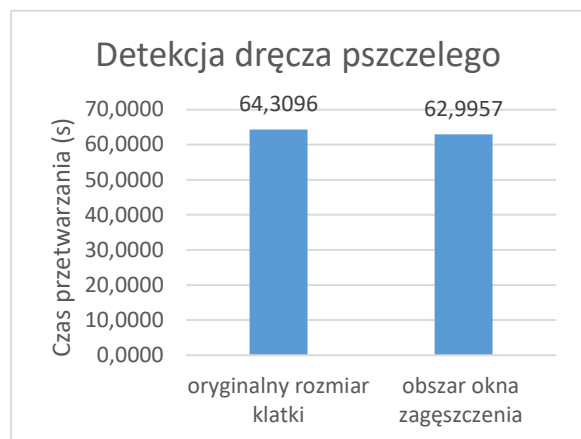
b)



c)



d)



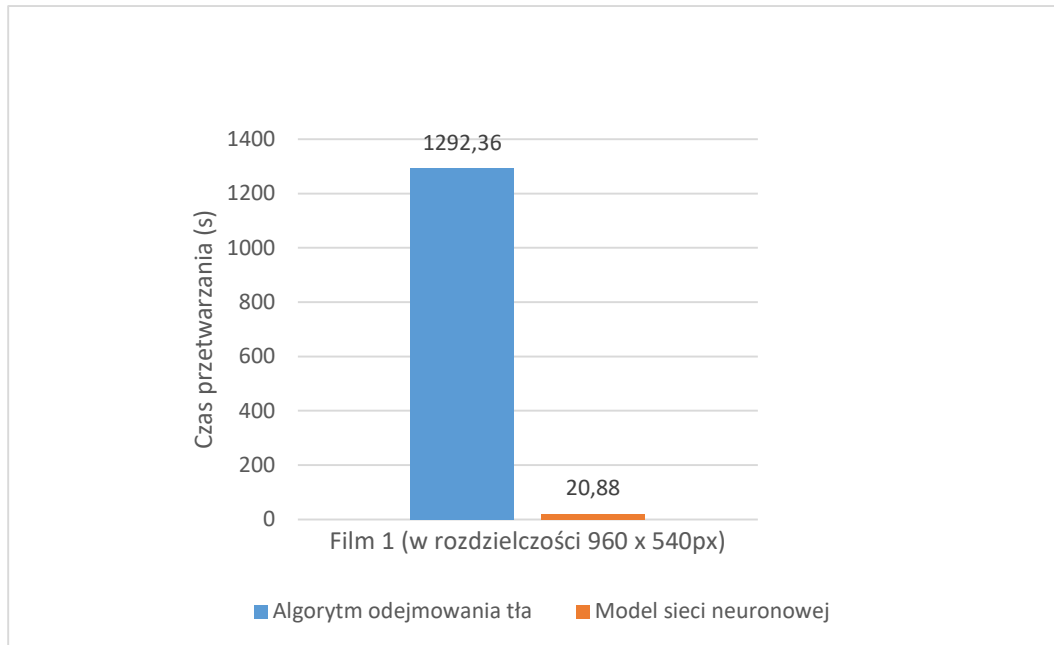
Rys. 40 Czasy przetwarzania filmu 5

5.6. Uzyskane wyniki na tle poprzednio zrealizowanych prac i badań

Przedstawione w poprzednich podrozdziałach eksperymenty badawcze i uzyskane rezultaty stanowią kontynuację badań mających na celu przygotowanie systemu umożliwiającego wykrywanie pszczoł z obecnym na ciele pasożytem – dręczem pszczelim. W ramach wcześniejszych prac, opisanych w artykułach [109] oraz [110] przygotowano inne rozwiązania. W pierwszym z nich [109] system został przygotowany z zastosowaniem komputera RaspberryPi. Aby umożliwić płynne przetwarzanie obrazów i zapewnić działanie systemu w czasie rzeczywistym wykorzystany został akcelerator Google Coral. Zagadnienia związane z detekcją pszczoł, a następnie dręcza pszczelego zostały zrealizowane z wykorzystaniem konwolucyjnej sieci neuronowej wybranej i zoptymalizowanej za pomocą usługi Google AutoML Vision. W drugim podejściu [110] jako jednostką obliczeniową wykorzystano minikomputer NVidia Jetson Nano, jednak w słabszej, posiadającej 2GB pamięci operacyjnej, wersji. Wśród algorytmów detekcji obiektów pszczoł sprawdzone zostały algorytmy odejmowania tła oraz model sieci neuronowej o architekturze zbliżonej do architektury modelu *DetectNet*, natomiast do wykrywania dręcza pszczelego wykorzystany został model sieci neuronowej oparty o architekturę GoogleLeNet. Porównanie czasów przetwarzania klatek służącemu detekcji pszczoł na obrazach z wykorzystaniem algorytmów odejmowania tła oraz sieci neuronowej, zaprezentowane na pokazuje, że adaptacyjne algorytmy odejmowania tła realizują wyznaczone zagadnienie w czasie dużo dłuższym niż model sieci neuronowej, przy takim samym parametrze określającym liczbę pomijanych klatek.

Uzyskane w wyniku obu podejść wyniki detekcji dręcza pszczelego, zestawione z rezultatami badań przeprowadzonych w ramach realizacji niniejszej rozprawy zostały przedstawione w Tab. 13. Uzyskane wyniki dotyczące detekcji dręcza pszczelego pokazują, że aktualne rozwiązanie radzi sobie nieco lepiej niż poprzednie próby. Miara czułości, określająca poziom, z jakim model prawidłowo klasyfikuje chore pszczoły osiąga najwyższy wynik dla algorytmu SSD. Dokładność klasyfikatora jest najwyższa dla rozwiązania działającego na komputerze RaspberryPi, natomiast różnica w odniesieniu do aktualnie proponowanego rozwiązania wynosi tylko 3 punkty procentowe. Wartość precyzji jest najwyższa dla rozwiązania działającego na komputerze NVidia Jetson Nano

2GB, i wskazuje na to, że zastosowany w tym rozwiązaniu klasyfikator bardzo dobrze poprawnie identyfikuje zdrowe pszczoły.



Rys. 41 Wykres czasu przetwarzania strumienia przy 20 pomijanych klatkach dla dwóch różnych algorytmów [110]

Tab. 13 Zestawienie wyników badań przeprowadzonych na różnych systemach

	Metryka	Rozwiązanie 1 [109]	Rozwiązanie 2 [110]	Aktualne rozwiązanie
Detekcja dręcza pszczelego	Dokładność	89%	80%	86%
	Precyzja	70%	89%	79%
	Czułość	94%	68%	99%

Poprawie uległy także wartości czasów przetwarzania, zarówno dla procesu detekcji pszczoł, jak i dręcza pszczelego. W tabeli zaprezentowane zostały uśrednione wartości czasów przetwarzania dla wczesnych algorytmów detekcji pszczoł i warrozy oraz aktualnego rozwiązania.

Tab. 14 Porównanie średnich czasów detekcji pszczół i detekcji dręcza pszczelego na obrazie

Średni czas klasyfikacji jednego obrazu (ms)	Rozwiązanie 1 [109]	Rozwiązanie 2 [110]	Aktualne rozwiązanie
Detekcja pszczół	778	39	12
Detekcja dręcza pszczelego	275	18	10

Na wczesnych etapach badań przeprowadzonych w oparciu o komputer Jetson Nano 2GB, do zapewnienia działania systemu w czasie rzeczywistym, konieczne było pomijanie klatek. Aktualne rozwiązanie wymaga pomijania klatek wyłącznie w sytuacji, w której pszczoły wentylują ul – jest ich bardzo dużo, są blisko siebie i nie zmieniają gwałtownie swojego położenia. Sytuacja ta zostanie zauważona w procesie obliczania okna zagęszczenia i wtedy ustalona zostanie liczba pomijanych klatek. Takie podejście umożliwia dopasowywanie sposobu działania systemu do aktualnego zachowania pszczół i dzięki temu liczba przetwarzanych obrazów jest odpowiednio duża.

