



AKADEMIA GÓRNICZO-HUTNICZA IM. STANISŁAWA STASZICA W KRAKOWIE

DZIEDZINA NAUK INŻYNIERYJNO-TECHNICZNYCH
DYSCYPLINA INŻYNIERIA ŁĄDOWA, GEODEZJA I TRANSPORT

ROZPRAWA DOKTORSKA

Wykrywanie azbestowo-cementowych pokryć
dachowych metodami teledetekcyjnymi

Autor: mgr inż. JUDYTA KSIAŻEK

Promotor rozprawy: prof. dr hab. inż. Beata Hejmanowska
Promotor pomocniczy: dr inż. Ewa Głowienka

Praca wykonana:
Akademia Górniczo-Hutnicza w Krakowie,
Wydział Geodezji Górniczej i Inżynierii Środowiska
Katedra Fotogrametrii, Teledetekcji Środowiska i Inżynierii Przestrzennej

Kraków, 2023



AGH

AGH UNIVERSITY OF SCIENCE AND TECHNOLOGY

FIELD OF SCIENCE ENGINEERING AND TECHNOLOGY
SCIENTIFIC DISCIPLINE CIVIL ENGINEERING, GEODESY AND TRANSPORT

DOCTORAL THESIS

Detection of asbestos-cement roofs using remote
sensing methods

Author: mgr inż. JUDYTA KSIAŻEK

First supervisor: prof. dr hab. inż. Beata Hejmanowska

Assisting supervisor: dr inż. Ewa Głowienka

Completed in:

AGH UST

Faculty of Geo-Data Science, Geodesy and Environmental Engineering
Department of Photogrammetry, Remote Sensing, and Spatial Engineering

Krakow, 2023

Składam serdecznie podziękowania

Prof. dr hab. inż. Beacie Hejmanowskiej

oraz

dr inż. Ewie Głowience

za zaangażowanie, życzliwość i wszelką udzieloną
mi pomoc.

Dziękuję też ***Mojej Rodzinie.***

STRESZCZENIE

W prezentowanej pracy podjęto problem wykrywania azbestowo-cementowych pokryć dachowych na podstawie wielospektralnego obrazu satelitarnego z zastosowaniem podstawowych modeli metod uczenia maszynowego. Obecnie nie ma wdrożonych rozwiązań automatycznego wykrywania dachów azbestowych przy użyciu danych teledetekcyjnych.

W przygotowanej rozprawie krótko scharakteryzowano azbest oraz jego negatywny wpływ na zdrowie człowieka. Następnie dokonano przeglądu literatury pod względem metod oraz materiałów źródłowych wykorzystywanych do automatycznej identyfikacji azbestu. W związku z brakiem wskazania w literaturze jednoznacznej metody wykrywania azbestowych pokryć dachowych na danych teledetekcyjnych, przeprowadzono własne badania. Podczas prac skupiono się na wykorzystaniu obrazu satelitarnego WorldView-3. W czasie klasyfikacji próbek dachowych opierano się wyłącznie na informacjach spektralnych zapisanych na obrazie jako wartość jasności piksela. W badaniach korzystano z pięciu klasyfikatorów binarnych. W wyniku przeprowadzonych badań opracowano autorski filtr, którego zadaniem było wzmocnienie informacji spektralnej zapisanej na obrazie satelitarnym, tak aby podstawowe klasyfikatory uczenia maszynowego mogły efektywnie odróżniać pokrycie azbestowe od pozostałych. Manualny dobór parametrów filtra jest trudnym zadaniem. W celu efektywnego rozwiązania tego zagadnienia, wykorzystano sztuczną inteligencję poprzez zastosowanie algorytmu genetycznego.

Otrzymane wyniki oceny dokładności potwierdziły skuteczność zaproponowanej metody klasyfikacji na obrazie wielospektralnym, dla sformułowanego problemu klasyfikacji binarnej: dachy azbestowe – inne pokrycia dachowe. Najwyższe wartości wybranych metryk oceny dokładności osiągnięto dla klasyfikatora RFC (*ang. RandomForestClassifier*) oraz GBC (*ang. GradientBoostingClassifier*). Ogólna dokładność dla obu klasyfikatorów wynosiła 86%, natomiast wartość miary F1 dla klasy azbestowej otrzymano odpowiednio na poziomie 78% oraz 77%. Uzyskane wyniki znajdują odzwierciedlenie w literaturze, gdzie również potwierdzono skuteczność prostych klasyfikatorów podczas wykrywania pokryć azbestowych na danych teledetekcyjnych. Nowością w przedstawionej metodzie klasyfikacji jest filtrowanie danych wg macierzy filtrującej wskazanej przez algorytm genetyczny.

ABSTRACT

The problem of detecting asbestos-cement roofs based on multispectral satellite imagery using basic machine learning methods has been analyzed in the presented dissertation. At present, there are no implemented solutions available for the automatic detection of asbestos roofs using remote sensing data.

In the dissertation, a brief characterization of asbestos and its negative impact on human health was provided. Subsequently, a literature review regarding the methods and source materials used for the automatic identification of asbestos was conducted. Due to the lack of a clear method for detecting asbestos roofs using remote sensing data in the literature, own research was performed. The presented work focused on using of the WorldView-3 satellite image. The classification of roof samples was based only on the spectral information stored on the image as a pixel brightness value. Five binary classifiers were used in the research. As a result of the conducted analyses, an innovative and original filter was developed. The task of this filter was to strengthen the spectral information from the image so that basic machine learning classifiers can effectively distinguish asbestos roofs from other ones. Manual selection of filter parameters is a difficult task. In order to effectively solve this problem, artificial intelligence was used through using of a genetic algorithm.

The obtained results of accuracy assessment confirmed the effectiveness of the proposed method of classification of asbestos roofs on a multispectral image, during the binary classification: asbestos roofs - other roofing materials. The highest values of selected accuracy metrics for the RFC (RandomForestClassifier) and GBC (GradientBoostingClassifier) classifiers were achieved. The overall accuracy for both classifiers was 86%, while the F1 score for the asbestos class was obtained at 78% and 77% respectively. The achieved outcomes are reflected in the literature, where the effectiveness of simple classifiers in detecting asbestos roofs using remote sensing data has also been confirmed. A novelty in the presented classification method is data filtering based on a filtering matrix determined by a genetic algorithm.

SPIS TREŚCI

STRESZCZENIE	6
ABSTRACT	7
1 Wprowadzenie	10
2 Charakterystyka azbestu.....	12
2.1 Azbest.....	12
2.2 Wpływ azbestu na zdrowie człowieka.....	13
2.3 Azbest w Polsce	15
3 Przegląd literatury.....	17
3.1 Zakres i podział prac badawczych.....	17
3.2 Badania laboratoryjne próbek dachów azbestowych.....	19
3.3 Rodzaje danych źródłowych	20
3.3.1 Obrazy hiperspektralne	21
3.3.2 Obrazy wielospektralne.....	23
3.4 Metody klasyfikacji.....	25
3.4.1 Klasyfikacja pikselowa	26
3.4.2 Klasyfikacja obiektowa.....	28
3.4.3 Sztuczna inteligencja w wykrywaniu dachów azbestowych	29
3.4.4 Klasyfikatory binarne.....	30
3.5 Algorytmy wspomagające proces klasyfikacji.....	34
3.5.1 Algorytmy genetyczne	35
3.6 Ocena przydatności klasyfikatorów.....	41
3.7 Nowe kierunki badań.....	44
4 Obszar badań	46
5 Dane	47
5.1 Dane in situ	47
5.2 Dane satelitarne	50
6 Metodyka prac badawczych	52
6.1 Przygotowanie zestawu danych trenujących, walidacyjnych oraz testowych.....	52
6.2 Wstępne przetworzenie danych.....	54
6.3 Algorytm filtrowania kanałów spektralnych.....	57
6.3.1 Przykład funkcjonalności opracowanego w pracy algorytmu filtrowania kanałów spektralnych.....	57
6.4 Analiza statystyczna klasyfikowanych danych	61

6.5	Przebieg eksperymentów obliczeniowych	68
6.5.1	Strategia I	70
6.5.2	Strategia II	73
6.5.3	Strategia III	73
6.5.4	Testowanie klasyfikatorów na przefiltrowanych danych	74
6.6	Środowisko obliczeniowe	74
7	Wyniki klasyfikacji	76
7.1	Wyznaczenie filtrów – strategia I	76
7.2	Ocena skuteczności filtrów – strategia I	83
7.3	Wyznaczenie filtrów – strategia II	85
7.4	Klasyfikacja w programie SAGA – strategia III	92
7.5	Wyniki oceny klasyfikacji na danych testowych	95
8	Analiza i dyskusja wyników	99
8.1	Porównanie wyników eksperymentów obliczeniowych	99
8.2	Porównanie miar oceny dokładności na danych walidacyjnych oraz testowych	103
8.3	Porównanie otrzymanych wyników z literaturą przedmiotu	104
9	Podsumowanie i wnioski	107
10	Literatura	109
	SPIS TABEL	115
	SPIS RYSUNKÓW	117
	ZAŁĄCZNIK 1	120
	ZAŁĄCZNIK 2	124
	ZAŁĄCZNIK 3	127

1 WPROWADZENIE

Podjęty w pracy problem badawczy dotyczy próby zastosowania podstawowych metod uczenia maszynowego dla zadania klasyfikacji dachów azbestowych na podstawie obrazów satelitarnych. Obecnie nie ma wdrożonych rozwiązań automatycznego wykrywania dachów azbestowych przy użyciu danych teledetekcyjnych. Identyfikacja dachów azbestowych jest formułowana jako pewne złożone zadanie, które często wymaga analizy wielu czynników, w tym kształtu, tekstury, koloru i spektralnych właściwości azbestu.

Istnieje wiele badań i prac naukowych poświęconych wykrywaniu dachów azbestowych przy użyciu danych teledetekcyjnych. W badaniach tych wykorzystuje się różne techniki, takie jak analiza obrazów i uczenie maszynowe (ML, *ang. Machine Learning*), aby opracować automatyczne algorytmy umożliwiające identyfikację dachów azbestowych z wykorzystaniem dostępnych danych teledetekcyjnych.

Rozwój prowadzonych badań w dziedzinie wykrywania dachów azbestowych aktualnie skupia się na wykorzystaniu danych dla dużych obszarów terenu z wysoką rozdzielczością przestrzenną i stosunkowo niskim kosztem pozyskania danych. Obecnie w procesie klasyfikacji nie wykorzystuje się dodatkowych danych wspomagających, tak jak miało to miejsce we wcześniejszych pracach badawczych, np. numerycznego modelu terenu, warstw wektorowych budynków. Natomiast w wyniku ograniczenia różnorodności danych wejściowych zmieniono wykorzystywane algorytmy klasyfikacyjne na bardziej złożone.

W prezentowanej pracy skupiono się na wykorzystaniu obrazu satelitarnego WorldView-3, zgodnie z tendencją rozwoju badań w tym przedmiocie polegającą na formułowaniu zadań opartych na pojedynczym źródle danych wejściowych. Podczas badań analizie poddawano informacje spektralne zapisane na obrazie jako wartość jasności piksela. Głównym celem niniejszej rozprawy było sprawdzenie możliwości wykorzystanie znanych metod uczenia maszynowego w procesie identyfikacji dachów azbestowych z wykorzystaniem jednego źródła danych spektralnych oraz dokonanie oceny ich przydatności w rozwiązaniu analizowanego problemu.

W związku z powyższym, można postawić tezę, że podstawowe klasyfikatory uczenia maszynowego są skutecznym narzędziem w procesie automatycznej identyfikacji dachów azbestowo cementowych na satelitarnych obrazach wielospektralnych.

W celu weryfikacji hipotezy postawiono cztery szczegółowe cele badawcze:

1. Zbadanie przydatności filtrowania danych spektralnych zaproponowanym sposobem podczas klasyfikacji obrazów wielospektralnych.
2. Zbadanie wpływu zarejestrowanego na zdjęciu otoczenia budynków na wynik klasyfikacji.
3. Zbadanie działania losowej oraz stratyfikowanej metody walidacji krzyżowej dla niezrównoważonych zbiorów danych wejściowych.
4. Zbadanie skuteczności i przydatności zastosowania algorytmu genetycznego podczas doboru macierzy filtrującej dane spektralne.

2 CHARAKTERYSTYKA AZBESTU

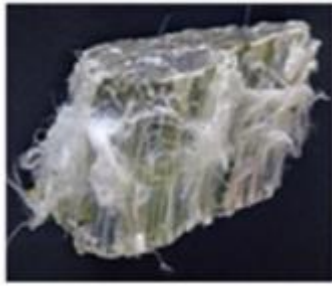
W rozdziale opisano właściwości azbestu, wyjaśniając powszechność stosowania go w budownictwie i przemyśle w drugiej połowie ubiegłego wieku. Następnie podano najważniejsze informacje o wpływie tego minerału na zdrowie człowieka oraz konsekwencjach długotrwałego kontaktu z nim. Wskazano również podstawy prawne regulujące wykorzystanie i utylizację materiałów zawierających azbest. Na końcu przedstawiono zakres problemu na terenie Rzeczypospolitej Polski.

2.1 AZBEST

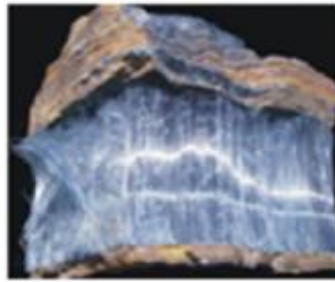
Azbestem określamy grupę naturalnie występujących minerałów krzemianowych o strukturze włóknistej. Wyróżniamy sześć typów azbestu ze względu na budowę włókien: serpentynowe o włóknach kręconych (chryzotyl) oraz amfibolowe o włóknach przypominających igły (aktynolit, amozyt, antofyllit, krokidolit, tremolit (Rys. 1). W stosunku do innych krzemianów, włókna azbestowe posiadają wyjątkowo małe średnice od mikro- do nanometrów (Militello *i in.*, 2021).

Właściwości fizyczno-chemiczne azbestu sprawiają, że był i jest nadal, przy zastosowaniu odpowiednich środków zabezpieczających, jednym z ważniejszych minerałów nieorganicznych powszechnie wykorzystywanych do celów przemysłowych, budowlanych i innych zastosowań technologicznych. W Polsce w latach 60. i 70. azbest był powszechnie dodawany jako zbrojenie rozproszone do cementu wykorzystywanego do produkcji płyt dachowych płaskich prasowanych, typu KARO czy falistych (eternit), rur otulinowo – izolacyjnych czy okładzin elewacyjnych. Można go także znaleźć w betonie, w ścianach budynków z tamtych lat, gdzie jego zawartość sięgała nawet kilkunastu procent.

Ponad 95% azbestu wykorzystywanego w przemyśle to chryzotyl. Minerały z grupy azbestów amfibolowych nie znalazły szerszego zastosowania z powodu kruchości ich cienkich włókien. Do najważniejszych zalet azbestu zalicza się jego odporność na wysokie temperatury, niepalność, elastyczność oraz wytrzymałość mechaniczną. Jest też odporny na kwasy i inne substancje żrące, trwały w szerokim zakresie pH w matrycy cementowej oraz charakteryzuje się niskim przewodnictwem elektrycznym (Virta, 2002).



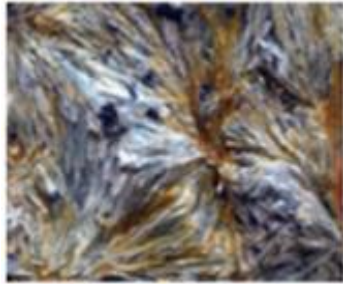
CHRYZOTYL



AKTYNOLIT



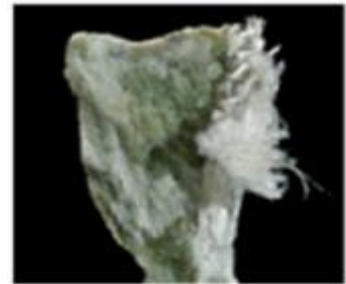
AMOZYT



ANTOFYLLIT



KROKIDOLIT



TREMOLIT

Rys. 1 Typy azbestu wyróżnione ze względu na budowę włókien
(Brandi i Tavorari, 2020, zmodyfikowano)

2.2 WPLYW AZBESTU NA ZDROWIE CZŁOWIEKA

Według Międzynarodowej Agencji Badań nad Rakiem, IARC (*ang. International Agency for Research on Cancer*) wdychanie drobnych włókien azbestowych oraz ich połykanie to główne drogi narażenia człowieka na azbest. Kontakt ze skórą nie jest niebezpieczny, chociaż może prowadzić do wtórego narażenia na włókna poprzez połknięcie lub wdychanie. Stopień zdolności do przenikania włókien azbestowych do płuc zależy głównie od ich średnicy, przy czym cienkie włókna mają największy potencjał do głębokiego penetracji tkanek w płucach i opłucnej (IARC, 2012).

Obecność włókien azbestowych w płucach wyzwała różne rodzaje odpowiedzi prowadzące do wywołania stanu zapalnego, uszkodzenia tkanek i komórek, co może skutkować w następstwie rozwojem złośliwych nowotworów i/lub przewlekłych chorób nienowotworowych układu oddechowego (Zephaniah i Aremu, 2019). Toksyczne skutki wdychania pyłu azbestowego zależą od kilku czynników, w tym skumulowanej dawki, czasu początkowej ekspozycji oraz właściwości fizycznych i chemicznych wspólnych dla różnych włókien azbestu (Kamp, 2009). Do chorób, co do których badania epidemiologiczne

potwierdziły związek z narażeniem na ekspozycje azbestem, należą: szereg postaci raka płuc, międzybłoniak opłucnej czy pylica azbestowa (inaczej azbestoza) (Zephaniah i Aremu, 2019).

Początki wykorzystania azbestu przez człowieka sięgają końca XIX wieku, z najwyższym zużyciem przypadającym na lata 60. i 70. XX wieku. Pierwsze obserwacje i prace wykazujące patogeny wpływ azbestu na organizm człowieka pojawiły się już na początku XX wieku, ale dopiero w latach 60. XX wieku pojawiły się duża liczba publikacji z całego świata, ukazująca zagrożenia dla zdrowia ze względu na ekspozycje na włókna azbestowe (Bolan i in, 2023).

Należy pamiętać, że dopiero naruszony azbest uwalnia włókna do atmosfery i powoduje zagrożenie. Azbest osadzony lub zawarty w nienaruszonych minerałach naturalnych lub stabilnych materiałach wytworzonych przez człowieka stwarza znikome ryzyko narażenia dla zdrowia ludzkiego. Emisja pyłów może następować na skutek uszkodzeń mechanicznych wynikających z działalności człowieka lub z powodów trwania rozkładu materiału np. emitują go stare płyty pokryć dachowych azbestowo-cementowych o naruszonej przez czynniki atmosferyczne lub chemiczne powierzchni zewnętrznej.

W raporcie Światowej Organizacji Zdrowia WHO (*ang. World Health Organization*) z 2014 roku stwierdzono, że obciążenie chorobami związanymi z azbestem wciąż rośnie, nawet w krajach, które zakazały stosowania azbestu już na początku lat 90. XX wieku. Związane jest to z długim okresem utajenia: skutki wdychania azbestu mogą dać o sobie znać dopiero po upływie 20 - 30 lat po narażeniu (WHO, 2014).

Zgodnie z dyrektywą 1999/77/WE, w całej Unii Europejskiej od 1 stycznia 2005 roku obowiązuje całkowity zakaz stosowania azbestu, ponadto do 2032 roku nałożony został obowiązek unieszkodliwiania wszystkich materiałów azbestowych na terenach Unii Europejskiej.

W kolejnych dyrektywach unijnych:

- a) 2003/18/WE - zakazuje się wydobycia azbestu oraz wytwarzania i przetwarzania produktów pochodzenia azbestowego.
- b) 2009/148/WE - reguluje się sprawy związane z ochroną pracowników przed ryzykiem związanym z narażeniem na działanie azbestu w miejscu pracy.

2.3 AZBEST W POLSCE

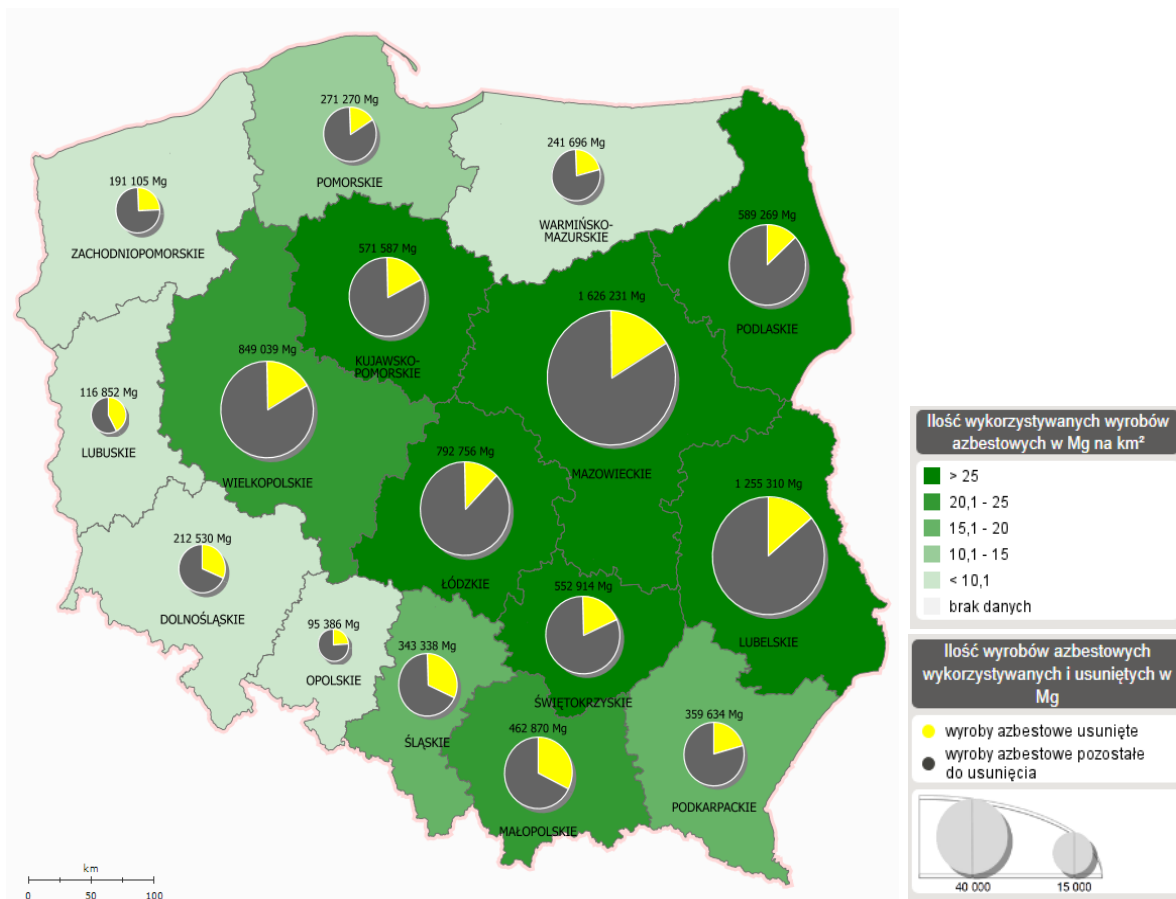
Zgodnie z obowiązującą w Polsce ustawą (ustawa z dnia 19 czerwca 1997 r. o zakazie stosowania wyrobów zawierających azbest. Dz.U. 1997, nr 101, poz. 628) już w 1997 został zakazany import, produkcja i handel azbestem i materiałami zawierającymi azbest. Uchwalone prawo rozwiązało problem narażenia zawodowego współczesnych mu pracowników sektorów przemysłowego i budowlanego, nie zmieniło jednak faktu obecności dużej ilości azbestu, który pozostał w naszym otoczeniu do dzisiaj. Zgodnie z zaleceniami wspomnianej już dyrektywy Unii Europejskiej 1999/77/WE, polska zobowiązana jest do bezpiecznej utylizacji materiałów zawierających azbest na swoim terytorium. W dniu 15 marca 2010 r. Rada Ministrów podjęła uchwałę w sprawie przyjęcia *Programu Oczyszczania Polski z Azbestu na lata 2009-2032* (POKA, 2010). Główne cele wprowadzonego programu to:

- a) Usuwanie i unieszkodliwianie wyrobów zawierających azbest.
- b) Minimalizacja negatywnych skutków zdrowotnych spowodowanych obecnością azbestu na terenie kraju.
- c) Likwidacja szkodliwego oddziaływania azbestu na środowisko.

Wymienione powyżej cele monitorowane są m. in. poprzez utworzenie i prowadzenie Bazy Azbestowej (<https://bazaazbestowa.gov.pl/pl/>) tj. narzędzia do gromadzenia i przetwarzania informacji na temat stanu materiałów zawierających azbest uzyskanych z poszczególnych jednostek terytorium samorządowego. Na podstawie Bazy Azbestowej szacuje się, że w Polsce nadal pozostaje blisko 7 mln ton azbestu wymagających unieszkodliwienia (6 964 701 249 kg, szacunek na maj 2023 roku). Ponad połowę wykorzystanego w przeszłości azbestu użyto w budownictwie indywidualnym, w tym w znacznej części na pokrycia dachowe, które nadal czekają na bezpieczną utylizację. Ilość wyrobów zawierających azbest: zinwentaryzowanych oraz usuniętych, z podziałem na województwa w Polsce, przedstawiano na rysunku (Rys. 2). Zgodnie z Bazą Azbestową najwięcej azbestu do utylizacji znajduje się w środkowo – wschodniej Polsce, szczególnie w województwach: mazowieckim i lubelskim.

W związku z chorobotwórczymi właściwościami azbestu oraz ogromną ilością niezutylizowanego materiału zawierającego azbest, wykrywanie azbestowych poryć dachowych jest ważnym i pilnym problemem. Przy czym warto zwrócić uwagę,

że inwentaryzacja terenowa jest możliwa do wykonania, ale wiąże się z dużym nakładem pracy i czasu (Krówczyńska i Wilk, 2013).



Rys. 2 Ilość wyrobów zawierających azbest: zinventaryzowanych oraz usuniętych z podziałem na województwa w Polsce. (Mg = tona).
 (https://esip.bazaazbestowa.gov.pl/analizy.html, dostęp: maj 2023)

3 PRZEGLĄD LITERATURY

W rozdziale przedstawiono zakres opisanych w literaturze badań dotyczących wykrywania dachów azbestowych z wykorzystaniem danych teledetekcyjnych. Opisano najważniejsze efekty prac laboratoryjnych. Pogrupowano przeprowadzone badania pod względem danych źródłowych oraz metod klasyfikacji. Scharakteryzowano klasyfikatory oraz algorytm genetyczny używane w trakcie badań. Przytoczono stosowane metody wspomagające selekcję danych oraz proces klasyfikacji. Wyjaśniono sens wybranych miar oceny dokładności klasyfikacji. Na końcu wskazano możliwe kierunki rozwoju badań w tym temacie.

3.1 ZAKRES I PODZIAŁ PRAC BADAWCZYCH

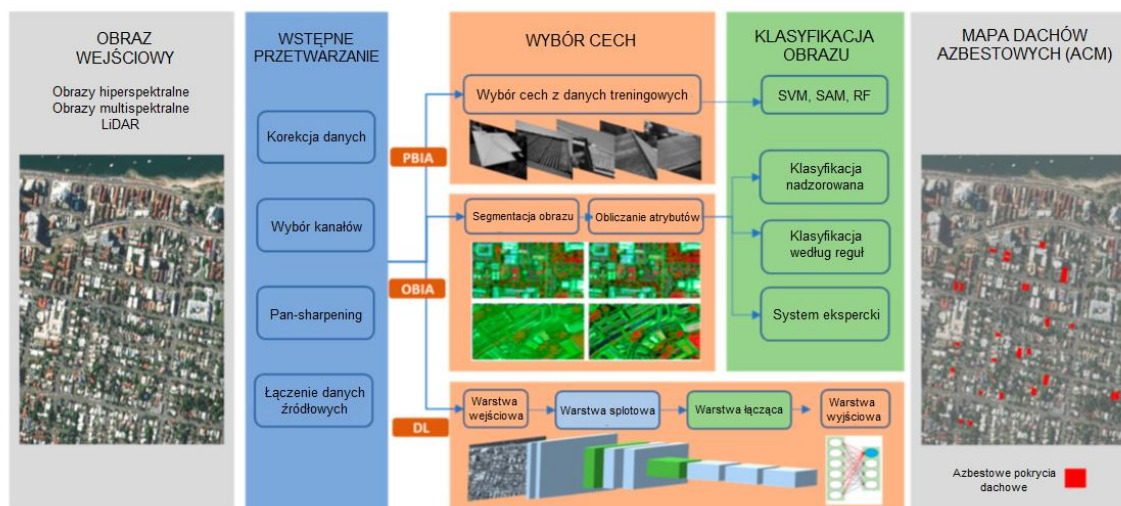
Automatyczne wykrywanie budynków z wykorzystaniem danych teledetekcyjnych¹ jest szeroko omawiane w literaturze naukowej (Ghanea i in., 2016; Luo i in., 2021; Tomljenovic i in., 2015). W zagadnieniu automatycznej identyfikacji budynków na obrazach można wyróżnić dwa podejścia do problemu. Pierwszym z nich jest wyłącznie zlokalizowanie budynku jako odrębnego obiektu. W drugim przypadku istotne jest sklasyfikowanie materiałów pokryć dachowych. W dostępnej literaturze dodatkowo można wskazać różne cele, dla których prowadzono badania, są to m. in. inwentaryzacja budynków przy ocenie szkód spowodowanych trzęsieniem ziemi (Vu i in., 2009), identyfikacja materiałów niebezpiecznych, takich jak azbest (Abbasi i in., 2022; Bassani i in., 2007; Raczek i in., 2022), czy też badania pozwalające określić efektywność energetyczną dachów pokrywających budynki (Abdulkarim i in., 2014). W związku z obszernym zakresem piśmiennictwa w tym temacie, w dalszej części zostanie przedstawiony aktualny stan wiedzy związany z tematem wykrywania azbestowo - cementowych pokryć dachowych.

Literaturę opisującą problem identyfikacji dachów azbestowych można podzielić na trzy zasadnicze grupy. W pierwszej celem badań była analiza laboratoryjna krzywych odbicia spektralnego azbestowych pokryć dachowych. Druga grupa zawierała próby

¹ Teledetekcja jest to nauka o pozyskiwaniu informacji o obiektach na podstawie pomiarów wykonanych z dala od obiektów, bez bezpośredniego kontaktu z nimi (Davis i in., 1978).

wykrywania szerokiego zakresu materiałów pokryć dachowych. Podczas klasyfikacji wyróżniano kilka – kilkanaście klas, z czego co najmniej jedna klasa definiowana była jako azbest. Natomiast ostatnia grupa obejmowała badania nad możliwością wydzielenia wyłącznie azbestowych pokryć dachowych z pozostałych rodzajów pokrycia (azbest – nie azbest).