Porównując uzyskane wyniki do rozwiązań przedstawionych w literaturze, przedstawione w rozprawie rozwiązanie uzyskuje lepsze wyniki dla czułości (99%) niż opisane w [104] rozwiązanie umożliwiające detekcję dręcza w ulu na podstawie analizy gazów, które uzyskało czułość na poziomie 75%. Elizondo et al. [101] zaprezentowali system umożliwiający wykrycie warrozy w komórkach pszczół, działający z dokładnością 90%. Opisane w rozprawie urządzenie IoT umożliwia detekcję dręcza pszczelego z niewiele gorszą dokładnością (87%), za to bezpośrednio na ciele pszczoły.

W systemach przedstawionych w [92] oraz [99] nie zostały zaprezentowane moduły umożliwiające detekcję dręcza pszczelego, zawarto jedynie informację o tym, że jest to jeden z elementów dalszych prac nad systemami.

Uzyskane wyniki, przedstawione w Tab. 13. udowadniają tezę w kontekście możliwości wykrywania dręcza pszczelego na ciele pszczoły, którego obecność stanowi niewątpliwie sytuację bezpośredniego zagrożenia życia pojedynczej pszczoły, jak również całej pszczelej rodziny. Wyniki przedstawione w Tab. 14, wskazują natomiast, że możliwa

jest realizacja tego procesu w czasie rzeczywistym, tym samym potwierdzając tezę w kontekście oczekiwanego czasu realizacji powyższego procesu.

6. Podsumowanie

W niniejszej rozprawie został poruszony problem wykrywania sytuacji, które mogą stanowić zagrożenie dla zdrowia i życia pszczół. Jednym z przykładów takich sytuacji jest niewątpliwie zarażenie rodziny pszczelej warrozą – chorobą powodowaną przez dręcza pszczelego (ang. *Varroa Destructor*). Wykorzystanie nowoczesnych technologii, które – chociaż stosowane w innych gałęziach nauki i przemysłu – nie znajdują szerokiego zastosowania w pszczelarstwie, pozwala na szybkie i skuteczne wykrywanie obecności dręcza pszczelego. Stanowi to usprawnienie pracy pszczelarzy i może przyczynić się do zmniejszenia strat spowodowanych przez warrozę.

6.1. Podsumowanie zrealizowanych prac i uzyskanych wyników

W wyniku realizacji niniejszej rozprawy został zaprojektowany system umożliwiający wykrywanie obecności dręcza pszczelego. W ramach zrealizowanych prac dokonano implementacji rozwiązania działającego jako urządzenie Internetu rzeczy, które za pomocą kamery gromadzi dane, następnie, wykorzystując przetwarzanie brzegowe dokonuje detekcji obiektów pszczół, a następnie klasyfikuje je ze względu na obecność dręcza pszczelego lub jego brak na ciele pszczoły. Jeśli pszczoła zostanie określona jako chora, uruchamiana jest część systemu odpowiedzialna za przesłanie odpowiedniego powiadomienia do pszczelarza, a zdjęcia chorej pszczoły do chmury, w celu zbudowania bazy danych zdjęć przedstawiających rzeczywiście chore osobniki. W procesach detekcji pszczół i dręcza pszczelego wykorzystano sieci neuronowe dedykowane dla rozwiązań Internetu Rzeczy i problematyki detekcji obiektów na obrazach. Aby zapewnić lepszą pracę urządzenia dokonano implementacji algorytmu adaptacyjnego wyznaczenia okna zagęszczenia dla analizowanych obrazów, które będzie określało obszary, na których znajdują się pszczoły oraz eliminowało te fragmenty obrazu, w których w zadanym czasie nie pojawiły się żadne osobniki pszczół. Działanie modeli wykonujących detekcje oraz algorytmu adaptacyjnego wyznaczenia okna zagęszczenia zostało zweryfikowane na

zaprojektowanym urządzeniu. W wyniku przeprowadzonych badań uzyskano wyniki, które wskazują na to, że jest możliwe wykrycie zarażenia warrozą w rodzinie pszczelej, za pomocą urządzenia Internetu Rzeczy wyposażonego w kamerę, które działa w czasie rzeczywistym i wykonuje operacje przetwarzania brzegowego. Tym samym teza badawcza, zdefiniowana w rozdziale 1.2 niniejszej rozprawy została potwierdzona. Ponadto proponowany system spełnia wszystkie postawione wymagania, tj. działa bez szkody dla rodzin pszczelich i nie ingeruje w ich codzienne aktywności, umożliwia pozyskiwanie i przetwarzanie obrazów przy różnych poziomach oświetlenia ula, działa w czasie rzeczywistym, rozdzielczość z jaką kamera pozyskuje obrazy, została wybrana w oparciu o uzyskane we wcześniejszych badaniach [109], [110] wyniki, co pozwala stwierdzić, że stanowi kompromis pomiędzy jakością pozyskiwanych danych a czasem ich przetwarzania. Wykorzystanie algorytmów i bibliotek dedykowanych dla wykonywania obliczeń z wykorzystaniem procesorów graficznych zapewnia optymalne wykorzystanie zasobów sprzętowych. Porównując uzyskane wyniki z rezultatami otrzymanymi w przeprowadzonych wcześniej badaniach [109], [110] można stwierdzić, że proponowany system lepiej niż poprzednie realizuje zagadnienia związane z wykrywaniem dręcza pszczelego.

6.2. Autorski wkład badawczy

Wśród rozwiązań dedykowanych pszczelarzom, które zostały dotychczas opisane w literaturze brakuje systemów i urządzeń czerpiących z najnowszych technologii. W niniejszej rozprawie zaprezentowane zostało urządzenie typu IoT, które wykorzystuje modele konwolucyjnych sieci neuronowych oparte o algorytm *Single Shot Detector* i umożliwia występowanie dręcza pszczelego. Praktyczna realizacja problemu badawczego została poprzedzona dogłębną analizą potrzeb pszczelarzy, aspektów biologicznych poruszanej tematyki, a także rozwiązań opisanych w światowej literaturze naukowej. Wśród elementów autorskiego wkładu badawczego, które odróżniają proponowany system od zaprezentowanych do tej pory systemów są:

- zaprojektowane urządzenie IoT, w którym zastosowano minikomputer NVIDIA Jetson Nano, dedykowany do przetwarzania obrazów oraz wykorzystanie kamery bez filtra podczerwonego, wyposażonej w moduł diod

IR LED, które umożliwia pozyskiwanie obrazów w dobrej jakości przy różnych poziomach oświetlenia ula,

- zastosowanie modeli głębokiego uczenia maszynowego opartych o algorytm *Single Shot Detector* i wykorzystanie ich do procesów detekcji pszczoł, a następnie ich klasyfikacji – uzyskano lepsze wyniki w porównaniu do wcześniej przeprowadzonych badań [109], [110],
- opracowanie oraz zastosowanie algorytmu adaptacyjnego wyznaczania okna zagęszczenia, który umożliwia dopasowanie parametrów działania urządzenia brzegowego do aktualnie zbieranych i analizowanych obrazów, z uwzględnieniem liczby pszczoł znajdujących się w przetwarzanym obszarze ramki. Ponadto, w procesie wyznaczania algorytmu adaptacyjnego wyznaczania okna zagęszczenia określany jest parametr systemu definiujący minimalną liczbę klatek, które należy pominąć, tak, aby urządzenie działało w czasie rzeczywistym,
- zgromadzenie indywidualnego zbioru danych, ich wstępne przygotowanie, a następnie wykorzystanie ich do budowy modeli sztucznych sieci neuronowych wykorzystujących sieć SSD do detekcji pszczoł oraz detekcji obecności dręcza pszczelego na ciele pszczoły.

Ponadto niniejsza rozprawa i przeprowadzone badania niosą za sobą wkład w upowszechnienie nowoczesnych technologii w pszczelarstwie, i tym samym rozwój pszczelarstwa precyzyjnego. Jest to ważny również aspekt, zwłaszcza, że zarówno definicja założeń dotyczących urządzenia IoT, jak również gromadzenie danych badawczych i przeprowadzane w ramach realizacji rozprawy eksperymenty były konsultowane z pszczelarzami, którzy określili specyfikę problemu i byli zaangażowali w każdy z wymienionych etapów.