Według Abbasi i in. (2022) badania prowadzące do klasyfikacji pokrycia azbestowego można podzielić na trzy etapy: przetwarzanie wstępne, wydobycie cech oraz klasyfikacja obrazu (Rys. 3). Przetwarzanie wstępne sprowadza się głównie do przekształcania i łączenia obrazów wejściowych, m. in. jest to korekcja obrazów wejściowych, czy pan-sharpening². Wydobycie cech polega na zebraniu zestawów danych, wynikających z informacji zarejestrowanej na obrazach. Mogą to być bezpośrednio odczytane wartości jasności pikseli, ale również zaliczają się do tego dane uzyskane po przekształceniach obrazu np. wartości średnie jasności, czy też cechy teksturalne. Etap ten dostarcza istotnych informacji dla procesu klasyfikacji, gdyż powoduje zmniejszenie wymiarowości danych teledetekcyjnych. Ostatnim krokiem jest wykonanie klasyfikacji, gdzie pojedyncze piksele lub wyodrębnione segmenty przypisywane są do odpowiednich pokryć dachowych. Klasyfikacja odbywa się zgodnie z wybranym modelem



Rys. 3 Ogólna procedura badań na temat wykrywania dachów azbestowych (Abbasi i in., 2022, zmodyfikowano)

² Pansharpening to technika przetwarzania obrazów, która polega na łączeniu danych wielospektralnych o niższej rozdzielczości przestrzennej z obrazem panchromatycznym o wyższej rozdzielczości przestrzennej. Efektem przetwarzania jest wyostrenie obrazu wielospektralnego (Smadi i in., 2022).

obliczeniowym, który może zostać wygenerowany na podstawie danych uczących dostarczonych na początku procesu klasyfikacji lub na podstawie systemu opartego na specjalistycznej wiedzy eksperckiej. Podczas opisanych w literaturze obliczeń (Rys. 3) wykorzystywano trzy metody klasyfikacji: metodę pikselową (PBIA, *ang. Pixel-Based Image Analysis*, patrz rozdział 3.4.1), metodę obiektową (OBIA, *ang. Object-Based Image Analysis*, patrz rozdział 3.4.2) oraz uczenie głębokie (DL, *ang. Deep Learning*, patrz rozdział 3.4.3).

Opisane badania w literaturze przedmiotu, można porównywać pod względem wielu kryteriów. Najistotniejsze różnice polegają na doborze teledetekcyjnych danych źródłowych wykorzystanych do identyfikacji pokryć dachowych lub też w sposobie przeprowadzenia klasyfikacji. Z uwagi na różnorodność danych wejściowych, metod ich przetwarzania i klasyfikacji stosowanych przez różnych autorów, jak i zróżnicowane parametry analizy dokładności, bezpośrednie porównywanie wszystkich wyników oceny dokładności nie jest odpowiednie.

3.2 BADANIA LABORATORYJNE PRÓBEK DACHÓW AZBESTOWYCH

W dostępnej literaturze znajduje się niewiele publikacji, opisujących problem szczegółowych analiz krzywych spektralnych pokryć dachowych, w celu określenia możliwości wykorzystania tych informacji w procesie klasyfikacji obrazów. W pracy Bassani i in. (2007) podjęto próbę określenia możliwości rozróżnienia stanu degradacji płyt azbestowo-cementowych oraz wyznaczenia długości fali elektromagnetycznej, dla której różnice odpowiedzi spektralnych pomiędzy azbestem i innymi materiałami są największe. Wyniki przeprowadzonych analiz wskazały na potencjał wykorzystania dwóch zakresów spektralnych: krótkofalowej podczerwieni (SWIR) o długości fali 2,32 μm oraz podczerwieni termalnej (TIR) o długości fali 9,44 μm . Z kolei w badaniach Krówczyńska i in. (2017), wyodrębniono dwanaście długości fal: 410, 550, 670, 740, 870, 990, 1310, 1700, 1840, 2130, 2200, 2270nm, dla których zaobserwowano lokalne piki i maksymalne różnice względem pozostałych pokryć dachowych. Warto zwrócić uwagę, że krzywe spektralne opracowane w trakcie badań były reprezentatywne dla nowych próbek pokryć dachowych. Pod wpływem warunków atmosferycznych wraz z upływem czasu, materiał pokrycia dachowego ulega degradacji: zmianom koloru, spękanom, porastaniem mchem. W związku

z tym wartości dla materiałów narażonych na długotrwałe czynniki zewnętrzne mogą różnić się od przedstawionych wyników.

Podczas analizowania krzywych spektralnych należy zwrócić uwagę na podobieństwo kształtu krzywej spektralnej pokryć azbestowych oraz dachówek cementowych. Jest to spowodowane tym, że ponad 80% składu pokrycia azbestowego stanowi materiał wiążący, którym jest właśnie cement (Krówczyńska i in., 2015).

Na podstawie prowadzonych badań opracowano biblioteki spektralne materiałów budowlanych, w tym również azbestu (Ilehag i in., 2019; Kotthaus i in., 2014). Mimo dostępności do bibliotek spektralnych, warto zaznaczyć, że w większości przypadków podejście do klasyfikacji obrazów na obszarach zurbanizowanych nie zmieniło się. Nadal preferowane są klasyfikacje oparte na próbkach szkoleniowych lub regułach decyzyjnych, zwłaszcza dla danych wielospektralnych (Gibril i in., 2017; Huang i in., 2019).

3.3 RODZAJE DANYCH ŹRÓDŁOWYCH

W celu wykrywania dachów azbestowych za pomocą technik teledetekcyjnych można wykorzystać różnorodne zestawy danych. Najczęściej danymi źródłowymi są obrazy satelitarne lub zdjęcia lotnicze, które różnią się rozdzielczością przestrzenną, spektralną i radiometryczną. Pierwsze prace badawcze wykonywano na obrazach hiperspektralnych³ pozyskanych z pułapu lotniczego (Marino i in., 2001; Bassani i in., 2007). W latach 2008-2015 badacze korzystali zarówno z lotniczego skanowania hiperspektralnego jak i wielospektralnych obrazów satelitarnych. W najnowszych opracowaniach omawianego tematu, począwszy od 2017 roku, analizowano obrazy satelitarne oraz wysokorozdzielcze zdjęcia lotnicze, wspomagane dodatkowym kanałem podczerwonym lub kanałem termalnym.

W tabeli (Tab. 1) zestawiono rodzaj wykorzystywanego obrazu źródłowego do wykrywania dachów azbestowych na przestrzeni lat. Obserwuje się wyraźny trend w kierunku badania obrazów stosunkowo tanich i szeroko dostępnych, jak ortofotomapy czy obrazy satelitarne.

³ Obrazy hiperspektralne to obrazy pozyskane w setkach spektralnie ciągłych kanałów, dla których możliwe jest określenie widma promieniowania każdego piksela (Goetz i in., 1985).

W celu wspomagania procesu identyfikacji dachów azbestowych, korzystano z dodatkowych danych m. in. numerycznego modelu terenu (Hamedianfar i in., 2014; Norman i in., 2020; Osińska-Skotak i Ostrowski, 2015). Informacje przestrzenne pozwalają na wykluczenie z obszaru zainteresowania obiektów innych niż budynki, natomiast informacje spektralne z obrazów służą do klasyfikacji materiału pokrycia dachowego (Huang i in., 2019).

Tab. 1 Rodzaj obrazu źródłowego wykorzystanego w identyfikacji dachów azbestowych na przestrzeni lat (opracowanie własne)

Rok	Obrazy hiperspektralne		Obrazy wielospektralne				Pulap lotniczy obraz RGB
	Pulap lotniczy		Pulap satelitarny				
	MIVIS	AISA	QuickBird	WV2	WV3	IKONOS	
2001	Marino i in.						
⋮							
2007	Bassani i in.		Pinho i in.				
2008							
2009	Cavalli i in.						
⋮							
2012	Fiumi i in. Frassy i in.					Pinho i in.	
2013				Taherzadeh i S.			
2014	Fiumi i in. Frassy i in.	Szabo i in.		Hamedianfar i in. Taherzadeh i in.	Osińska-Skotak		
2015	Cilia i in.			Hamedianfar i in. Osińska-Skotak i O.			
⋮							
2017				Gibril i in.			
2018				Abriha i in.	Despini i in.		
2019					Tommasini i in.		
2020					Norman i in.		Krówczyńska i in. (+ CIR)
⋮							
2022							Raczkowski i in.
2023					Hikuwai i in.		Hikuwai I in. Kaplan i in. (+ termalny)

3.3.1 Obrazy hiperspektralne

Obrazy hiperspektralne wykonywane z pułapu lotniczego charakteryzują się kilkumetrową rozdzielczością przestrzenną przy równocześnie bardzo wysokiej rozdzielczości spektralnej. Dlatego też wiele badań skupiało się na wykorzystaniu danych hiperspektralnych w celu zbadania efektywności samego obrazowania hiperspektralnego lub porównania dokładności klasyfikacji osiągniętej przez różne klasyfikatory (Tab. 2).

Wyniki badań Bassani i in. (2007) prowadzone na terenie Włoch wskazują, że możliwe jest wykorzystanie danych MIVIS⁴ oraz klasyfikacji Spectral Feature Fitting (SFF)⁵ do mapowania wyrobów azbestowo-cementowych, z ogólną dokładnością klasyfikacji na poziomie 80-90%.

Tab. 2 Zestawienie badań na temat wykrywania dachów azbestowych z wykorzystaniem obrazów hiperspektralnych (Abbasi i in., 2022, zmodyfikowano)

Sensor hiperspektralny	Liczba kanałów spektralnych	Zakres fali [nm]	Rozdzielczość przestrzenna [m]	Badanie
MIVIS	102	433 - 1270	3	Marino i in., 2001 Bassani i in, 2007
			3-4	Cavalli i in, 2009
			4	Fiumi i in, 2012
			4 - 9	Frassy i in., 2014
			3	Cilia i in., 2015 Fiumi i in, 2014
AISA Eagle II	126	400 -1000	1	Szabo i in., 2014

W początkowych badaniach przeprowadzanych na obrazach hiperspektralnych, pomimo wykorzystania szerokiego spektrum fal elektromagnetycznych, zauważano znaczne błędy pominięcia małych powierzchni zawierających azbest. Rozdzielczość przestrzenna zdjęć nie była wystarczająca, aby zidentyfikować niewielkie obiekty. W celu poprawy otrzymywanych wyników, w badaniu Frassy i in. (2014) zaproponowano, aby pojedynczy dach identyfikować przy użyciu okna 3x3 piksele, przy rozdzielczości przestrzennej 4m. Przy takim założeniu, ogólna dokładność klasyfikacji wzrosła z 43% (dla klasyfikacji pojedynczego piksela) do 75%, a błąd pominięcia (OE, *ang. omission error*) zmniejszył się z 48% do 19% dla całego zbioru danych. Natomiast w przypadku klasyfikacji jedynie dużych pokryć dachowych (średnia powierzchnia dachów: 1200 m²), uzyskiwano zadowalające wyniki dokładności: 89% średniej dokładności klasyfikacji (Fiumi i in, 2012).

W prowadzonych badaniach podjęto problem odbicia światła od nieregularnych kształtów dachów. Całkowita dokładność klasyfikacji, przeprowadzonej za pomocą

⁴ MIVIS (*ang. Multispectral Infrared and Visible Imaging Spectrometer*) – lotniczy skaner hiperspektralny w zakresie fali widzialnej i podczerwieni.

⁵ Metoda dopasowania krzywych spektralnych polega na analizie porównawczej krzywej spektralnej referencyjnej i krzywej spektralnej z obrazu metodą najmniejszych kwadratów (Clark i in., 1990).

algorytmu SAM⁶, bez uwzględniania kształtu dachów, wyniosła 65% (Fiumi i in., 2014). Po przeprowadzeniu badań terenowych i uwzględnieniu w analizach stopnia skośności dachu, weryfikacja danych pozwoliła na osiągnięcie całkowitej dokładności na poziomie 75%. Na to zagadnienie zwrócono również uwagę w pracy Szabo i in. (2014), gdzie oryginalny zbiór danych poddawano transformacji w celu usunięcia szumów spowodowanych różnym kątem odbicia promieni świetlnych. Klasyfikacja algorytmem SVM⁷ osiągnęła ogólną dokładność na poziomie 80%.

Wraz ze wzrostem rozdzielczości przestrzennej otrzymywano wyższe wyniki klasyfikacji (Hamedianfar i in., 2014; Cilia i in., 2015). Jednak główną przeszkodę w powszechnym wykrywaniu pokryć azbestowo-cementowych na podstawie danych hiperspektralnych stanowią koszty pozyskania danych (Krówczyńska i Wilk, 2013). Dodatkowo sensory hiperspektralne dostarczają ogromną ilość danych, które powinno się zredukować, w celu ich efektywnego wykorzystania. Istotny jest więc wybór odpowiednich kanałów spektralnych, długości fali oraz rozdzielczości przestrzennej do procesu klasyfikacji (Hamedianfar i in., 2014).

3.3.2 Obrazy wielospektralne

Obrazy wielospektralne składają się z dwóch do kilkunastu kanałów spektralnych. Koszty uzyskania tego rodzaju danych są znacznie mniejsze w porównaniu do obrazów hiperspektralnych (Gibril i in., 2017). Obrazy pozyskane z pułapu satelitarnego charakteryzują się szerokim pokryciem terenu w obrębie jednej sceny (Tommasini i in., 2019). Dodatkowo, wraz ze zmniejszeniem liczby kanałów spektralnych, dochodzi nie tylko do redukcji wymiarowości danych, ale także do ograniczenia ich objętości.

Rozdzielczość przestrzenna obrazu wykorzystywanego w klasyfikacji terenów zurbanizowanych nie powinna być niższa niż 5m (Welch, 1982; Small, 2003). Wykorzystywane obrazy satelitarne w omawianych badaniach nie przekraczały tej wartości (Tab. 3). Można zaobserwować ogólny trend pokazujący zmianę wykorzystywanych

⁶ Spectral Angle Mapper (SAM) to automatyczna metoda polegająca na porównaniu wektorów spektralnych każdego piksela na analizowanym obrazie z wektorami spektralnymi odpowiadającymi obiektom czystym spektralnie, reprezentującym klasy wynikowe, w celu określenia odległości spektralnej i przypisania pikselom odpowiednich klas (Kruse i in., 1993).

⁷ Support Vector Machine (SVM) - maszyna wektorów nośnych, polega na tworzeniu hiperpłaszczyzny optymalnie separującej dane, umożliwiając klasyfikację punktów na podstawie ich przynależności do różnych klas (Cortes i Vapnik, 1995).

do klasyfikacji danych źródłowych. Redukuje się liczbę kanałów spektralnych, przy jednoczesnym dążeniu do poprawy rozdzielczości przestrzennej (Raczko i in., 2022; Kaplan i in., 2023; Hikuwai i in., 2023).