6.3. Możliwe kierunki rozwoju

Zaprezentowany system i przeprowadzone eksperymenty badawcze, pomimo szerokiego zakresu wykonanych prac, nadal pozostawiają możliwości dalszego rozwoju. Wśród potencjalnych kierunków przyszłych prac należy wymienić przede wszystkim

rozszerzenie działania systemu o analizę obrazu pod kątem innych aspektów oraz o gromadzenie i przetwarzanie danych innych niż obraz. Dręcz pszczeleli, mimo niewątpliwego negatywnego wpływu na zdrowie i życie pszczół nie jest dla nich jedynym zagrożeniem. Pozostając przy dalszej analizie obrazów możliwe jest rozszerzenie działania systemu o wykrywanie zarażeniem innymi pasożytami bądź też rozpoznawanie pszczół z deformacjami skrzydeł lub tułowia. Przy wykorzystaniu kilku urządzeń IoT takich, jak te zaprezentowane w niniejszej rozprawie, i rozszerzeniu obserwacji na kilka uli jednocześnie można przeprowadzić analizę zmiany pozycji pszczół w kolejnych klatkach. Jeśli pozycje wykrytych obiektów nie będą się zmieniały, prawdopodobne jest, że pszczoły są martwe i, gdy zostanie to zaobserwowane na kilku urządzeniach, może wskazywać na wystąpienie zatrucia lub rodzaju paraliżu pszczół.

Innym możliwym kierunkiem dalszego rozwoju jest połączenie przetwarzania danych graficznych i dźwiękowych. Przykładowo, analiza akustyczna dźwięków połączona z przetwarzaniem obrazów z okolicy ula mogłaby pomóc pszczelarzom w wykrywaniu okresu rojenia się pszczół albo obecności niepożądanych w pasiece intruzów. Wśród rozwojowych aspektów powyższej propozycji można wymienić utworzenie modelu sztucznej sieci neuronowej umożliwiającego analizę obu typów danych.

Ze względu na sposób powiadamiania pszczelarzy, możliwym kierunkiem rozwoju systemu jest implementacja aplikacji mobilnej, obsługującej powiadomienia wysyłane za pomocą usług Amazon AWS. Co więcej, możliwe byłoby dostosowanie przygotowanych modeli uczenia maszynowego do działania na urządzeniach mobilnych, poprzez konwersję do odpowiednich formatów. Zastosowany model sieci neuronowej SSD MobileNetV2 Lite, zaimplementowany z wykorzystaniem biblioteki Tensorflow może być wyeksportowany do formatu, który umożliwi uruchomienie procesu wnioskowania na urządzeniach mobilnych. Takie rozwiązanie stanowiłoby uzupełnienie działania systemu i mogłoby być wykorzystywane przez pszczelarzy w procesie przeglądania ramek, który zostałby zainicjowany wiadomością e-mail z informacją o wykryciu dręcza pszczelego w ulu.

Publikacje Autorki w trakcie realizacji rozprawy doktorskiej

I. Artykuły z czasopism:

1. Wachowicz Anna, Pytlik Jakub, Małysiak-Mrozek Bożena, Tokarz Krzysztof, Mrozek Dariusz: Edge computing in IoT-enabled honeybee monitoring for the detection of Varroa destructor, *International Journal of Applied Mathematics & Computer Science*, 2022, vol. 32, nr 3, s.355-369. DOI:10.34768/amcs-2022-0026

liczba punktów MNiSW: 100

IF: 1,9

2. Mrozek Dariusz, Górny Rafał, Wachowicz Anna, Małysiak-Mrozek Bożena: Edge-based detection of varroosis in beehives with IoT devices with embedded and TPU-accelerated machine learning, *Applied Sciences-Basel*, 2021, vol. 11, nr 22, s.1-22, Numer artykułu:11078. DOI:10.3390/app112211078

liczba punktów MNiSW: 100,

IF: 2,838

II. Rozdziały z monografii:

1. Wachowicz Anna: IoT monitoring systems in beekeeping, W: *Networking issues in innovative applications based on cyber-physical systems paradigm : Praca zbiorowa / Stój Jacek, Pułka Andrzej (red.)*, Monografia, [Politechnika Śląska], 2020, vol. 874, Politechnika Śląska, s.222-238, ISBN 978-83-7880-736-0

liczba punktów MNiSW: 20

2. Wachowicz Anna, Mrozek Dariusz: Fuzzy join as a preparation step for the analysis of training data, W: *Beyond databases, architectures and structures : Paving the road to smart data processing and analysis. 15th International*

conference, BDAS 2019, Ustroń, Poland, May 28-31, 2019. Proceedings / Kozielski Stanisław [i in.] (red.), Communications in Computer and Information Science, 2019, vol. 1018, Springer, s.263-273, ISBN 978-3-030-19092-7. DOI:10.1007/978-3-030-19093-4_20

liczba punktów MNiSW: 20

III. Materiały konferencyjne:

1. Wachowicz Anna, Małysiak-Mrozek Bożena, Mrozek Dariusz: Combining data from fitness trackers with meteorological sensor measurements for enhanced monitoring of sports performance, W: Computational Science - ICCS 2019 : 19th International Conference, Faro, Portugal, June 12-14, 2019. Proceedings / Rodrigues J. M. F. [i in.] (red.), Lecture Notes In Computer Science, 2019, vol. 11538, Springer, s.692-705, ISBN 978-3-030-22743-2. DOI:10.1007/978-3-030-22744-9_54

liczba punktów MNiSW: 140

Bibliografia

- [1] J. P. van der Sluijs, N. Simon-Delso, D. Goulson, L. Maxim, J.-M. Bonmatin i L. P. Belzunces, „Neonicotinoids, bee disorders and the sustainability of pollinator services,” *Current Opinion in Environmental Sustainability*, tom 5, pp. 293-305, 2013.
- [2] R. S. Cornman, D. R. Tarpy, Y. Chen, L. Jeffreys, D. Lopez, J. S. Pettis, D. vanEngelsdorp i J. D. Evans, „Pathogen Webs in Collapsing Honey Bee Colonies,” *PLoS ONE*, pp. 1-15, 2012.
- [3] D. vanEngelsdorp, K. S. Traynor, M. Andree, E. M. Lichtenberg, Y. Chen, C. Saegerman i D. L. Cox-Foster, „Colony Collapse Disorder (CCD) and bee age impact honey bee pathophysiology,” *PLoS ONE*, pp. 1-23, 2017.
- [4] A. Zacepins, A. Kvišis, A. Pecka i V. Osadcuks, „Development of Internet of Things concept for Precision Beekeeping,” w *18th International Carpathian Control Conference (ICCC)*, Sinaia, Romania, 2017.
- [5] N. Zogovic, M. Mladenovic i S. Rašić, „From Primitive to Cyber-Physical Beekeeping,” w *ICIST 2017 Proceedings Vol.1*, Kopaonic, Serbia, 2017.
- [6] E. Guzmán-Novoa, L. Eccles, Y. Calvete, J. McGowan, P. G. Kelly i A. Correa-Benítez, „Varroa destructor is the main culprit for the death and reduced populations of overwintered honey bee (*Apis mellifera*) colonies in Ontario, Canada,” *Apidologie*, tom 41, pp. 443-450, 2010.
- [7] A. B. Barron, „Death of the bee hive: understanding the failure of an insect society,” *Current Opinion in Insect Science*, tom 10, pp. 45-50, 2015.
- [8] S. Trzybiński, Współczesna gospodarka pasieczna. Tom I: Organizacja pasieki, produkty pszczele, Kęty: Bee & Honey sp. z o.o., 2013.
- [9] M. A. Aizen, L. A. Garibaldi, S. A. Cunningham i A. M. Klein, „How much does agriculture depend on pollinators? Lessons from long-term trends in crop production,” *Annals of Botany*, tom 103, pp. 1579-1588, 2009.
- [10] A.-M. Klein, B. E. Vaissière, J. H. Cane, I. Steffan-Dewenter, S. A. Cunningham, C. Kremen i T. Teja, „Importance of pollinators in changing landscapes for world crops,” *Proceedings of the Royal Society B*, tom 274, pp. 303-313, 2007.
- [11] S. W. T. Batra, „Bees and pollination in our changing environment,” *Apidologie*, tom 26, pp. 361-370, 1995.
- [12] M. Spivak, E. Mader, M. Vaughan i N. H. J. Eullis, „The plight of the bees,” *Environmental Science & Technology*, tom 45, pp. 34-38, 2011.
- [13] P. Rhoades, „The importance of bees in natural and agricultural ecosystems,” w *National Proceedings: Forest and Conservation Nursery Associations - 2012*, Fort Collins, 2013.
- [14] C. E. Aslan, C. T. Liang, B. Galindo, H. Kimberly i W. Topete, „The role of honey bees as pollinators in natural areas,” *Natureal Areas Journal*, tom 36, pp. 478-488, 2016.
- [15] T. E. Murray, M. Kuhlmann i S. G. Potts, „Conservation ecology of bees:

-
- populations, species and communities,” *Apidologie*, tom 40, pp. 211-236, 2009.
- [16] T. D. Seeley, „The Honey Bee Colony as a Superorganism,” *American Scientist*, tom 77, nr 6, p. 546–53, 1989.
- [17] J. Bryden, R. J. Gill i R. A. A. Mitt, „Chronic sublethal stress causes bee colony failure,” *Ecology Letters*, tom 16, pp. 1463-1469, 2013.
- [18] V. Patel, N. Pauli, E. Biggs, L. Barbour i B. Boruff, „Why bees are critical for achieving sustainable development,” *Ambio*, tom 50, pp. 49-59, 2021.
- [19] Organizacja Narodów Zjednoczonych, „Rezolucja zgromadzenia ogólnego ONZ: 70/1. Transforming our world: the 2030 Agenda for Sustainable Development,” 25 09 2015. [Online]. Available: www.un.org/ga/search/view_doc.asp?symbol=A/RES/70/1&Lang=E. [Data uzyskania dostępu: 2021].
- [20] D. van Engelsdorp, R. Underwood, D. Caron i J. Hayes, „An Estimate of Managed Colony Losses in the Winter of 2006 - 2007: A Report Commissioned by the Apiary Inspectors of America,” *American Bee Journal*, tom 147, pp. 599-600, 2007.
- [21] I. Potamitis, I. Rigakis, N.-A. Tatlas i S. Kouzoupis, „A novel electronic gate that identifies and counts bees based on their RGB backscattered light,” w *23rd International Conference on Circuits, Systems, Communications and Computer*, Athens, 2019.
- [22] B. Daiant, D. vanEngelsdorp i P. Neumann, „Collony collapse disorder in Europe,” *Environmental microbiology reports*, tom 4, pp. 123-125, 2012.
- [23] R. Johnson, „Honey bee colony collapse disorder.,” Congressional Research Service, Washington, 2010.
- [24] S. M. Swinton, F. Lupi, G. P. Robertson i S. K. Hamilton, „Ecosystem services and agriculture: Cultivating agricultural ecosystems for diverse benefits,” *Ecological Economics*, tom 64, pp. 245-252, 2007.
- [25] Y. Le Conte i M. Navajas, „Climate change: Impact on honey bee populations and diseases,” *Revue scientifique et technique (International Office of Epizootics)*, tom 27, pp. 499-510, 2008.
- [26] G. Maxence, V. Maryse, T. Wood i D. Michez, „Global warming and plant–pollinator mismatches,” *Emerging Topics in Life Sciences*, tom 4, nr 1, pp. 77-86, 2020.
- [27] E. Maori, N. Paldi, S. Shafir, H. Kalev, E. Tsur, E. Glick i I. Sela, „IAPV, a bee-affecting virus associated with Colony Collapse Disorder can be silenced by dsRNA ingestion,” *Insect molecular biology*, tom 18, pp. 55-60, 2009.
- [28] J. R. de Miranda, G. Cordoni i G. Budge, „The Acute bee paralysis virus–Kashmir bee virus–Israeli acute paralysis virus complex,” *Journal of Invertebrate Pathology*, Tomy %1 z %2Volume 103, Supplement,, pp. 30-47, 2010.
- [29] E. Maori, R. Mozes-Koch, Y. Gantman, Y. Peretz, O. Edelbaum, E. Tanne i I. Sela, „Isolation and characterization of Israeli acute paralysis virus, a dicistrovirus affecting honeybees in Israel: evidence for diversity due to intra- and inter-species recombination,” *Journal of General Virology*, tom 88, p. 3428–3438, 2007.
- [30] Y. P. Chen, J. S. Pettis, M. Corona, W. P. Chen, C. J. Li, M. Spivak, P. K. Visscher i G. DeGrandi-Hoffmann, „Israeli Acute Paralysis Virus: Epidemiology, Pathogenesis and Implications for Honey Bee Health,” *PLoS Pathogens*, tom 10, pp. 1-15, 2014.
- [31] M. Goblirsch, „Nosema ceranae disease of the honey bee (*Apis mellifera*),” *Apidologie*, tom 49, pp. 131-150, 2018.

- [32] R. J. Paxton, „Does infection by *Nosema ceranae* cause "Colony Collapse Disorder in honey bees (*Apis mellifera*)"?,” *Journal of Apicultural Research*, tom 49, nr 1, pp. 80-84, 2010.
- [33] I. Fries i F. Feng, „Cross-infectivity of *Nosema ceranae* in *Apis mellifera* and *Apis cerana*,” w *Proceedings of the 34th International Apicultural Congress, Apimondia*, Bucharest, Romania, 1995.
- [34] I. Fries, F. Feng, A. D. Silva, S. S. B i N. J. Pieniasek, „*Nosema ceranae* n. sp. (Microspora, Nosematidae), morphological and molecular characterization of a microsporidian parasite of the Asian honey bee *Apis cerana* (Hymenoptera, Apidae).,” *European Journal of Protistology*, pp. 356-365, 1996.
- [35] I. Fries, R. Martín Hernández, A. Meana, P. García-Palencia i M. Higes, „Natural infections of *Nosema ceranae* in European honey bees,” *Journal of Apicultural Research*, pp. 230-233, 2006.
- [36] V. Doublet, M. Labarussias, J. R. de Miranda, R. F. A. Mortiz i R. J. Paxton, „Bees under stress: Sublethal doses of a neonicotinoid pesticide and pathogens interact to elevate honey bee mortality across the life cycle,” *Environmental Microbiology*, p. 969–983, 2015.
- [37] M. Frazier, C. Mullin, J. Frazier i S. Ashcraft, „What Have Pesticides Got to Do with It?,” *American Bee Journal*, tom 148, nr 6, pp. 1-5, 2008.
- [38] A. Decourtye, J. Devillers, P. Aupinel, F. Brun, C. Bagnis, J. Fourrier i M. Gauthier, „Honeybee tracking with microchips: a new methodology,” *Ecotoxicology*, tom 20, p. 29–437, 2011.
- [39] R. M. Francis, S. L. Nielsen i P. Kryger, „Varroa-Virus Interaction in Collapsing Honey Bee Colonies,” *PLoS ONE*, tom 8, pp. 1-9, 2013.
- [40] G. Borsuk, K. Czerska, K. Olszewski, A. Strachecka, J. Paleolog i J. Chobotow, „Current state of knowledge of *Varroa destructor*,” *Medycyna Weterynaryjna*, tom 68, pp. 579-584, 2012.
- [41] S. D. Ramsey, R. Ochoa, G. Bauchan, C. Gulbranson, J. D. Mowery, A. Cohen, D. Lim, J. Joklik, J. M. Cicero, J. D. Ellis, D. Hawthorne i D. vanEngelsdorp, „*Varroa destructor* feeds primarily on honey bee fat body tissue and not hemolymph,” *Proceedings of the National Academy of Sciences of the United States of America*, tom 116, nr 5, p. 1792–1801, 2019.
- [42] P. Rosenkranz, P. Aumeier i B. Ziegelmann, „Biology and control of *Varroa destructor*,” *Journal of Invertebrate Pathology*, tom 103, pp. 96-119, 2010.
- [43] A. Noël, Y. Le Conte i M. Fanny, „*Varroa destructor*: how does it harm *Apis mellifera* honey bees and what can be done about it?,” *Emerging Topics in Life Sciences*, tom 4, nr 1, p. 45–57, 2020.
- [44] S. Trzybiński, *Współczesna gospodarka pasieczna. Tom II: Rok w pasiece, Kęty: Bee & Honey sp. z o.o.*, 2013.
- [45] G. Goras, C. H. Tananaki, S. Gounari, M. Dimou, E. Lazaridou, E. Karazafiris, D. Kanelis, V. Liolios, H. El Taj i A. & Thrasylvoulou, „Hyperthermia -a non-chemical control strategy against varroa,” *Journal of the Hellenic Veterinary Medical Society*, tom 66, nr 4, p. 249–256, 2018.
- [46] A. Kablau, S. Berg, S. Härtel i R. & Scheiner, „Hyperthermia treatment can kill immature and adult *Varroa destructor* mites without reducing drone fertility,” *Apidologie*, tom 51, pp. 307-315, 2020.
- [47] „Internet World Stats,” MiniWatts Marketing Group, 21 Styczeń 2023. [Online].