Tab. 3 Zestawienie badań na temat wykrywania dachów azbestowych z wykorzystaniem obrazów wielospektralnych (Abbasi i in., 2022, zmodyfikowano)

Sensor hiperspektralny	Liczba kanałów spektralnych	Dane wspomagające	Rozdzielczość przestrzenna [m]	Badanie
QuickBird	4	-	panchromatyczny: 0,6 multispektralny: 2,4	Pinho i in., 2008
WV-2	8	-	panchromatyczny: 0,5 multispektralny: 1,85	Abriha i in., 2018 Gibril i in., 2017 Hamedianfar i S., 2015 Taherzadeh i S., 2013 Hikuwai i in., 2023
		LiDAR		Hamedianfar i in., 2014 Osińska-Skotak i O., 2015
WV-3	8	-	panchromatyczny: 0,31 Multispektralny: 1,24	Tommasini i in., 2019 Hikuwai i in., 2023
		LiDAR		Norman i in., 2020
IKONOS II	4	-	0,68	Pinho i in., 2012
Zdjęcia lotnicze	3 (RGB)	kompozycja CIR	0,25	Krówczyńska i in., 2020
		-		Raczko i in., 2022
		kanał termalny	0,52	Kaplan i in., 2023
		-	0,08	Hikuwai i in., 2023

W pracach badawczych, opierających się na obrazach satelitarnych, wykorzystywano metodę pansharpeningu, w celu uzyskania obrazu o wyższej rozdzielczości (Abriha i in., 2018; Gibril i in., 2017; Taherzadeh i in., 2014). W pracy Abriha i in. (2018) dowodzą efektywności stosowania pansharpeningu w procesie wykrywania dachów azbestowych. Otrzymana ogólna dokładność klasyfikacji była na poziomie 85%, a dla azbestu ponad 95%. W przypadku Gibril i in. (2017) uzyskana dokładność wyniosła ponad 90%.

W procesie klasyfikacji materiałów dachowych, jako dane wspomagające wykorzystywano dane z LiDAR-u⁸ (Hamedianfar i in., 2014; Norman i in., 2020; Osińska-Skotak i Ostrowski, 2015). W przeciwieństwie do dwuwymiarowych obrazów, dane pozyskane z tej technologii są niewrażliwe na zmienne warunki oświetleniowe (Hodgson i in., 2003). Uzyskane wartości parametrów oceny klasyfikacji potwierdziły pozytywny udział danych lidarowych w procesie klasyfikacji pokryć dachowych. Zwrócono jednak uwagę na wybór metody łączącej dane wejściowe, gdyż ma ona istotny wpływ na otrzymywane wyniki klasyfikacji (Norman i in., 2020). Ważnym czynnikiem wpływającym na rezultaty wykrywania pokryć dachowych jest również wybór metody i algorytmu klasyfikacji (Hamedianfar i in., 2014).

W badaniach Osińska-Skotak i Ostrowski (2015) wykonywano korekcje topograficzne obrazu satelitarnego WorldView-2 z wykorzystaniem numerycznego modelu terenu⁹ (DTM) oraz numerycznego modelu powierzchni terenu¹⁰ (DSM). Otrzymane wyniki wskazały na brak znaczących różnic w otrzymywanej dokładności klasyfikacji niezależnie od wykorzystywanego modelu terenu.

Wraz z rozwojem sztucznej inteligencji (AI, *ang. Artificial Intelligence*), zmieniło się również podejście do klasyfikacji obrazów. Aktualnie prowadzone badania dotyczące wykrywania dachów azbestowych opierają się na wysokorozdzielczych zdjęciach lotniczych, z wykorzystaniem zaawansowanych algorytmów sztucznej inteligencji m. in. sieci konwolucyjnych¹¹ (CNN) (Raczko i in., 2022; Kaplan i in., 2023, Hikuwai i in., 2023).

3.4 METODY KLASYFIKACJI

Identyfikowanie pokryć dachowych na obrazach realizowano przy użyciu różnych metod klasyfikacji, które można podzielić na trzy grupy: klasyfikacja pikselowa,

⁸ LiDAR (*ang. Light Detection and Ranging*) – technika pomiaru odległości z wykorzystaniem światła laserowego (Lillesand, 2015).

⁹ DTM⁹ (*ang. Digital Terrain Model*) to model cyfrowy przedstawiający powierzchnię terenu bez uwzględniania obiektów znajdujących się na nim, takich jak budynki, drzewa (Lillesand, 2015).

¹⁰ DSM¹⁰ (*ang. Digital Surface Model*) to model cyfrowy przedstawiający pełną powierzchnię terenu, z uwzględnieniem obiektów znajdujących się na nim, takich jak budynki, drzewa (Lillesand, 2015).

¹¹ CNN (*ang. Convolutional Neural Networks*) to rodzaj sieci neuronowej, należącej do grupy algorytmów głębokiego uczenia i jest szczególnie używany do rozpoznawania obrazów i zadań obejmujących przetwarzanie danych pikselowych (Nielsen, 2015).

klasyfikacja obiektowa oraz klasyfikacja wykorzystująca zaawansowane algorytmy uczenia maszynowego (Tab. 4).

3.4.1 Klasyfikacja pikselowa

Początkowe badania nad wykrywaniem azbestowo – cementowych pokryć dachowych opierały się na pikselowej analizie obrazu (Marino i in., 2001; Fiumi i in., 2012; Frassy i in., 2014; Szabo i in., 2014; Cilia i in., 2015; Osińska-Skotak i Ostrowski, 2015). Klasyfikacja pikselowa to metoda, w której każdy piksel rozpatrywany jest niezależnie. W tym podejściu, klasyfikacja odbywa się na poziomie poszczególnych pikseli, co pozwala na przypisanie wszystkich pikseli do odpowiednich klas wg ich informacji spektralnej (Lillesand i in., 2015).

Na podstawie przeprowadzonych badań i otrzymanych wyników dokładności, nie można jednoznacznie wskazać najlepszego klasyfikatora w kontekście klasyfikacji pikselowej (Tab. 4). Wynika to z faktu, iż prowadzone badania oparto o różne dane źródłowe, niejednolite przetwarzanie wstępne oraz zastosowano inne algorytmy klasyfikacji.

W pracy Szabo i in. (2014) porównano wyniki trzech klasyfikatorów (SAM, SVM, MLC) na tym samym zestawie danych. Najwyższy wynik klasyfikacji uzyskano dla klasyfikatora SVM. Zwrócono uwagę, że wykorzystanie klasyfikatora SAM wymaga bardzo dobrze zdefiniowanych pól treningowych oraz pozwala uzyskać wyższe wyniki dla większej liczby klas.

Współczesne obrazy charakteryzują się coraz wyższą rozdzielczością przestrzenną. Niemniej jednak, gdy rozmiar piksela terenowego maleje, klasyfikacja na poziomie piksela staje się mniej skuteczna. Odpowiedź spektralna pokryć dachowych jest bardzo zróżnicowana, co może mieć negatywny wpływ na wyniki procesu klasyfikacji (Myint i in., 2011; Hamedianfar i in., 2014).

Tab. 4 Zestawienie metod klasyfikacji, stosowanych klasyfikatorów oraz oceny dokładności, w badaniach nad wykrywaniem azbestowo – cementowych pokryć dachowych (Abbasi i in., 2022, zmodyfikowano)

Badanie	Obraz źródłowy ¹²	Klasyfikator	Ocena dokładności ¹³ [%]
Klasyfikacja pikselowa			
Marino i in. 2001	HSI	SAM	OA: 85
Bassani i in, 2007	HSI	SFF	OA: 80-90
Fiumi i in. 2012	HSI	SAM	OA: 88, PA (azbest): 94
Fiumi i in. 2014	HSI	SAM	OA: 65
Frassy i in. 2014	HSI	SAM	OA: 80, PA (azbest): 43
Osińska-Skotak 2014	MSI	Maximum Likelihood (MLC)	PA (azbest): 82-87 UA: (azbest): 61-74
Szabo i in. 2014	HSI	SVM SAM	OA: 80 OA: 60
Cilia i in, 2015	HSI	SAM	PA: 89, UA: 86
Osińska-Skotak i Ostrowski, 2015	MSI + ALS	Maximum Likelihood (MLC)	PA (azbest): 81-85 UA: (azbest): 57-66
Abriha i in. 2018	MSI	DFA, RF	OA: 85
Tommasini i in. 2019	MSI	RF	OA: 82
Klasyfikacja obiektowa			
Pinho i in., 2012	MSI	Drzewo decyzyjne (DT)	OA: 72 PA (azbest): 38-90
Taherizade i Shafri., 2013	MSI	Maximum Likelihood (MLC) wg reguł	OA: 46 OA: 82
Hamedianfar i in., 2014	MSI + LiDAR	wg reguł	OA: 93 PA: 81, UA: 93 (azbest)
Hamedianfar i Shafri, 2015	MSI	wg reguł	OA: 88 PA: 96, UA: 84 (azbest)
Gibril i in., 2017	MSI	Bayes, k-NN, SVM, RF wg reguł	OA: 72-82 OA: 90-93
Norman i in., 2020	MSI + LiDAR	SVM, Drzewo decyzyjne (DT)	OA: 70 OA: 87
Klasyfikacja z wykorzystaniem uczenia maszynowego			
Krówczyńska i in., 2020	RCB + CIR	CNN	OA: 89, PA: 89, UA: 88 (azbest)
Raczko i in., 2022	RGB	CNN	OA: 88-91, PA: 89-90, UA: 84-90 (azbest)
Yu i in., 2022	HSI, MSI, RGB	SVM, CNN	OA: 88
Hikuwai i in., 2023	MSI, RGB	CNN	Precyzja: 94
Kaplan i in., 2023	RGB, kanał termalny	RF SVM XGBoost	Balanced Accuracy: 89 - 93

¹² HSI (ang. *HyperSpectral Imagery*) – obraz hiperspektralny; MSI (ang. *MultiSpectral Imagery*) – obraz wielospektralny; ALS (ang. *Airborne Laser Scanning*) – skanowanie lotnicze.

¹³ OA (ang. *Overall Accuracy*) – dokładność całkowita; PA (ang. *Producer's Accuracy*) – dokładność producenta; UA – (ang. *User's Accuracy*) – dokładność użytkownika (Lillesand i in., 2015).

3.4.2 Klasyfikacja obiektowa

Klasyfikacja obiektowa obrazu polega na podzieleniu go na segmenty (obiekty), które są grupami pikseli o podobnych cechach. W tym podejściu analizuje się zarówno cechy przestrzenne, jak i spektralne obiektów. Wykorzystuje się atrybuty takie jak tekstura, rozmiar i geometria, aby opracować reguły klasyfikacji (Lillesand i in., 2015). W związku z tym, klasyfikacja obiektowa umożliwia bardziej precyzyjną i kompleksową analizę obrazów o wysokiej rozdzielczości przestrzennej. Przeprowadzone badania, w temacie wykrywania dachów azbestowych metodą obiektową, zestawiono w tabeli (Tab. 4).

Klasyfikacja obiektowa dzieli się na dwa etapy: segmentację obrazu na oddzielne obiekty oraz ich klasyfikację (Lillesand i in., 2015). W badaniach na temat wykrywania dachów azbestowych korzystano z segmentacji opartej na krawędziach lub regionach. Nie analizowano szeroko tego etapu, często opierając się na dostępnej wiedzy i doświadczeniu innych badaczy.

Etap klasyfikacji utworzonych obiektów w opisywanych badaniach można podzielić na dwa podejścia: klasyfikacja nadzorowana (Gibril i in., 2017) oraz klasyfikacja wg reguł (Pinho i in., 2012; Hamedianfar i in., 2014). Klasyfikacja nadzorowana wykorzystuje parametry statystyczne pól treningowych, czyli wskazanych fragmentów obrazu, jako wzorców klas. Na podstawie pozyskanej informacji, klasyfikator przypisuje piksele do zdefiniowanych klas. Z kolei klasyfikacja wg reguł korzysta ze zdefiniowanego zestawu wytycznych. Wykorzystywane reguły mogą być wyznaczane na podstawie danych treningowych lub być definiowane na podstawie specjalistycznej wiedzy eksperta (Abbasi i in. 2022).

Klasyfikacja obiektowa przeprowadzana wg reguł, przy dobrze zdefiniowanym modelu, może być wykorzystywana na obrazach różnych terenów, bez konieczności ponownego przeprowadzania procesu nauki. W pracy Gibril i in. (2017) zastosowano zdefiniowany zestaw reguł na niezależnym obrazie WorldView-2, przedstawiającym dwukrotnie większy obszar. Otrzymane wyniki były zadowalające, gdyż osiągnięto dokładność na poziomie 92%.

3.4.3 Sztuczna inteligencja w wykrywaniu dachów azbestowych

Rozwój uczenia maszynowego, w szczególności w kierunku uczenia głębokiego, pozwolił na poprawę wykrywania obiektów na obrazach (Guo i in., 2018). W związku z tym, w literaturze pojawiły się publikacje dotyczące detekcji dachów azbestowych z wykorzystaniem sztucznej inteligencji (Tab. 4). Wykonane badania były bardziej złożone, ale dotyczyły większych obszarów terenu, w porównaniu z metodami tradycyjnymi (Raczko i in., 2022; Yu i in., 2022; Hikuwai i in., 2023).

Autorzy Krówczyńska i in. (2020) zaproponowali klasyfikację dachów azbestowych z wykorzystaniem ortofotomap o rozdzielczości przestrzennej 0,25m: kompozycja RGB oraz CIR¹⁴, z wykorzystaniem głębokiego uczenia CNN. Dokładność ogólna opisanej metody wynosiła 89%. Zwrócono uwagę, że chociaż wyniki dla obszarów wiejskich są obiecujące, nie jest pewne, czy proponowana metoda może być skutecznie stosowana w gęsto zaludnionych obszarach miejskich, takich jak duże miasta. Z kolei w pracy Raczko i in. (2022) przedstawiono dalsze badania nad zaproponowaną w 2020 roku metodą. Ograniczono zestaw danych wejściowych wyłącznie do ortofotomapy RGB, natomiast poszerzono badania o drugi niezależny obszar. Zmodyfikowana metoda identyfikacji dachów azbestowych była niewrażliwa na termin wykonania zdjęcia lotniczego jak i na lokalizację badanego obszaru (otrzymano najniższą dokładność na poziomie 88%).

W pracy Yu i in. (2022) wykorzystano kilka obrazów terenu o różnej rozdzielczości przestrzennej i spektralnej: archiwalne lotnicze zdjęcia hiperspektralne, archiwalne ortofotomapy oraz obraz satelitarny Sentinel 2. W związku z tym, klasyfikacja przebiegała etapowo. Otrzymano dokładność wykrywania dachów azbestowych powyżej 85%. Wskazano, iż istotny wpływ na uzyskane wyniki miało zastosowanie obrazów hiperspektralnych w zestawieniu z wynikami skanowania oryginalnych próbek dachowych w laboratorium. Zwrócono również uwagę na czas przetwarzania danych. Wykonano klasyfikację przy użyciu algorytmu SVM oraz głębokiego uczenia CNN. Działanie algorytmu SVM mniej obciążało sprzęt komputerowy i przebiegało szybciej. Natomiast metoda CNN oferowała lepszą dokładność za pomocą uniwersalnego modelu treningowego dla rozległych obszarów, ale była czasochłonna.

¹⁴ Kompozycja CIR (*ang. Color Infrared*) – kompozycja barwna wykorzystująca kanał zielony, czerwony oraz bliską podczerwień.

Badanie Kaplan i in. (2023) opierało się na lotniczych zdjęciach RGB z dodatkowym kanałem termalnym. W czasie klasyfikacji korzystano z informacji o jasności pikseli, wybranych wskaźnikach spektralnych oraz cechach teksturalnych. Poddawano ocenie trzy klasyfikatory: RF¹⁵, SVM, XGBoost¹⁶. Klasyfikacja wg algorytmu SVM, wykorzystująca wyłącznie wskaźniki spektralne dla kanałów RGB, osiągnęła dokładność na poziomie 93%. Uzyskane wyniki pokazały, że wskaźniki spektralne liczone na kanałach RGB mogą z powodzeniem być stosowane podczas klasyfikacji dachów azbestowych, gdyż otrzymane dokładności są na podobnym poziomie jak w przypadku klasyfikacji przy użyciu danych hiper – i wielospektralnych.

W pracach badawczych Hikuwai i in. (2023) wykrywano dachy azbestowe na wysokorozdzielczych zdjęciach lotniczych RGB (piksel terenowy – 8cm) oraz ośmiokanałowym obrazie satelitarnych Worldview-3 (piksel terenowy – 1,24m). Badania oparto o klasyfikację głębokiego uczenia CNN. Model wykorzystujący wyłącznie dane z wysokorozdzielczych obrazów lotniczych uzyskał najwyższą precyzję, na poziomie 94%.

Przytoczone wyniki dokładności klasyfikacji potwierdzają skuteczność połączenia zaawansowanych technik sztucznej inteligencji i obrazów teledetekcyjnych do wykrywania pokryć dachowych zawierających azbest.

3.4.4 Klasyfikatory binarne

Ogólnie za klasyfikator binarny możemy uznać taki algorytm uczenia maszynowego, który służy do przewidywania przynależności obiektów do jednej z dwóch klas. Klasyfikacja binarna odnosi się zatem do zadania podziału danych na dwie kategorie lub klasy. Podziału dokonujemy na podstawie pewnych dostępnych cech lub atrybutów (Géron, 2020).

W przypadku klasyfikatora binarnego, każdy obiekt wejściowy przypisywany jest do jednej z dwóch klas, które często oznaczane są jako klasa pozytywna – wartość 1 oraz klasa negatywna – wartość 0. Algorytm binarnej klasyfikacji nadzorowanej opiera się na analizie cech (atrybutów) obiektów treningowych, które są jednoznacznie oznaczone –

¹⁵ RandomForestClassifier (RF) – patrz rozdział 3.4.4.

¹⁶ XGBoost (*ang. eXtreme Gradient Boosting*) – klasyfikator będący rozszerzeniem tradycyjnego algorytmu GradientBoostingClassifier (GBC, patrz rozdział 3.4.4) poprzez dodanie nowych funkcji i optymalizacji, takich jak automatyczne dostrajanie hiperparametrów, obsługa różnych funkcji straty i obsługa rozproszonego przetwarzania (<https://arxiv.org/abs/1603.02754v3>).

należą do jednej z dwóch klas. Zadaniem wytrenowanego klasyfikatora jest przypisanie klas nowym, nieoznaczonym jeszcze obiektom (Géron, 2020).

Przykładami klasyfikatorów są: SGD Classifier (SGD), Support Vector Machines (SVM), Random Forests Classifier (RFC), Gradient Boosting Classifier (GBC), Extra Trees Classifier (ETC). Każdy z tych algorytmów uczenia maszynowego posiada swoje unikalne cechy, zalety czy ograniczenia. Wszystkie jednak z powodzeniem można wykorzystywać do rozwiązywania problemów klasyfikacji binarnej (Géron, 2020). Wartościowe jest również to, że wspomniane klasyfikatory są dostępne w bibliotece Scikit-Learn w języku *Python*, co ułatwia ich implementację oraz integrację z innymi narzędziami i bibliotekami wykorzystywanymi do analizy danych.

W dalszej części rozdziału zaprezentowano krótką charakterystykę stosowanych w pracy klasyfikatorów.

Klasyfikator SGD

SGDClassifier jest klasyfikatorem opartym na stochastycznym spadku wzdłuż gradientu (Stochastic Gradient Descent), który wykorzystuje liniową funkcję decyzyjną do przewidywania etykiet klas na podstawie danych wejściowych (Géron, 2020). Sam algorytm SGD jest jedynie techniką optymalizacji i nie odpowiada konkretnej rodzinie modeli uczenia maszynowego (Pedregosa i in., 2011).

Głównym celem algorytmu SGD jest minimalizacja funkcji kosztu poprzez dobór parametrów modelu na podstawie gradientu (spadku) funkcji kosztu oszacowanego dla wybranych losowo pojedynczych danych treningowych. Można ogólnie założyć, że działa on na zasadzie iteracyjnego aktualizowania parametrów modelu w celu znalezienia optymalnego minimum funkcji kosztu (Géron, 2020).

Klasyfikator ten posiada pewne zalety, ale również wady. Do zalet można zaliczyć jego efektywność oraz łatwość implementacji (Pedregosa i in., 2011). Dodatkowo jest on szybki i odpowiedni dla dużych zestawów danych, gdyż w każdym przebiegu algorytmu wykorzystywane są pojedyncze, losowo wybrane dane treningowe (Géron, 2020). Wadami klasyfikatora SGD jest jego wrażliwość na skalowanie funkcji oraz wymóg definiowania hiperparametrów, takich jak funkcja straty, regularyzacja, czy liczba iteracji (Pedregosa i in., 2011).

Klasyfikator SVC

SVC (*ang. Support Vector Classifier*) to klasyfikator binarny oparty na tak zwanych maszynach wektorów nośnych (*ang. Support Vector Machines*). Jest to kolejna popularna metoda wykorzystywana w klasyfikacji, której celem jest znalezienie hiperpłaszczyzny rozdzielającej przestrzeń cech na dwie klasy (Géron, 2020).

Działanie klasyfikatora SVC opiera się na trzech podstawowych zasadach. Pierwsza z nich to tworzenie wektorów nośnych, czyli wskazanie danych treningowych znajdujących się najbliżej granicy decyzyjnej (hiperpłaszczyzny). Druga zasada to optymalizacja tzw. marginesu, to znaczy klasyfikator poszukuje hiperpłaszczyzny z największą odległością między wyznaczonymi wektorami nośnymi, a granicami decyzyjnymi. Ostatnia zasada działania SVC to wykorzystanie transformacji nieliniowej w celu przekształcenia przestrzeni cech, tak by dane były bardziej podzielne liniowo (Géron, 2020; Pedregosa i in., 2011).

Klasyfikator SVC jest skuteczny w przestrzeniach wielowymiarowych, nawet jeżeli liczba wymiarów jest większa niż liczba próbek (Pedregosa i in., 2011). Z kolei za wadę SVC można uznać wrażliwość na parametry definiowane podczas pracy z tym klasyfikatorem. Wybór odpowiednich wartości parametrów może być trudny i wymagający eksperymentowania. Nieprawidłowo dobrane wartości parametrów mogą prowadzić do przeuczenia lub niedouczenia modelu (Géron, 2020).

Klasyfikator RFC

RandomForestClassifier (RFC) to klasyfikator oparty na losowym lesie drzew decyzyjnych, czyli na złożeniu wielu klasyfikatorów zbudowanych na bazie pojedynczych drzew decyzyjnych. Każde drzewo decyzyjne jest trenowane na różnych podzbiorach danych treningowych i losowo wybranych podzbiorach cech (Géron, 2020; Ho, 1995).

Pojedynczy klasyfikator drzewa decyzyjnego składa się z korzenia, gdzie rozpoczyna się podejmowanie decyzji. Następnie są węzły (testowanie danych) i krawędzie (wyniki), zaś na końcu każdego rozgałęzienia jest liść (przypisanie klasy) (Géron, 2020). W module `Stick-Learn` dostępne są wyłącznie drzewa binarne, czyli węzły niebędące liśćmi zawsze posiadają tylko dwoje potomków (Pedregosa i in., 2011).

Algorytm losowego lasu ma za zadanie wprowadzić losowość do wzrostu drzew. Zamiast wyszukiwania najlepszej cechy podczas podziału na podrzędne węzły, szuka on najlepszej cechy wśród losowego podzbioru cech. W efekcie występuje większe zróżnicowanie drzew, co najczęściej oznacza lepszy model (Géron, 2020).

Klasyfikator RFC posiada kilka istotnych zalet. Do najważniejszych korzyści można zaliczyć odporność na przeuczenie oraz zdolność pracy z dużą liczbą cech i na różnych rodzajach danych. Pomimo swojej złożoności klasyfikator ten stanowi jeden z najpotężniejszych współczesnych algorytmów uczenia maszynowego (Géron, 2020; Pedregosa i in., 2011)

Klasyfikator ETC

ExtraTreesClassifier (ETC) to klasyfikator stanowiący rozszerzenie RFC i jest on oparty na losowym lesie tak zwanych ekstremalnych drzew decyzyjnych (Geurts i in., 2006). Główna różnica między tymi dwoma klasyfikatorami polega na tym, że ETC wykorzystuje jeszcze bardziej przypadkowe drzewa poprzez wprowadzenie losowych progów dla każdej cechy, zamiast poszukiwania najlepszego kryterium podziału węzłów jak w RFC (Pedregosa i in., 2011).

ETC oferuje podobne korzyści jak RFC, takie jak odporność na przeuczenie oraz zdolność do obsługi dużych zbiorów danych. Dodatkowo, losowy podział węzłów może prowadzić do większej różnorodności w drzewach, co może zwiększyć zdolność do wykrywania subtelnych wzorców w danych. Generalnie losowe lasy są narzędziem, które jest bardzo pomocne w określaniu cech mających istotne znaczenie podczas klasyfikacji (Géron, 2020).

Klasyfikator GBC

GradientBoostingClassifier (GBC) to klasyfikator oparty na gradientowym wzmacnianiu drzew decyzyjnych. Jest to popularny algorytm zespołowy wykorzystujący wiele słabych klasyfikatorów (w tym przypadku drzew decyzyjnych), które są sekwencyjnie dopasowywane do danych treningowych w celu poprawy jakości predykcji (Friedman, 2001; Géron, 2020; Pedregosa i in., 2011).

Działanie klasyfikatora GBC przebiega wg kilku zasad. Na początku, na podstawie danych treningowych, dopasowywany jest prosty model bazowy, np. drzewo decyzyjne. Następnie sekwencyjnie dopasowuje się kolejne modele do reszt, czyli różnic między rzeczywistymi etykietami, a predykcjami poprzednich modeli. Każdy nowy model jest trenowany w celu minimalizacji błędów reszt, co prowadzi do stopniowej poprawy predykcji. Wszystkie modele mają przypisywane i aktualizowane wagi w trakcie treningu na podstawie błędów resztkowych. Ostateczna prognoza wynika z połączenia predykcji wszystkich modeli, przy uwzględnieniu ich wag (Friedman, 2001; Géron, 2020; Pedregosa i in., 2011).

3.5 ALGORYTMY WSPOMAGAJĄCE PROCES KLASYFIKACJI

Schemat badań wykrywania wybranych obiektów na obrazach teledetekcyjnych, w głównej mierze opiera się na wiedzy i doświadczeniu ekspertów. W celu poprawy rezultatów klasyfikacji, oprócz analizy jasności piksela jako podstawowych danych z obrazów teledetekcyjnych, zaleca się stosowanie dodatkowych przekształceń i wskaźników. Sugerowane jest stosowanie pan-sharpeningu w przypadku występowania różnych rozdzielczości przestrzennych dostępnych obrazów (Wang i in. 2019; Gašparović i Jogun, 2018). Do wyznaczenia pomocniczych danych rastrowych można wykorzystać algebrę rastrów i wyliczyć dodatkowe wskaźniki spektralne (Viana i in., 2019; Talukdar i in., 2020). Z kolei podczas analizy wysokorozdzielczych obrazów satelitarnych, proponuje się użycie dodatkowych cech teksturalnych (GLCM, *ang. Gray-Level Co-Occurrence Matrix*) (Haralick, 1979). Cechy GLCM, nazywane również parametrami teksturalnymi Haralick'a, zostały opracowane i wprowadzone przez Haralick'a w 1979 roku. Wielu autorów podkreśla wzrost dokładności klasyfikacji obrazów z wykorzystaniem cech GLCM (np. Huang i in., 2020; Park i Guldmann, 2020).

Problem wspomaganie selekcji danych w celu poprawy uzyskiwanych wyników klasyfikacji dachów azbestowych podjęto w pracy Kaplan i in. (2023). Przeanalizowano wpływ wyboru danych wejściowych na proces klasyfikacji z wykorzystaniem algorytmu Boruta (Kursa i Rudnicki, 2010). Koncepcja metody Boruta oparta jest na poszerzeniu całego zakresu dostępnych informacji o tzw. cienie (nieistotne cechy), które wykorzystywane są jako odniesienie dla oceny ważności oryginalnych cech w całej strukturze danych. Miara przydatności każdej cechy szacowana jest przez systemy uczące w

czasie treningu. W domyślnych ustawieniach wykorzystywana jest metoda lasu losowego (RF) (Breiman, 2001). Selekcja odbywa się iteracyjnie, gdzie krokowo usuwane są cechy uznawane za nieistotne.

Podczas badań wykorzystano obraz RGB o rozdzielczości przestrzennej 53cm oraz pojedynczy kanał termalny (Kaplan i in., 2023). Klasyfikacja odbywała się na różnych zestawach danych wejściowych, składających się z wartości jasności pikseli, wyliczonych prostych wskaźników spektralnych oraz parametrów teksturalnych Haralick'a. Otrzymane wyniki pokazały, że obraz kanału termalnego oraz cechy GLCM nie poprawiły dokładności procesu klasyfikacji. Natomiast wyznaczenie wskaźników spektralnych spowodowało zwiększenie skuteczności klasyfikatorów. Dodatkowo, należy podkreślić, że cechy GLCM wyznaczone dla próbek azbestowych, nie są odporne na zmianę kontrastu, jasności i ostrości obrazu (Książek, 2018). W związku z powyższym, w dalszych badaniach należy poszukiwać innych cech tekstury obrazu lub skupić się na maksymalnym wykorzystaniu rejestrowanej informacji spektralnej.

W przypadku klasyfikacji metodą obiektową, segmentacja obrazu jest pierwszym i bardzo ważnym krokiem, od którego w dużej mierze zależy wynik klasyfikacji. W badaniach Gibril i in. (2017) zastosowano technikę statystyczną Taguchi¹⁷ do wskazania najlepszych parametrów segmentacji obrazu WorldView-2 dla niejednorodnego obszaru miejskiego. Zastosowane rozwiązanie zmniejszyło liczbę eksperymentów wykonanych w celu wyznaczenia optymalnych parametrów segmentacji, poprawiło wydajność procesu klasyfikacji oraz skróciło czas przetwarzania danych.

3.5.1 Algorytmy genetyczne

Pierwszy algorytm genetyczny (AG) został opracowany przez Johna Hollanda w latach 60. Holland był wybitnym amerykańskim naukowcem zajmującym się systemami adaptacyjnymi i sztuczną inteligencją. Jego prace nad algorytmami genetycznymi były inspirowane mechanizmami ewolucji biologicznej oraz dziedziczenia genetycznego (Goldberg, 1998).

¹⁷ Metoda Taguchi optymalizuje procesy (np. wybór parametrów modelu uczenia maszynowego, ustawienie maszyn w procesach produkcyjnych) poprzez minimalizację wpływu czynników zakłócających i identyfikację optymalnych ustawień parametrów (Aggarwal i in., 2008).