Available: <https://www.internetworldstats.com/>. [Data uzyskania dostępu: 30 Czerwiec 2023].

- [48] M. Hermann, T. Pentek i B. Otto, „Design Principles for Industrie 4.0 Scenarios,” w *49th Hawaii International Conference on System Sciences*, 2016.
- [49] M. Mohamed, „Challenges and Benefits of Industry 4.0: an overview.,” *International Journal of Supply and Operations Management*, tom 5, nr 3, pp. 256-265, 2018.
- [50] D. Ding, Q.-L. Han, Y. Xiang, X. Ge i X.-M. Zhang, „A survey on security control and attack detection for industrial cyber-physical systems,” *Neurocomputing*, pp. 1674-1683, 31 01 2018.
- [51] F. D. Rinkevich, R. G. Danko i K. B. Healy, „Influence of Varroa Mite (*Varroa destructor*) Management Practices on Insecticide Sensitivity in the Honey Bee (*Apis mellifera*),” *Insects*, pp. 1-12, 2017.
- [52] K. Ashton, „That “Internet of Things” Thing: In the Real World Things Matter More than Ideas,” *RFID Journal*, 2009.
- [53] Y.-K. Chen, „Challenges and opportunities of internet of things,” w *17th Asia and South Pacific Design Automation Conference*, 2012.
- [54] E. Sisinni, A. Saifullah, S. Han, U. Jennehag i M. Gidlund, „Industrial Internet of Things: Challenges, Opportunities, and Directions.,” *IEEE Transactions on Industrial Informatics*, pp. 1-27, 2018.
- [55] L. Tawalbeh, F. Muheidat, M. Tawalbeh i M. Quwaider, „IoT Privacy and Security: Challenges and Solutions,” *Applied Sciences*, pp. 1-17, 15 06 2020.
- [56] A. R. Sfar, E. Natalizio, Y. Challal i Z. Chtourou, „A roadmap for security challenges in the Internet of Things.,” *Digital Communications and Networks*, pp. 118-137, 2018.
- [57] H. Xu, W. Yu, D. Griffith i N. Golmie, „A Survey on Industrial Internet of Things: A Cyber-Physical Systems Perspective,” *IEEE Access*, pp. 78238-78259, 2018.
- [58] R. Baheti i H. Gill, „Cyber-physical systems,” *The impact of control technology*, tom 12, nr 1, pp. 161-166, 2011.
- [59] I. Lee, O. Sokolsky, S. Chen, J. Hatcliff, E. Jee, B. Kim, A. King, M. Mullen-Fortino, S. Park, A. Roederer i K. K. Venkatasubramanian, „Challenges and Research Directions in Medical Cyber-Physical Systems,” *Proceedings of the IEEE*, pp. 75-90, 2011.
- [60] M. S. Mahmoud, M. M. Hamdan i U. A. Baroudi, „Modeling and control of Cyber-Physical Systems subject to cyber attacks: A survey of recent advances and challenges,” *Neurocomputing*, pp. 101-115, 2019.
- [61] P. Wang, „On defining artificial intelligence,” *Journal of Artificial General Intelligence*, pp. 1-37, 2019.
- [62] P. Ongsulee, „Artificial Intelligence, Machine Learning and,” w *15th International Conference on ICT and Knowledge Engineering (ICT&KE)*, 2017.
- [63] W. Shi, J. Cao, Q. Zhang, Y. Li i L. Xu, „Edge Computing: Vision and Challenges,” *IEEE Internet of Things Journal*, pp. 637-646, 2016.
- [64] W. Shi i S. Dustdar, „The Promise of Edge Computing,” *Computer*, pp. 78-81, 2016.
- [65] B. Varghese, N. Wang, S. Barbhuiya, P. Kilpatrick i D. Nikolopoulos, „Challenges and Opportunities in Edge Computing,” *IEEE SmartCloud*, pp. 1-6, 2016.
- [66] T. Popović, N. Latinović, A. Pešić, Ž. Zečević, B. Krstajić i S. Djukanović, „Architecting an IoT-enabled platform for precision agriculture and ecological

- monitoring: A case study,” *Computers and Electronics in Agriculture*, pp. 255-265, 2017.
- [67] B. N. Gates, *The Temperature of the Bee Colony*, Waszyngton D.C.: Bulletin of the U.S. Department of Agriculture, 1914, pp. 1-29.
- [68] W. E. Dunham, „Hive temperatures for each hour of a day,” *The Ohio Journal of Science*, pp. 181-188, 1931.
- [69] A. E. Lundie, *The flight activities of the honey bee*, Waszyngton, D.C.: United States Department of Agriculture, 1925.
- [70] W. H. Brittain, *Apple pollination studies in the Annapolis Valley*, Kanada: Dominion of Canada, Department of Agriculture, 1933.
- [71] H. G. Spangler, „Photoelectrical Counting of Outgoing and Incoming Honey Bees,” *Journal of Economic Entomology*, p. 1183–1184, 1 10 1969.
- [72] E. H. Erickson, H. H. Miller i D. J. Sikkema, „A Method of Separating and Monitoring Honeybee Flight Activity at the Hive Entrance,” *Journal of Apicultural Research*, pp. 119-125, 1975.
- [73] C. Liu, J. J. Leonard i J. J. Feddes, „Automated Monitoring of Flight Activity at a Beehive Entrance using Infrared Light Sensors,” *Journal of Apicultural Research*, pp. 20-27, 1990.
- [74] M. H. Struye, H. J. Mortier, G. Arnold, C. Miniggio i R. Borneck, „Microprocessor-controlled monitoring of honeybee flight activity at the hive entrance,” *Apidologie*, pp. 384-395, 1994.
- [75] A. Feldman i T. Balch, „Representing Honey Bee Behavior for Recognition Using Human Trainable Models,” *Adaptive Behavior*, p. 241–250, 1 12 2004.
- [76] T. Landgraf i R. Rojas, „Tracking honey bee dances from sparse optical flow fields,” Free Berlin Univeristy, Berlin, 2007.
- [77] J. Campbell, L. Mummert i R. Sukthankar, „Video Monitoring of Honey Bee Colonies at the Hive Entrance,” w *19th International Conference on Pattern Recognition - Visual observation and analysis of animal and insect behavior*, Tampa, Floryda, 2008.
- [78] A. Zacepins, E. Stalidzans i J. Meitalovs, „Application of information technologies in precision apiculture,” w *Proceedings of the 13th International Conference on Precision Agriculture*, Indianapolis, USA, 2012.
- [79] A. Zacepins, A. Kvišis, E. Stalidzans, M. Liepniece i J. Meitalovs, „Remote detection of the swarming of honey bee colonies by single-point temperature monitoring,” *Biosystems Engineering*, pp. 76-80, 2016.
- [80] S. Gil-Lebrero, F. Quiles-Latorre, M. Ortiz-López, V. Sánchez-Ruiz, V. Gámiz-López i J. J. Luna-Rodríguez, „Honey Bee Colonies Remote Monitoring System,” *Sensors*, pp. 1-21, 29 12 2016.
- [81] A. Zacepins, A. Pecka, V. Osadcuks, S. Kvišis i A. Engel, „Solution for automated bee colony weight monitoring,” *Agronomy Research*, pp. 585-593, 2017.
- [82] A. Braga, E. Hassler, D. Gomes, B. Freitas i J. Cazier, „IoT for Development: Building a Classification Algorithm to Help Beekeepers Detect Honeybee Health Problems Early.” w *Twenty-fifth Americas Conference on Information Systems*, Cancun, 2019.
- [83] S. Kontogiannis, „An Internet of Things-Based Low-Power Integrated Beekeeping Safety and Conditions Monitoring System,” *Inventions*, pp. 1-29, 2019.

-
- [84] D. J. Rustia, N. Ngo i T.-T. Lin, „An IoT-based information system for honeybee in and out activity with beehive environmental condition monitoring,” w *Conference on Bio-Mechatronics and Agricultural Machinery Engineering*, 2016.
- [85] F. Edwards-Murphy, M. Magno, P. M. Whelan, J. O’Halloran i E. M. Popovici, „b+WSN: Smart beehive with preliminary decision tree analysis for agriculture and honey bee health monitoring,” *Computers and Electronics in Agriculture*, pp. 211-219, 2016.
- [86] O. Debauche, M. E. Moulat, S. Mahmoudi, S. Boukraa, P. Manneback i F. Lebeau, „Web Monitoring of Bee Health for Researchers and Beekeepers Based on the Internet of Things,” w *The 8th International Symposium on Frontiers in Ambient and Mobile Systems (FAMS 2018)*, Porto, 2018.
- [87] J. Meitalovs, A. Histjajevs i E. Stalidzans, „Automatic microclimate controlled beehive observation system,” w *Engineering for Rural Development. Proceedings of the International Scientific Conference*, Jelgava, 2009.
- [88] I. Z. Ochoa, S. Gutierrez i F. Rodríguez, „Internet of Things: Low Cost Monitoring BeeHive System using Wireless Sensor Network,” w *International Conference on Engineering Veracruz*, Veracruz, 2019.
- [89] R. Machhamer, J. Altenhofer, K. Ueding, L. Czenkusch, F. Stolz, M. Harth, M. Mattern, A. Latif, S. Haab, J. S. A. G. K. Herrmann i G. Dartmann, „Visual programmed Iot beehive monitoring for decision aid by machine learning based anomaly detection,” w *2020 9th Mediterranean Conference on Embedded Computing (MECO)*, 2020.
- [90] D. A. Mezquida i J. L. Martínez, „Short communication. Platform for bee-hives monitoring based on sound analysis. A perpetual warehouse for swarm’s daily activity,” *Spanish Journal of Agricultural Research*, pp. 824-828, 2009.
- [91] A. Qandour, I. Ahmad, D. Habibi i M. Leppard, „Remote beehive monitoring using acoustic signals,” *Acoustics Australia*, pp. 204-209, 2014.
- [92] A. König, „Indusbee 4.0 – integrated intelligent sensory systems for advanced bee hive instrumentation and hive Keepers’ Assistance Systems,” *Sensors & Transducers*, pp. 109-121, 30 11 2019.
- [93] A. Zgank, „Bee Swarm Activity Acoustic Classification for an IoT-Based Farm Service,” *Sensors*, pp. 1-14, 2019.
- [94] A. Imoize, S. Odeyemi i J. Adebisi, „Development of a Low-Cost Wireless Bee-Hive Temperature and Sound Monitoring System.,” *Indonesian Journal of Electrical Engineering and Informatics*, pp. 476-485, 2020.
- [95] R. Tashakkori, A. Hamza i M. Crawford, „Beemon: An IoT-based beehive monitoring system,” *Computers and Electronics in Agriculture*, p. 106427, 2021.
- [96] C. Chen, E.-C. Yang, J.-A. Jiang i T.-T. Lin, „An imaging system for monitoring the in-and-out activity of honey bees,” *Computers and Electronics in Agriculture*, pp. 100-109, 2012.
- [97] Z. Babic, R. Pilipovic, V. Risojević i G. Mirjanic, „Pollen bearing honey bee detection in hive entrance video recorded by remote embedded system for pollination monitoring,” w *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, Praga, 2016.
- [98] T. N. Ngo, D. J. A. Rustia, E.-C. Yang i T.-T. Lin, „Automated monitoring and analyses of honey bee pollen foraging behavior using a deep learning-based imaging system,” *Computers and Electronics in Agriculture*, pp. 1-14, 09 06 2021.

- [99] J. Marstaller, F. Tausch i S. Stock, „Deepbees - building and scaling convolutional neuronal nets for fast and large-scale visual monitoring of bee hives,” w *International Conference on Computer Vision*, Seul, 2019.
- [100] Y.-L. Chen, H.-Y. Chien, T.-H. Hsu, Y.-J. Jing, C.-Y. Lin i Y.-C. Lin, „A pi-based beehive IoT system design,” w *Security with Intelligent Computing and Big-data Services*, Cham, 2020.
- [101] V. Elizondo, J. C. Briceño, C. M. Travieso i J. B. Alonso, „Video Monitoring of a Mite in Honeybee Cells,” *Advanced Materials Research*, pp. 1107-1114, 2013.
- [102] S. Schurischuster, B. Remeseiro, P. Radeva i M. Kampel, „A Preliminary Study of Image Analysis for Parasite Detection on Honey Bees,” w *Image Analysis and Recognition. ICIAR 2018. Lecture Notes in Computer Science()*, Cham, 2018.
- [103] K. Bjerge, C. E. Frigaard, P. H. Mikkelsen, T. H. Nielsen, M. Misbih i P. Kryger, „A computer vision system to monitor the infestation level of Varroa destructor in a honeybee colony,” *Computers and Electronics in Agriculture*, pp. 1-10, 2019.
- [104] A. Szczurek, M. Maciejewska, J. Wilk, J. Wilde i M. Siuda, „Detection of Honeybee Disease: Varroosis using a Semiconductor Gas Sensor Array,” w *SENSORNETS 2019: Proceedings of the 8th International Conference on Sensor Networks*, Setubal, 2019.
- [105] A. Szczurek, M. Maciejewska, Ż. Zajiczek, B. Bąk, J. Wilk, J. Wilde i M. Siuda, „The Effectiveness of Varroa destructor Infestation Classification Using an E-Nose Depending on the Time of Day,” *Sensors*, pp. 1-18, 29 04 2020.
- [106] M. Bencsik, J. Bencsik, M. Baxter, A. Lucian, J. Romieu i M. Millet, „Identification of the honey bee swarming process by analysing the time course of hive vibrations,” *Computers and Electronics in Agriculture*, pp. 44-50, 2011.
- [107] S. Van Goethem, S. Verwulgen, F. Goethijn i J. Steckel, „An IoT solution for measuring bee pollination efficacy,” w *2019 IEEE 5th World Forum on Internet of Things (WF-IoT)*, 2019.
- [108] A. Zgank, „IoT-Based Bee Swarm Activity Acoustic Classification Using Deep Neural Networks,” *Sensors*, pp. 1-14, 20 01 2021.
- [109] D. Mrozek, R. Górny, A. Wachowicz i B. B. Małyśiak-Mrozek, „Edge-Based Detection of Varroosis in Beehives with IoT Devices with Embedded and TPU-Accelerated Machine Learning,” *Applied Sciences*, pp. 1-22, Listopad 2021.
- [110] A. Wachowicz, J. Pytlik, B. Małyśiak-Mrozek, K. Tokarz i D. Mrozek, „Edge computing in an IoT-enabled honeybee monitoring for the detection of Varroa destructor,” *International Journal of Applied Math and Computer Sciences*, pp. 355-369, 20 Maj 2022.
- [111] Z. Zivkovic, „Improved adaptive Gaussian mixture model for background subtraction,” w *Proceedings of the 17th International Conference on Pattern Recognition*, Cambridge, UK, 2004.
- [112] Z. Zivkovic i F. van der Heijden, „Efficient adaptive density estimation per image pixel for the task of background subtraction,” *Pattern Recognition Letters*, tom 27, pp. 773-780, 2006.
- [113] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama i T. Darrell, „Caffe: Convolutional Architecture for Fast Feature Embedding,” w *Proceedings of the 22nd ACM international conference on Multimedia*, 2014.
- [114] „<https://developer.nvidia.com/tensorrt>,” NVIDIA Corporation, [Online]. Available: <https://developer.nvidia.com/tensorrt>. [Data uzyskania dostępu: 01 08 2023].