W 1975 roku John Holland opublikował swoją kluczową pracę "Adaptation in Natural and Artificial Systems" (Adaptacja w systemach naturalnych i sztucznych), w której opisał podstawowe zasady algorytmu genetycznego oraz jego zastosowanie do optymalizacji (Holland, 1975). Ta praca uznawana jest za kamień milowy w rozwoju algorytmów genetycznych i przyczyniła się do rozpowszechnienia tej techniki w dziedzinie sztucznej inteligencji i optymalizacji (Goldberg, 1998). Można przyjąć, że od tego czasu algorytmy genetyczne stały się popularnym narzędziem w wielu dziedzinach, takich jak inżynieria, nauki społeczne, ekonomia, czy biologia (Winiczenko, 2008). Wiele różnych wariantów i modyfikacji algorytmów genetycznych zostało opracowanych przez innych naukowców na przestrzeni lat, co daje możliwość dostosowania tej techniki do rozwiązywania różnorodnych problemów w różnych warunkach (Goldberg, 1998; Michalewicz, 2003).

Jako podstawowy przykład problemu, który trudno jest rozwiązać tradycyjnymi metodami, ale dla którego algorytm genetyczny może być skuteczny, to tzw. problem komiwojażera (*ang. Traveling Salesman Problem*, TSP) (Michalewicz, 2003, Pál, 1993; Potvin, 1996). Problem komiwojażera polega na znalezieniu najkrótszej drogi, przemierzając którą dostawca odwiedzi każde z n miast dokładnie raz i wróci do miasta początkowego. Jest to problem optymalizacyjny, który ma wiele możliwych kombinacji tras i staje się bardzo trudny do rozwiązania w przypadku większej liczby miast (Pál, 1993; Potvin, 1996).

Tradycyjne metody rozwiązywania problemu komiwojażera, takie jak przegląd zupełny lub programowanie dynamiczne, stają się niepraktyczne lub nieefektywne dla większych instancji danych. Algorytmy genetyczne mogą jednak być wykorzystane do znalezienia rozwiązań przybliżonych dla tego problemu poprzez generowanie i ewolucję populacji tras (Michalewicz, 2003). Dzięki swojej zdolności do przeszukiwania dużych przestrzeni rozwiązań i znajdowania dobrych przybliżeń optymalnych, algorytmy genetyczne są często wykorzystywane do rozwiązywania tak złożonych zadań jak przytoczony przykład zadania komiwojażera tras (Michalewicz, 2003).

Konkretnie algorytm genetyczny to pewna metaheurystyczna technika optymalizacji inspirowana naturalnymi procesami ewolucyjnymi zaczerpniętymi z przyrody. Jest szeroko wykorzystywany do rozwiązywania problemów optymalizacyjnych, w których poszukujemy najlepszego rozwiązania zazwyczaj dla bardzo dużej przestrzeni rozwiązań

dopuszczalnych (Goldberg, 1998). Podstawową ideą algorytmu genetycznego jest symulowanie ewolucji biologicznej w celu znalezienia optymalnego rozwiązania (Michalewicz, 2003). Należy zauważyć, że ze względu na to, iż algorytm ten jest heurystyką, nie mamy pewności, że otrzymane rozwiązanie jest optymalne (Goldberg, 1998; Winiczenko, 2008). Sam algorytm operuje na populacji osobników, które reprezentują potencjalne rozwiązania problemu. Każdy osobnik w populacji jest kodowany za pomocą genotypu, który jest zazwyczaj reprezentowany jako ciąg bitów lub inna struktura danych, np. permutacje, drzewa lub grafy (Winiczenko, 2008). W niniejszej rozprawie wykorzystano kodowanie chromosomu jako ciągu liczb rzeczywistych, będących filtrem dla danych spektralnych na obrazie wielospektralnym.

Możemy przyjąć, że algorytm genetyczny składa się zawsze z kilku głównych kroków (Goldberg, 1998; Michalewicz, 2003):

1. **Inicjalizacja populacji:** Na początku tworzona jest populacja początkowa, zazwyczaj losowo generowana lub zdefiniowana w sposób heurystyczny. Populacja składa się z pewnej liczby osobników (Winiczenko, 2008). W pracy do wyznaczenia populacji początkowej wykorzystano prosty losowy algorytm heurystyczny polegający na poszukiwaniu dobrych rozwiązań poprzez wielokrotne losowanie rozwiązań, i wybranie tych o najlepszej funkcji celu.
2. **Ocena funkcji celu (funkcji dopasowania):** Dla każdego osobnika w populacji obliczana jest wartość funkcji celu, która określa jakość rozwiązania reprezentowanego przez dany osobnik. Funkcja celu jest zależna od problemu, który próbujemy rozwiązać (Winiczenko, 2008; Eiben i Smith, 2015). W pracy do oceny funkcji dopasowania służyły dwie miary: dokładność klasyfikacji oraz parametr F1 dla azbestu.
3. **Selekcja:** Następnie wybierane są osobniki do reprodukcji na podstawie ich wartości funkcji celu. Oczekuje się, że osobniki o najwyższym stopniu przystosowania do funkcji celu będą mieć liczne potomstwo, tzn. ich materiał genetyczny będzie wykorzystany do tworzenia kolejnej populacji (Winiczenko, 2008). Selekcja może odbywać się na różne sposoby, na przykład za pomocą turnieju, ruletki lub rankingowania (Eiben i Smith, 2015). W pracy wykorzystano selekcję turniejową, polegającą na losowym wyborze z całej populacji kilku osobników (tzw. grupa turniejowa), a później z tej grupy wybierany jest osobnik najlepiej

przystosowany i on przepisany jest do nowo tworzonej populacji. Losowanie grup turniejowych oraz wybieranie z nich najlepszego osobnika jest powtarzane aż do utworzenia całej nowej populacji (Eiben i Smith, 2015).

4. **Krzyżowanie:** Wybrane osobniki są poddawane operacji krzyżowania, w celu wymiany informacji genetycznej. Proces ten polega na losowym przecięciu osobników, a następnie zamianie podzielonych części. W wyniku tej operacji powstaje potomstwo, które dziedziczy cechy od swoich rodziców (Winiczenko, 2008).

5. **Mutacja:** Potomstwo podlega mutacji, która jest losową zmianą genotypu. Mutacja wprowadza różnorodność genetyczną do populacji, umożliwiając badanie nowych obszarów przestrzeni rozwiązań. Operacja mutacji polega na losowej zmianie pojedynczego fragmentu osobnika (Winiczenko, 2008).

6. **Nowa generacja:** Na podstawie wybranych osobników, potomstwa i mutacji tworzona jest nowa populacja, która zastępuje poprzednią (Winiczenko, 2008).

Algorytm genetyczny działa przez określoną liczbę iteracji lub do spełnienia pewnego kryterium zatrzymania (tzw. „stopu”), takiego jak osiągnięcie zadowalającego rozwiązania lub upływu czasu (Smoliński, 2016). W pracy kryterium zatrzymania algorytmu stanowiła liczba generacji populacji. Algorytm genetyczny jest iteracyjny i powtarza powyższe kroki, starając się poprawiać jakość rozwiązań w każdej kolejnej generacji (Eiben i Smith, 2015).

Wykorzystując algorytmy genetyczne można modyfikować ustawienia poszczególnych kroków, które wpływają na sposób działania i zbieżność algorytmu. Niektóre z tych ustawień to:

1. **Elitaryzm:** Najlepsze osobniki z każdej generacji są przechowywane bez zmiany, aby zapewnić zachowanie najlepszego rozwiązania. Elitaryzm pozwala uniknąć utraty znalezionej dotąd najlepszego rozwiązania. (Baluja i Caruana, 1995). Strategie tę wykorzystano w niniejszej pracy.

2. **Operator selekcji:** Istnieje wiele różnych operatorów selekcji, które decydują, które osobniki zostaną wybrane do reprodukcji. Może to być m. in. turniej

selekcyjny, selekcja ruletkowa, selekcja rankingowa (Eiben i Smith, 2015). W pracy zastosowano wspomnianą wcześniej selekcję turniejową.

3. Operator krzyżowania: W zależności od rodzaju problemu i reprezentacji osobników, istnieje wiele operatorów krzyżowania, takich jak krzyżowanie jednopunktowe, dwupunktowe, jednorodne, a także bardziej zaawansowane operatory krzyżowania dostosowane do konkretnego problemu (Eiben i Smith, 2015). W pracy zastosowano krzyżowanie dwupunktowe, dzielące wylosowanego osobnika na trzy części.

4. Operator mutacji: Mutacja wprowadza losowe zmiany do genotypu osobników w celu zwiększenia różnorodności populacji. W zależności od reprezentacji genotypu, istnieją różne operatory mutacji, takie jak mutacja jednopunktowa, zamiana bitów, mutacja gaussowska (Eiben i Smith, 2015). W pracy zastosowano operator mutacji jednopunktowej, czyli mutowano jeden element z osobnika.

5. Parametry algorytmu: Algorytm genetyczny ma wiele parametrów, takich jak rozmiar populacji, liczba iteracji, prawdopodobieństwo mutacji, prawdopodobieństwo krzyżowania. Dobra kalibracja tych parametrów może mieć istotny wpływ na skuteczność i zbieżność algorytmu (Eiben i Smith, 2015, Michalewicz, 2003). W pracy zastosowano populację składającą się ze 100 osobników, 30 iteracji, prawdopodobieństwo mutacji na poziomie 0,02 oraz prawdopodobieństwo krzyżowania na poziomie 0,65.

Ogólnie można założyć, że algorytmy genetyczne są elastycznym narzędziem, które można dostosować do różnych problemów optymalizacyjnych (Winiczenko, 2008). Niestety sama efektywność algorytmu genetycznego zależy od odpowiedniego dostosowania parametrów, reprezentacji osobników, operacji genetycznych i strategii selekcji. Jednak w praktyce, eksperymentowanie z różnymi strategiami i dostosowywanie parametrów pozwala na wyznaczenie bardzo dobrych, często zbliżonych do optymalnego, rozwiązań dla danego problemu (Eiben i Smith, 2015).

Należy również wskazać że pomimo, iż algorytmy genetyczne są bardzo skutecznymi narzędziami do rozwiązywania różnorodnych problemów optymalizacyjnych, mają również swoje wady, wynikające z ich konstrukcji. Do głównych wad algorytmów genetycznych można m. in. zaliczyć:

1. Wymaganie dużych zasobów obliczeniowych. Algorytmy genetyczne mogą być czasochłonne i wymagać obliczeń na dużą skalę, szczególnie dla problemów o dużej złożoności. Duże populacje, liczba iteracji i operacje na chromosomach mogą znacząco zwiększać czas obliczeń (Grygierek, 2014).

2. Oczywisty brak gwarancji znalezienia optymalnego rozwiązania. Algorytmy genetyczne działają na zasadzie heurystycznej, co oznacza, że nigdy nie ma gwarancji, że algorytm znajdzie optymalne rozwiązanie. W zależności od jakości kodowania, operatorów genetycznych i innych parametrów, algorytmy genetyczne mogą zatrzymać się w lokalnym optimum lub osiągnąć tylko rozwiązanie zbliżone do optymalnego (Smoliński, 2016).

3. Słaba skalowalność. Wraz ze wzrostem złożoności problemu, algorytmy genetyczne mogą napotykać trudności z efektywnym skalowaniem. Zwiększenie liczby wymiarów, rozmiaru populacji lub liczby iteracji może znacząco wpływać na wydajność i czas wykonania algorytmu (Goldberg, 1998).

4. Zależność od parametrów. Algorytmy genetyczne posiadają różne parametry, takie jak rozmiar populacji, prawdopodobieństwo krzyżowania i mutacji, które muszą być odpowiednio dostrojone. Nieprawidłowe ustawienie tych parametrów może prowadzić do suboptymalnych wyników lub braku zbieżności algorytmu (Smoliński, 2016).

5. Ograniczenia w reprezentacji problemu. Algorytmy genetyczne wymagają odpowiedniego kodowania w postaci chromosomów. Niektóre zagadnienia mogą być trudne do przedstawienia za pomocą chromosomów lub wymagać specjalnego kodowania genetycznego (Goldberg, 1998).

Podsumowując, można stwierdzić, że pomimo pewnych wad, algorytmy genetyczne pozostają ważnym narzędziem w dziedzinie optymalizacji oraz wykorzystywane są w wielu dziedzinach nauki i techniki (Goldberg, 1998). Odpowiednie dostrojenie modelu poprzez dobór właściwych parametrów oraz selekcja odpowiednich operatorów genetycznych dla większości zadań pozwalają na minimalizację wspomnianych wad oraz uzyskanie skutecznego działania algorytmu genetycznego (Michalewicz, 2003).

3.6 OCENA PRZYDATNOŚCI KLASYFIKATORÓW

Do analizy wyników klasyfikacji wykorzystano macierz błędów (*ang. confusion matrix*), która przedstawia wyniki klasyfikacji modelu na podstawie wartości prawdziwych (rzeczywistych) i wartości przewidywanych (prognozowanych) (Géron, 2020; Foody, 2002).

Macierz błędów to tablica o rozmiarze $N \times N$, gdzie N reprezentuje liczbę klas (etykiet) podczas klasyfikacji (Tab. 5). W rozpatrywanym przypadku rozpatrujemy klasyfikację binarną ($N=2$), gdyż występują tylko dwie klasy: dachy azbestowe (wartość klasy 1) i inne pokrycia dachowe (wartość klasy 0). Tablica organizowana jest w następujący sposób: na osi poziomej znajdują się przewidywane etykiety (prognozy), natomiast na osi pionowej etykiety prawdziwe (rzeczywistość). Każda komórka w macierzy reprezentuje liczbę przypadków, które zostały sklasyfikowane w konkretnej kombinacji prawdziwej klasy i przewidywanej klasy. W ten sposób macierz błędów dostarcza szczegółowych informacji o wynikach klasyfikacji, pozwalając na analizę różnych miar oceny wydajności modelu (Géron, 2020; Harrison, 2020).

Tab. 5 Macierz błędów dla klasyfikatora binarnego (Géron, 2020, zmodyfikowano)

		Rzeczywistość	
		1	0
Prognoza	Klasa		
	1	Prawdziwie pozytywne TP (<i>ang. True Positive</i>)	Fałszywie pozytywne FP (<i>ang. False Positive</i>)
	0	Fałszywie negatywne FN (<i>ang. False Negative</i>)	Prawdziwie negatywne TN (<i>ang. True Negative</i>)

gdzie: TP - oznacza liczbę przypadków, w których model poprawnie przewidział klasę 1
FN - oznacza liczbę przypadków, w których model błędnie przewidział klasę 0 zamiast klasy 1
FP - oznacza liczbę przypadków, w których model błędnie przewidział klasę 1 zamiast klasy 0
TN - oznacza liczbę przypadków, w których model poprawnie przewidział klasę 0

Jedną z miar używaną do oceny wydajności klasyfikatora jest **dokładność ACC** (*ang. Accuracy*), opisana wzorem (1). Dokładność ACC mierzy proporcję poprawnie sklasyfikowanych przypadków w stosunku do ogólnej liczby przypadków w zbiorze testowym. Jest to jedna z najpopularniejszych miar oceny klasyfikacji i wyrażana jest jako wartość procentowa (Harrison, 2020).

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

Dokładność (ACC) daje ogólny pogląd na skuteczność klasyfikatora w poprawnym przewidywaniu obu klas. Im wyższa dokładność ACC, tym lepsza dokładność klasyfikatora (Harrison, 2020).