-
- [115] „NVIDIA Jetson Nano - specyfikacja techniczna,” NVIDIA, [Online]. Available: <https://www.nvidia.com/pl-pl/autonomous-machines/embedded-systems/jetson-nano/product-development/>. [Data uzyskania dostępu: 30 Czerwiec 2023].
- [116] „Moduł kamery IMX219-160IR - specyfikacja techniczna,” [Online]. Available: <https://www.waveshare.com/product/imx219-160ir-camera.htm>. [Data uzyskania dostępu: 30 Czerwiec 2023].
- [117] „Karta sieciowa TP-link - specyfikacja techniczna,” TP-Link, [Online]. Available: <https://www.tp-link.com/pl/home-networking/adapter/tl-wn725n/#specifications>. [Data uzyskania dostępu: 30 Czerwiec 2023].
- [118] G. Van Rossum i F. L. Drake, „Python 3 Reference Manual,” CreateSpace, Scotts Valley, CA, 2009.
- [119] „TensorFlow: Large-scale machine learning on heterogeneous systems,” Software available from tensorflow.org, 2015.
- [120] J. Redmon, S. Divvala, R. Girshick i F. Ali, „You Only Look Once: Unified, Real-Time Object Detection,” w *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, USA, 2016.
- [121] R. Girshick, „Fast R-CNN,” *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 1440-1448, 18 Luty 2016.
- [122] S. Ren, K. He, R. Girshick i J. Sun, „Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” w *Advances in Neural Information Processing Systems 28*, Montreal, Canada, 2015.
- [123] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu i A. C. Berg, „SSD: Single Shot MultiBox Detector,” w *Computer Vision ECCV 2016*, Springer International Publishing, 2016, pp. 21-37.
- [124] R. Girshick, J. Donahue, T. Darrell i J. Malik, „Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation,” w *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, USA, 2014.
- [125] K. Simonyan i A. Zisserman, „Very Deep Convolutional Networks for Large-Scale Image Recognition,” arXiv preprint [arXiv:1409.1556](https://arxiv.org/abs/1409.1556), 2015.
- [126] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn i A. Zisserman, „PASCAL VOC 2007,” 2007. [Online]. Available: <http://host.robots.ox.ac.uk/pascal/VOC/index.html>. [Data uzyskania dostępu: 30 Czerwiec 2023].
- [127] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov i L.-C. Chen, „MobileNetV2: Inverted Residuals and Linear Bottlenecks,” *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4510-4520, Marzec 2018.
- [128] A. Buslaev, V. Iglovikov, E. Khvedchenya, A. Parinov, M. Druzhinin i A. A. Kalinin, „Albumentations: Fast and Flexible Image Augmentations,” *Information*, pp. 1-20, 2020.

Spis skrótów i symboli

<i>IoT</i>	Systemy Internetu Rzeczy (ang. <i>Internet of Things</i>)
<i>IIoT</i>	Systemy Przemysłowego Internetu Rzeczy (ang. <i>Industrial Internet of Things</i>)
<i>CCD</i>	zespół masowego ginięcia pszczoły miodnej (ang. <i>Colony Collapse Disorder</i>)
<i>PB lub PA</i>	przeczelarstwo precyzyjne (ang. <i>Precision Beekeeping, Precision Apiculture</i>)
<i>GPU</i>	procesor graficzny (ang. <i>Graphics Processing Unit</i>)
<i>SDG</i>	cele zrównoważonego rozwoju (ang. <i>Sustainable Development Goals</i>)
<i>WEF</i>	Światowe Forum Ekonomiczne (ang. <i>World Economic Forum</i>)
<i>KBV</i>	wirus kaszmirski (ang. <i>Kashmir Bee Virus</i>)
<i>IAPV</i>	wirus izraelskiego ostrego paraliżu pszczół (ang. <i>Israeli acute paralysis virus</i>)
<i>ABPV</i>	wirus ostrego paraliżu pszczół (ang. <i>acute bee paralysis virus</i>)
<i>DWV</i>	wirus zdeformowanych skrzydeł (ang. <i>deformed wing virus</i>)
<i>SOA</i>	architektura oparta na usługach (ang. <i>Service-Oriented Architecture</i>)
<i>M2M</i>	usługa komunikacji między urządzeniami w oparciu o sieć komórkową (ang. <i>Machine To Machine</i>)
<i>CPS</i>	Systemy cyber-fizyczne (ang. <i>Cyber – Physical Systems</i>)
<i>SCADA</i>	System umożliwiający kontrolowanie działanie i przetwarzania danych w procesach produkcyjnych (ang. <i>Supervisory Control And Data Acquisition</i>)
<i>PLC</i>	programowalny sterownik logiczny (ang. <i>programmable logic controller</i>)
<i>AI</i>	sztuczna inteligencja (ang. <i>Artificial Intelligence</i>)
<i>NLP</i>	przetwarzanie języka naturalnego (ang. <i>natural language processing</i>)
<i>ML</i>	uczenie maszynowe (ang. <i>machine learning</i>)
<i>CNN</i>	konwolucyjna sieć neuronowa (ang. <i>Convolutional Neural Network</i>)

<i>SVM</i>	maszyna wektorów nośnych (ang. <i>Support Vector Machine</i>)
<i>GPIO</i>	zestaw wyprowadzeń z portami wejścia/wyjścia ogólnego przeznaczenia (ang. <i>general-purpose input/output</i>)
<i>IR</i>	filtr podczerwieni (ang. <i>infra-red filter</i>)
<i>CCD</i>	matryca światłoczuła (ang. <i>charge-coupled device</i>)
<i>IR LED</i>	diody podczerwieni
<i>UV</i>	promieniowanie ultrafioletowe (ang. <i>ultraviolet</i>)
<i>TN</i>	wartości prawdziwie ujemne (ang. <i>True Negative</i>)
<i>FN</i>	wartości fałszywie ujemne(ang. <i>False Negative</i>)
<i>TP</i>	wartości prawdziwie dodatnie (ang. <i>True Positive</i>)
<i>FP</i>	wartości fałszywie dodatnie(ang. <i>False Positive</i>)
<i>ACC</i>	dokładność (ang. <i>accuracy</i>)
<i>PPV</i>	precyzja (ang. <i>positive predictive value</i>)
<i>TPR</i>	czułość (ang. <i>true positive rate</i>)
<i>TNR</i>	swoistość (ang. <i>true negative rate</i>)
<i>IoU</i>	przecięcie z unią (ang. <i>Intersection over Union</i>)
<i>MCC</i>	współczynnik korelacji Matthews'a, współczynnik ϕ (ang. <i>Matthews Correlation Coefficient</i>),
<i>YOLO</i>	grupa modeli uczenia maszynowego umożliwiająca detekcję obiektów, opierająca swoje działanie o jednostopniową analizę całego obrazu (ang. <i>You Only Look Once</i>)
<i>R-CNN</i>	konwolucyjna sieć neuronowa wykorzystująca analizę regionów (ang. <i>Regional Convolutional Neural Network</i>)
<i>SSD</i>	algorytm detekcji obrazów wykorzystujący sieci neuronowe, opierający swoje działanie o jednostopniową analizę całego obrazu (ang. <i>Single Shot Detector</i>)
<i>FPS</i>	Liczba klatek/s (ang. <i>frames per second</i>)