Obliczanie wyłącznie parametru ACC w przypadku, gdy zbiór danych jest nie zrównoważony (tzn. występuje różna liczba przypadków w różnych klasach), nie jest odpowiednie, gdyż klasyfikator może być skuteczny wyłącznie dla bardzo licznej klasy. W przygotowanym zbiorze danych występuje nierówność klas: 125 próbek azbestu oraz 310 próbek innych pokryw dachowych. W związku z tym należy rozważyć dodatkowe miary, takie jak czułość, precyzja oraz parametr F1 (Géron, 2020; Harrison, 2020).

Precyzja (ang. *Precision, positive predictive value*) mierzy, jak wiele zidentyfikowanych przez klasyfikator pozytywnych wyników (klasa 1) jest faktycznie poprawnych. Wzór (2) przedstawia matematyczny zapis omawianej miary. Im wyższa precyzja, tym mniejsze jest ryzyko fałszywie pozytywnych wyników, co oznacza, że klasyfikator jest bardziej dokładny w identyfikowaniu pozytywnych przypadków (klasy 1) (Géron, 2020; Harrison, 2020).

$$Precyzja = \frac{TP}{TP + FP} \quad (2)$$

Czułość (ang. *Recall, true positive rate*) jest miarą zdolności klasyfikatora do poprawnego zidentyfikowania wszystkich pozytywnych przypadków (klasy 1). Wyraża on, jaka część prawdziwie pozytywnych przypadków została poprawnie wykryta przez klasyfikator. Wyznaczyć można ją stosując wzór (3) (Géron, 2020, Harrison, 2020).

$$Czułość = \frac{TP}{TP + FN} \quad (3)$$

W celu oceny skuteczności klasyfikatorów w opisywanym przypadku, podjęto decyzję o uwzględnieniu kompromisu między precyzją a czułością dla klasy dachów azbestowych. Bowiem wysoka precyzja wiąże się z większą dokładnością przewidywania pozytywnych przypadków, ale może prowadzić do pominięcia niektórych (lub nawet większej części) prawdziwie pozytywnych przypadków, co oznacza niską czułość. Z kolei wysoka czułość oznacza, że większość prawdziwie pozytywnych przypadków zostaje poprawnie wykryta, ale może prowadzić do większej liczby fałszywie pozytywnych przypadków, co oznacza niską precyzję (Géron, 2020).

Bardziej zrównoważoną oceną klasyfikatora binarnego jest **miara F1** (*ang. F1-score*). Wartość F1 łączy czułość i precyzję w pojedynczą miarę, która pozwala na lepszą ocenę wydajności klasyfikatora binarnego, uwzględniając zarówno poprawność przewidywanych pozytywnych przypadków, jak i ich kompletność. Sposób obliczenia F1 prezentuje wzór (4) (Géron, 2020; Harrison, 2020).

$$F1 = 2 \cdot \frac{\text{Precyzja} \cdot \text{Czułość}}{\text{Precyzja} + \text{Czułość}} = \frac{2TP}{2TP + FP + FN} \quad (4)$$

Im wartość F1 jest bliższa 1, tym algorytm klasyfikujący jest bardziej skuteczny. W najlepszym przypadku F1 przyjmuje wartość 1, kiedy mamy do czynienia z idealną czułością i precyzją (Géron, 2020).

W opisanych powyżej miarach oceny dokładności klasyfikacji korzystano z klasy pozytywnej (1). Niemniej jednak trzeba pamiętać, że miary czułości, precyzji oraz F1 można wyznaczać również dla klasy negatywnej (0) (Géron, 2020). Prowadzone badania dotyczyły klasyfikacji dachów azbestowych, którym przypisano etykietę pozytywną (1). W związku z tym podczas oceny skuteczności klasyfikacji uwzględniano miarę F1 dla klasy pozytywnej, czyli dla dachów azbestowych.

3.7 NOWE KIERUNKI BADAŃ

Przegląd literatury pozwolił zebrać i porównać prace badawcze prowadzone nad wykrywaniem azbestowo – cementowych pokryć dachowych. Zestawienie badań wykonano pod względem danych źródłowych oraz metod klasyfikacji. Z uwagi na analizę różnych danych teledetekcyjnych, wybór niejednorodnych metod opracowania oraz niejednakowe parametry oceny dokładności, nie dokonano bezpośredniego porównania wartości wyników oceny dokładności. **Po przeprowadzonej analizie nie można wskazać jednej najlepszej metody, która byłaby jedynym i niezawodnym rozwiązaniem problemu identyfikacji pokryć azbestowych.**

Według Abbasi i in. (2022) można wskazać sześć głównych problemów, które można napotkać korzystając z metod wskazanych w badaniach opisanych w literaturze. Należą do nich:

1. Niska rozdzielczość przestrzenna obrazów hiperspektralnych, klasyfikowanych metodami pikselowymi w początkowych badaniach w tym temacie.
2. Wielość wymiarów danych hiperspektralnych, prowadzący do mieszania się odpowiedzi spektralnej dla różnych obiektów i materiałów.
3. Niejednorodne wzorce materiałów powierzchniowych na obrazach wielospektralnych o wysokiej rozdzielczości przestrzennej, przy klasyfikacji tradycyjnymi metodami pikselowymi.
4. Nieoptymalne parametry segmentacji w klasyfikacji obiektowej powodujące niską dokładność klasyfikacji.
5. Niepełne dane treningowe przy metodach nadzorowanych.
6. Błędy reguł w klasyfikacji obiektowej, wynikające z subiektywnych oceny eksperta.

Aktualnie dostępne obrazy lotnicze i satelitarne charakteryzują się coraz wyższą rozdzielczością przestrzenną. Wraz ze wzrostem rozdzielczości przestrzennej, odpowiedzi spektralne pojedynczych obiektów, np. dachów, stają się bardziej zróżnicowane. Niejednorodność jasności pikseli na obrazach, może powodować problemy przy klasyfikacji, szczególnie pikselowej. Dotychczas zaproponowane rozwiązania, oparte na sztucznej inteligencji, osiągają zadowalający poziom skuteczności w rozwiązywaniu tego problemu (Raczko i in., 2022; Yu i in., 2022; Hikuwai i in., 2023). Należy jednak zwrócić

uwagę, że przygotowanie dobrego modelu i wytrenowanie go na odpowiednio dużej liczbie próbek treningowych, jest pracochłonne i czasochłonne.

Modele korzystające z głębokiego uczenia uwzględniają konwencjonalne podejście do wykrywania obiektów, które opiera się na wartości jasności piksela oraz wykorzystaniu dodatkowych cech, takich jak tekstura.

Można postawić pytanie, czy możliwe jest **wskazanie metody klasyfikacji dachów azbestowych**, poza głębokim uczeniem, **która będzie niewrażliwa na zróżnicowanie jasności pikseli dla pojedynczych obiektów na wysokorozdzielczych obrazach?** W związku z tym, w prezentowanej pracy zaproponowano rozwiązanie, które korzysta z filtra zamieniającego jasność pikseli na uśrednioną długość fali. Zadaniem filtra jest wzmocnienie informacji spektralnej, tak aby proste klasyfikatory uczenia maszynowego mogły efektywnie odróżnić pokrycie azbestowe od innych. Manualny dobór parametrów filtra jest zadaniem trudnym. W tym celu, zasugerowano wykorzystanie sztucznej inteligencji, poprzez zastosowanie algorytmu genetycznego, który powinien wskazać skuteczny filtr.

4 OBSZAR BADAŃ

Badania przeprowadzono dla obszaru wsi Krzczonów, położonej w województwie małopolskim, w powiecie myślenickim, w gminie Tokarnia (Rys. 4; Rys. 5). Obszar miejscowości rozciąga się wzdłuż potoku Krzczonówka. Według Narodowego Spisu Powszechnego Ludności i Mieszkań z 2021 roku liczba ludności we wsi Krzczonów wynosi 2 426. Zabudowania znajdują się w dolinie potoku, wzdłuż drogi łączącej Pcim ze Skomielną Czarną oraz podchodzą na północne zbocza doliny.

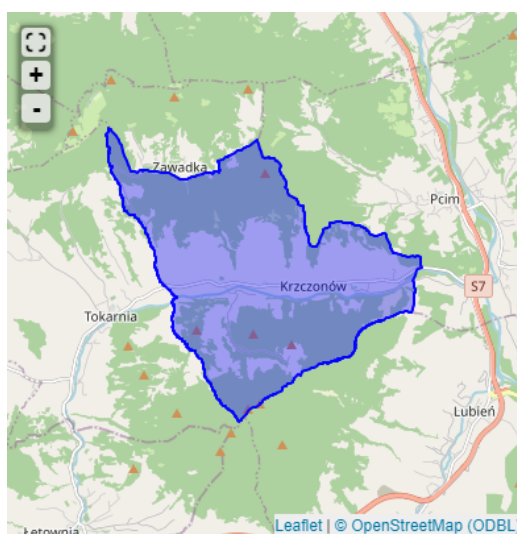
(A)



(B)



Rys. 4 Lokalizacja obszaru badań: (A) w granicach Rzeczypospolitej Polski; (B) w granicach województwa małopolskiego (opracowanie własne)



Rys. 5 Obszar wsi Krzczonów
(https://www.polskawliczbach.pl/wies_Krzczonek_malopolskie, dostęp: maj 2023)

5 DANE

W rozdziale opisano przebieg pozyskania informacji o pokryciach dachowych. Następnie opisano teledetekcyjne dane źródłowe wykorzystywane w procesie klasyfikacji azbestowo - cementowych pokryć dachowych.

5.1 DANE IN SITU

Informacje na temat rodzajów pokrycia dachowego na badanym obszarze uzyskano w trakcie prac terenowych. Prace te były prowadzone w formie wywiadu terenowego, w czasie którego na przygotowanych obrazach badanego obszaru, zaznaczano etykiety pokryć dachowych. Litera A oznaczała dach pokryty azbestem, litera N – inny rodzaj pokrycia dachowego. Na rysunkach (Rys. 6; Rys. 7) przedstawiono przykłady dachów klasyfikowanych jako azbest oraz inne pokrycia dachowe. Wyjazd w teren zrealizowano w listopadzie 2017 roku. Wybrany termin wiązał się z naturalnym zanikiem wegetacji roślin. Drzewa przesłaniające widoczność budynków oraz ich dachów zrzuciły liście na zimę.

W wyniku przeprowadzonych obserwacji zapisano informacje o 125 dachach azbestowych oraz 310 innych pokryciach dachowych dla miejscowości Krzczonów. Pozyskano również informacje na temat pokryć dachowych dla fragmentów miejscowości przyległych: Tokarnia, Pcim, Lubień, zarejestrowanych na obrazie satelitarnym poddawanych klasyfikacji (patrz rozdział 5.2). Zapisano informacje o 26 dachach azbestowych oraz 44 innych pokryciach dachowych. Dodatkowe próbki wykorzystano do testowania klasyfikatorów. Na przygotowanych materiałach, zaznaczano etykiety na dachach, które były zobrazowane na obrazie satelitarnym w sposób jednoznaczny. Pomijano małe budynki gospodarcze, przybudówki, gdyż wartość jasności pojedynczych pikseli na obrazie satelitarnym, nie była odpowiednią w całości od jednego rodzaju pokrycia dachowego.

W czasie pobytu w terenie zauważono, że dachy azbestowe w większości znajdują się już tylko na budynkach gospodarczych. W zdecydowanej większości na budynkach mieszkalnych wybudowanych pod koniec XX wieku zmieniono pokrycia dachowe. Może to wynikać z dotacji unijnych dostępnych w gminach przeznaczonych właśnie na ten cel.



Rys. 6 Azbestowe pokrycia dachowe we wsi Krzczonów
(*Google Street View*)



Rys. 7 Inne pokrycia dachowe we wsi Krzczonów: (A) dachówka cementowa, (B) blachodachówka, (C) dachówka cementowa (Google Street View)

5.2 DANE SATELITARNE

W prezentowanej pracy badaniom poddawano wielospektralny obraz satelitarny wykonany przez satelitę WorldView-3. Sensor ten wyprodukowany został przez firmę DigitalGlobe, która jest jednym z wiodących dostawców danych satelitarnych. Główną cechą wyróżniającą WorldView-3 jest jego zdolność do zbierania obrazów z niezwykle wysoką rozdzielczością. Umieszczony na wysokości 617 km, WorldView-3 pozyskuje obrazy panchromatyczne z rozdzielczością przestrzenną 0,31 m, osiem obrazów w zakresie widzialnym i bliskiej podczerwieni (VNIR) z rozdzielczością przestrzenną 1,24 m oraz osiem krótkofalowych pasm podczerwieni (SWIR) o rozdzielczości przestrzennej 3,70 m. Wartość jasności pojedynczego piksela zapisywana jest na 16 bitach. Dostępny do analizy obraz satelitarny został zarejestrowany w dniu 19 marca 2015 roku i składał się z obrazu panchromatycznego oraz ośmiu kanałów w zakresie widzialnym i bliskiej podczerwieni, bez kanałów podczerwieni krótkofalowej (Rys. 8; Tab. 6). Zakup tego obrazu został zrealizowany przy pomocy grantu dziekańskiego o numerze 15.11.150.418, na Wydziale Geodezji Górniczej i Inżynierii Środowiska AGH w Krakowie.

Zarejestrowane zakresy długości fal elektromagnetycznych na obrazie WorldView-3 nie pozwalają jednoznacznie odróżnić azbestu od innych materiałów wyłącznie na podstawie charakterystycznych cech odbicia w pojedynczych kanałach spektralnych. W związku z tym, w procesie klasyfikacji uwzględniono osiem kanałów spektralnych z rozdzielczością przestrzenną 1,24m. Obraz panchromatyczny służył wyłącznie do wyostrzenia pozostałych kanałów spektralnych w czasie manualnego oznaczania pokryć dachowych na obrazie wielospektralnym.

Tab. 6 Parametry kanałów spektralnych satelity WorldView-3 (www.digitalglobe.com, dostęp: maj 2023)

Numer kanału spektralnego	Nazwa kanału spektralnego	Rozdzielczość przestrzenna [m]	Zakres długości rejestrowanej fali [nm]
1	Coastal blue	1,24	400 - 450
2	Blue		450 - 510
3	Green		510 - 580
4	Yellow		585 - 625
5	Red		630 - 690
6	Red edge		705 - 745
7	Near-IR1		770 - 895
8	Near-IR2		860 - 1040
9	Panchromatic	0,31	450 - 800



Rys. 8 Obraz satelitarny WorldView-3

6 METODYKA PRAC BADAWCZYCH

W rozdziale wskazano informacje na temat wykorzystywanego w trakcie obliczeń sprzętu komputerowego, języka programowania oraz oprogramowania. Następnie przybliżono sposób przygotowania i wstępnego przetwarzania danych. Kolejną część wyjaśnia sposób działania własnego algorytmu filtrującego dane na przykładowych wartościach. Obliczono statystyki przygotowanych danych wejściowych, wykorzystywanych do procesu klasyfikacji. Zwrócono uwagę na podobieństwo charakterystyk statystycznych dla wszystkich rodzajów pokryć dachowych. Na końcu przedstawiono trzy strategie wykonywanych analiz oraz sposób przetestowania otrzymanych macierzy filtrujących. Zaprezentowano schemat blokowy wykorzystywanych eksperymentów obliczeniowych w języku *Python*.