F	<i>analizowany film</i>
f_i	<i>kolejna, i-ta klatka analizowanego filmu F</i>
i	<i>indeks klatek w filmie F</i>
$ F $	<i>liczba klatek w filmie F</i>
g	<i>siatka obrazu (w procesie algorytmu adaptacyjnego wyznaczania okna zagęszczenia)</i>
s	<i>szerokość obrazu bazowego (w procesie algorytmu adaptacyjnego)</i>
w	<i>wysokość obrazu bazowego (w procesie algorytmu adaptacyjnego)</i>
c	<i>liczba kolumn siatki g</i>
r	<i>liczba wierszy siatki g</i>
h	<i>rozmiar siatki g</i>
$p(x,y)$	<i>współrzędne pszczoły p na obrazie bazowym</i>
(j,k)	<i>współrzędne pola siatki g</i>
m	<i>liczba klatek analizowanych w procesie wyznaczania okna zagęszczenia</i>
oz	<i>okno zagęszczenia</i>
t	<i>wartość progowa, określająca zakończenie procesu wyznaczania okna zagęszczenia</i>
a	<i>współczynnik regulacji wartości progowej t</i>
lp	<i>stała rozgraniczająca liczbę pszczoł na obrazie, dla której w procesie algorytmu adaptacyjnego wyznaczania okna zagęszczenia należy wykorzystać jedną bądź cztery siatki</i>



Spis rysunków

<i>Rys. 1 Pszczoła zaatakowana przez dręcza pszczelego</i>	13
<i>Rys. 2 Cztery rewolucje przemysłowe</i>	16
<i>Rys. 3 Zastosowanie systemów cyber-fizycznych</i>	23
<i>Rys. 4 Proces uczenia maszynowego</i>	25
<i>Rys. 5 Sztuczna inteligencja i jej dziedziny</i>	26
<i>Rys. 6 Przetwarzanie brzegowe</i>	28
<i>Rys. 7 Historia monitorowania pszczół</i>	33
<i>Rys. 8 Systemy pszczelarstwa precyzyjnego</i>	40
<i>Rys. 9. Urządzenie IoT oparte o komputer RaspberryPi z akceleratorem USB Google Coral ([109])</i>	51
<i>Rys. 10 Ogólna architektura systemu IoT wykorzystującego komputer RaspberryPi oraz chmurę AWS i Google Cloud Platform ([109])</i>	51
<i>Rys. 11 Urządzenie IoT działające w oparciu o komputer NVidia Jetson Nano w wersji 2GB [110].</i>	53
<i>Rys. 12 Ogólna architektura systemu IoT wykorzystującego komputer NVidia Jetson Nano w wersji 2GB oraz chmurę AWS [110]</i>	53
<i>Rys. 13 Urządzenie brzegowe oparte o komputer NVidia Jetson Nano w wersji 4GB</i>	56
<i>Rys. 14 Architektura systemu umożliwiającego wykrywanie sytuacji zagrażających życiu pszczół.</i>	58
<i>Rys. 15 Porównanie architektury Single Shot Detector i YOLO (źródło: [122])</i>	64
<i>Rys. 16 Schemat działania urządzenia IoT umożliwiającego wykrywanie dręcza pszczelego</i>	65
<i>Rys. 17 Przykładowa reprezentacja graficzna siatki g</i>	67
<i>Rys. 18. Przykładowe okno zagęszczenia (zaznaczone na czerwono) w odniesieniu do obrazu bazowego.</i>	69
<i>Rys. 19 Etapy właściwej detekcji pszczół na klatkach filmowych</i>	73
<i>Rys. 20 Przykładowy obraz z zaznaczonymi obiektami wykrytymi w procesie detekcji pszczół</i>	74
<i>Rys. 21 Przykładowe wyniki detekcji pszczół zwrócone przez model 4k05 na obrazach z większym (a) i mniejszym (b) zagęszczeniem obiektów.</i>	84
<i>Rys. 22 Miara dokładności modeli wykrywających obiekty pszczół</i>	85
<i>Rys. 23 Jakość klasyfikacji modeli wyrażona jako metryka precyzji</i>	85

<i>Rys. 24</i> Czułość modeli detekcji pszczoł	86
<i>Rys. 25</i> Miara F1 modeli detekcji pszczoł	87
<i>Rys. 26</i> Okno zagęszczenia obliczone dla filmu 3 dla następujących parametrów: $h=40$, $a=0.8$	92
<i>Rys. 27</i> Okno zagęszczenia obliczone dla filmu 4 dla następujących parametrów $h=40$, $a=0.7$	93
<i>Rys. 28</i> Okna zagęszczenia wyliczone za pomocą algorytmu adaptacyjnego dla filmu 1 z użyciem jednej (zdjęcie a.) i czterech (zdjęcie b.) siatek	94
<i>Rys. 29</i> Okno zagęszczenia zwrócone przez algorytm adaptacyjny dla filmu 2	95
<i>Rys. 30</i> Późniejsze klatki filmu	96
<i>Rys. 31</i> Okno zagęszczenia wyznaczone dla filmu 5.	96
<i>Rys. 32</i> Późniejsze klatki analizowanego filmu 5.	97
<i>Rys. 33</i> Kolejne etapy działania algorytmu adaptacyjnego wyznaczania okna zagęszczenia	98
<i>Rys. 34</i> Średnie czasy przetwarzania poszczególnych etapów algorytmu wyznaczającego okno zagęszczenia dla 1 siatki.	99
<i>Rys. 35</i> Średnie czasy przetwarzania poszczególnych etapów algorytmu wyznaczającego okno zagęszczenia dla 4 siatek.	100
<i>Rys. 36</i> Czasy przetwarzania filmu 1	102
<i>Rys. 37</i> Czasy przetwarzania filmu 2	103
<i>Rys. 38</i> Czasy przetwarzania dla filmu 3	104
<i>Rys. 39</i> Czasy przetwarzania dla filmu 4	105
<i>Rys. 40</i> Czasy przetwarzania filmu 5	106
<i>Rys. 41</i> Wykres czasu przetwarzania strumienia przy 20 pomijanych klatkach dla dwóch różnych algorytmów [110]	108

Spis tabel

Tab. 1 Porównanie urządzeń IoT i IIoT	20
Tab. 2 Zestawienie rozwiązań opisanych w literaturze.....	42
Tab. 3 Porównanie modeli sieci neuronowych SSD300, YOLO i Faster R-CNN dedykowanych detekcji obiektów na obrazach (na podstawie [122]).....	61
Tab. 4 Macierz pomyłek	76
Tab. 5 Macierze pomyłek dla wytrenowanych modeli	83
Tab. 6 Uśrednione wartości metryk jakości modeli	87
Tab. 7 Macierz pomyłek detekcji dręcza pszczelego.....	89
Tab. 8 Miary oceny jakości detekcji dręcza pszczelego.....	90
Tab. 9 Wyniki działania algorytmu adaptacyjnego dla rozmiaru siatki $h = 40$	91
Tab. 10 Wyniki działania algorytmu adaptacyjnego dla rozmiaru siatki $h = 80$	91
Tab. 11 Czasy detekcji pszczół na 1 klatce filmu	101
Tab. 12 Czasy klasyfikacji jednego obrazu z obiektem pszczoły pod kątem obecności dręcza pszczelego	101
Tab. 13 Zestawienie wyników badań przeprowadzonych na różnych systemach.....	108
Tab. 14 Porównanie średnich czasów detekcji pszczół i detekcji dręcza pszczelego na obrazie ...	109



Spis algorytmów

Algorytm 1 Działanie systemu umożliwiającego wykrywanie sytuacji zagrażających życiu pszczół	59
Algorytm 2 Przetwarzanie wstępne algorytmu adaptacyjnego wyznaczenia okna zagęszczenia	70
Algorytm 3 Działanie algorytmu adaptacyjnego wyznaczenia okna zagęszczenia	71
Algorytm 4 Detekcja dręcza pszczelego	74



Załączniki

Repozytorium zawierające kody źródłowe:

<https://github.com/AniaKowalczevska/RozprawaDoktorska>