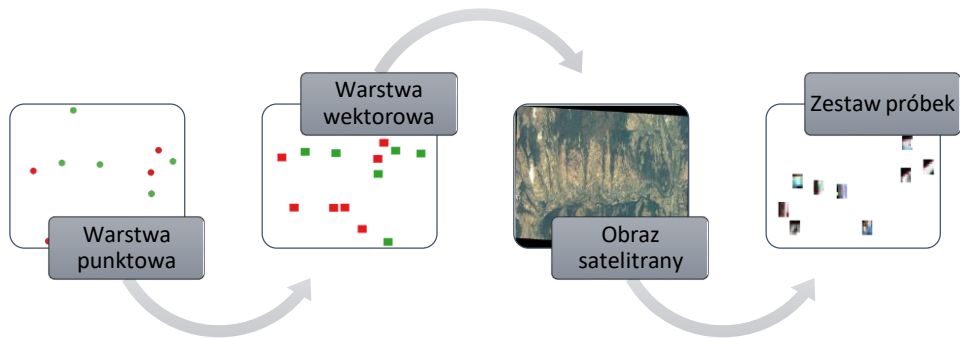
6.1 PRZYGOTOWANIE ZESTAWU DANYCH TRENUJĄCYCH, WALIDACYJNYCH ORAZ TESTOWYCH

Opracowanie danych wejściowych wykorzystywanych w klasyfikacji wykonano w wolnym i otwartym oprogramowaniu QGIS (<https://qgis.org/>). Zarys wykonanych czynności przedstawiono na rysunku (Rys. 9). W pierwszej kolejności utworzono warstwę punktową, reprezentującą środek dachu (Rys. 10). Zgodnie z pracami polowymi, warstwa punktowa dla wsi Krzczonów składała się z 435 elementów. Każdy punkt otrzymał atrybut określający rodzaj pokrycia dachowego. Etykieta *a* oznaczała dach azbestowy, etykieta *n* – inny rodzaj pokrycia dachowego. Kolejnym krokiem było utworzenie dwóch warstw wektorowych pozwalających na wycięcie próbek dachów o wymiarach 7x7 pikseli (8,68 x 8,68 m) oraz 14x14 pikseli (17,36 x 17,36 m) (Rys. 11). Warstwy wektorowe utworzono tak, aby wycinając fragment źródłowego obrazu nie dochodziło do dodatkowych przekształceń – brak powtórnego próbkowania jasności pikseli (resamplingu).

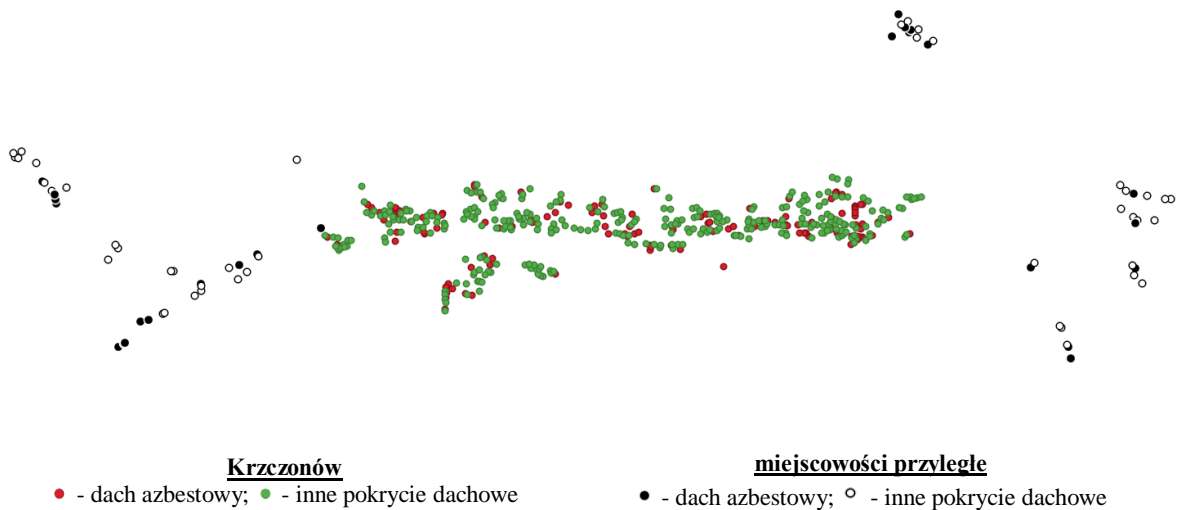
Efektem końcowym przygotowania danych było otrzymanie dwóch zestawów po 435 obrazów prezentujących pokrycia dachowe. Próbkę dachów w obu zestawach różniły się wymiarami obrazu. Warto zwrócić uwagę, że obrazy w zestawie 7x7 pikseli składały się z jednorodnej informacji. Każdy piksel reprezentował w większości przypadków jasność

wyłącznie pokrycia dachowego. W przypadku próbek 14x14 pikseli, na pojedynczym obrazie oprócz dachu zarejestrowano w dużej mierze otoczenie budynku.

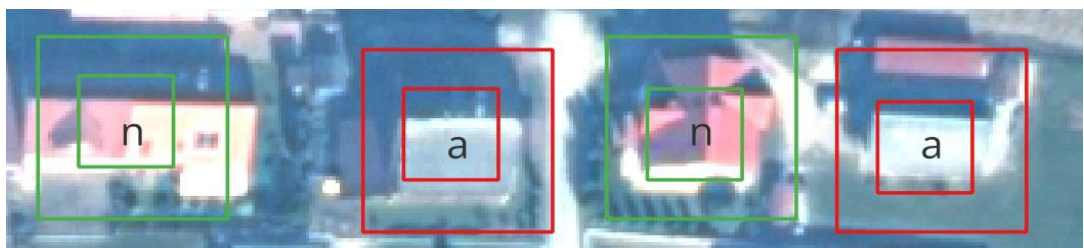
Utworzone dane dla wsi Krzczonów wykorzystywano jako dane trenujące oraz walidacyjne. W analogiczny sposób przygotowano dane testowe, zawierające próbki dachowe dla fragmentów miejscowości przyległych. Otrzymano dwa zestawy danych (7x7 pikseli, 14x14 pikseli) po 70 próbek pokryć dachowych.



Rys. 9 Schemat wstępnego opracowania danych (opracowanie własne)



Rys. 10 Warstwa punktowa przedstawiająca rozmieszczenie dachów (opracowanie własne)



Rys. 11 Wektory przygotowane do wycięcia fragmentów dachów na tle wyostrożonego obrazu satelitarnego WorldView-3
a – azbest; n – inne pokrycie dachowe
(opracowanie własne)

6.2 WSTĘPNE PRZETWORZENIE DANYCH

Przygotowane w QGIS zestawy danych zostały poddane wstępnemu przetworzeniu, tak aby mogły być wykorzystywane w dalszych badaniach. Na schemacie (Rys. 13) zaprezentowano wykonane przekształcenia w języku *Python*. W pierwszym etapie rozdzielano wszystkie wielospektralne obrazy dachów na pojedyncze kanały spektralne z wykorzystaniem biblioteki *gdal* (<https://gdal.org/>) i zapisano je w oddzielnych plikach w formacie *tif*. Rysunek (Rys. 12) przedstawia fragment wykorzystanego kodu *Python*. W następnej kolejności przetworzono pliki z formatu *tif* na macierze biblioteki *NumPy* (<https://numpy.org/>) oraz dokonano spłaszczenia tych macierzy do jednego wymiaru (Rys. 14). Następnie złączono wszystkie macierze *NumPY* dla pojedynczego kanału spektralnego w jedną macierz X_i . Dodatkowo utworzono zero - jedynkowe macierze y , w których zakodowano informacje o rodzaju próbki. Macierz dla próbek ze wsi Krzczonów składała się z 435 wierszy (liczba próbek) i jednej zero – jedynkowej kolumny. Dla fragmentów pozostałych miejscowości wymiar macierz y wynosił 70 wierszy oraz jedna zero – jedynkowa kolumna. Wartość 1 oznaczała pokrycie azbestowe, wartość 0 – inny rodzaj pokrycia dachowego. W każdej macierzy X_i oraz przypisanej jej macierzy y , jeden wiersz odpowiadał tej samej próbce pokrycia dachowego. Fragment kodu łączącego dane w macierze X_i oraz y przedstawiono na rysunku (Rys. 15).

```
# pobranie listy plików TIF z folderu input
tif_files = [f for f in os.listdir(input_folder) if f.endswith('.tif')]

# iteracja po wszystkich plikach TIF w folderze wejściowego
for tif_file in tif_files:

    # ścieżka do wejściowego pliku TIF
    input_file = os.path.join(input_folder, tif_file)

    # otwarcie pliku TIF
    input_data = gdal.Open(input_file)

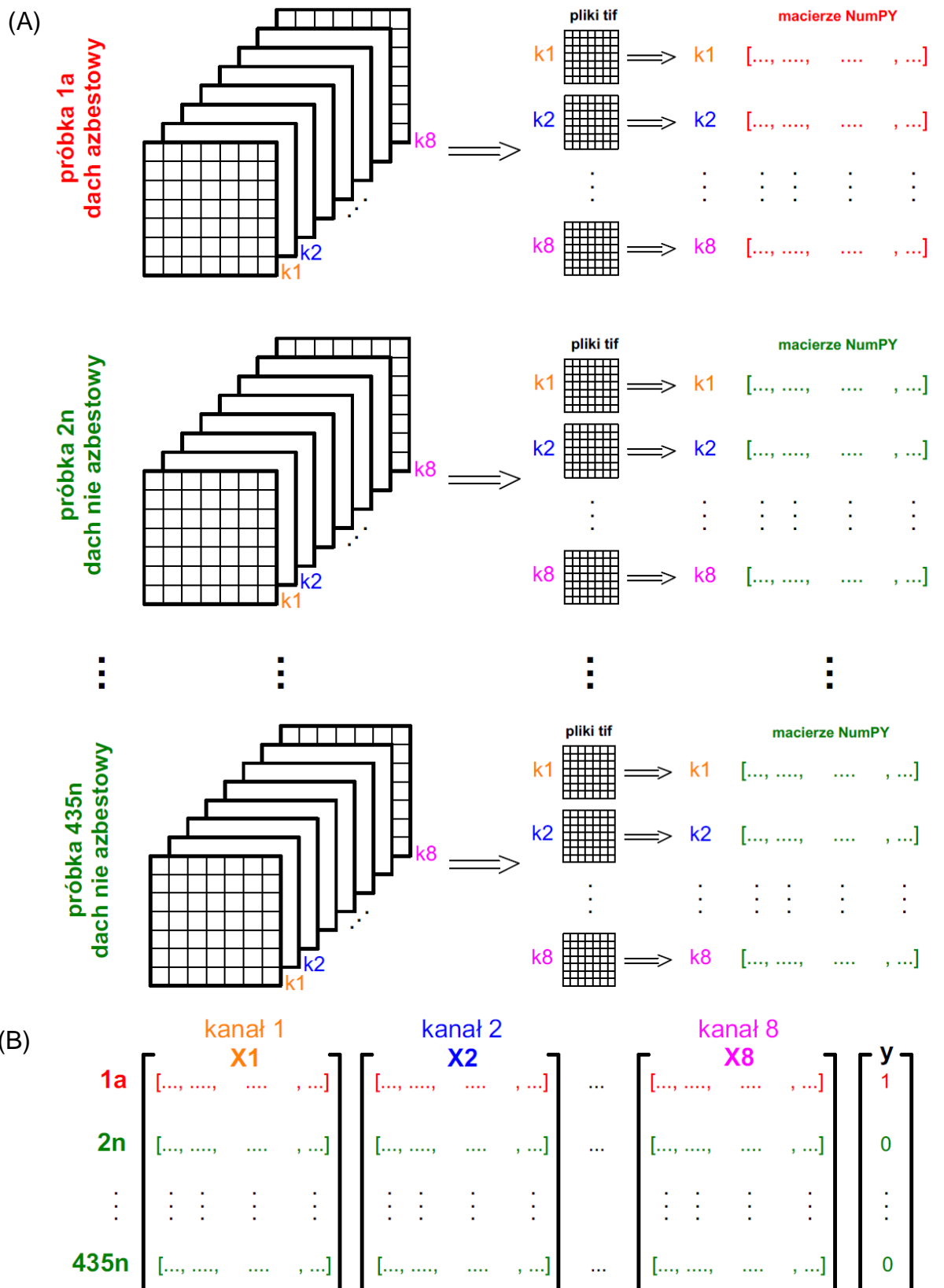
    # pobranie liczby kanałów
    num_bands = input_data.RasterCount

    # iteracja po wszystkich kanałach
    for i in range(num_bands):

        # pobranie i zapisanie pojedynczego kanału w oddzielnym pliku tif
        band = input_data.GetRasterBand(i + 1)
        output_file = os.path.join(output_folder, f'{tif_file[:-4]}_k{i+1}.tif')
        driver = gdal.GetDriverByName('GTiff')
        output_data = driver.Create(output_file, input_data.RasterXSize,
                                   input_data.RasterYSize, 1, band.DataType)
        output_data.SetGeoTransform(input_data.GetGeoTransform())
        output_data.SetProjection(input_data.GetProjection())
        output_data.GetRasterBand(1).WriteArray(band.ReadAsArray())

    # zamknięcie pliku TIF wejściowego
    input_data = None
    output_data = None
```

Rys. 12 Fragment kodu *Python* rozdzielającego obrazy wielospektralne na pojedyncze kanały (opracowanie własne)



Rys. 13 Schemat wstępnego przygotowania danych do procesu klasyfikacji (opracowanie własne)

```

## Przekształcenie tif w npy

# pobranie listy plików TIF z folderu
tif_files = [f for f in os.listdir(input_folder) if f.endswith('.tif')]

# iteracja po wszystkich plikach TIF w folderze
for tif_file in tif_files:

    # ścieżka do pliku TIF wejściowego
    input_file = os.path.join(input_folder, tif_file)

    # otwarcie pliku TIF
    image = imageio.imread(input_file)
    print(tif_file)

    # konwersja na tablicę NumPy
    image_array = np.array(image)

    # Zapisywanie macierzy do pliku tekstowego
    np.save(os.path.join(output_folder, tif_file[:-4]), image_array)

    # zamknięcie pliku TIF wejściowego
    input_data = None

#####

## Spłaszczenie macierzy npy

#iteracja po wszystkich plikach w folderze
for npy_file in npy_files:
    f = np.load(os.path.join(input_folder, npy_file))
    f_flatten = f.reshape(-1)
    np.save(os.path.join(output_folder, npy_file), f_flatten)

```

Rys. 14 Fragment kodu *Python* przekształcającego format danych tif na macierz npy oraz spłaszczającego macierzy npy (opracowanie własne)

```

# pobranie listy plików npy z folderu input, określenie kanatu spektralnego ki
numpies = [f for f in os.listdir(input_folder) if f.endswith('.npy') and 'k8' in f]
nb = len(numpies)

for num in numpies:
    shape = np.load(os.path.join(input_folder, num)).shape

    X = np.zeros((nb, shape[0]))

    y = np.zeros((nb))

# iteracja po wszystkich plikach numpy na liście numpies
i = 0
for f in numpies:
    row = np.load(os.path.join(input_folder, f))
    X[i] = row
    y[i] = 1 if 'a' in f else 0
    i += 1

np.save(os.path.join(output_folder, 'x8'), X)
np.save(os.path.join(output_folder, 'y'), y)

```

Rys. 15 Fragment kodu *Python* tworzącego macierze X_i oraz y (opracowanie własne)

6.3 ALGORYTM FILTROWANIA KANAŁÓW SPEKTRALNYCH

W przypadku klasyfikacji obrazów, gdzie pojedyncza próbka składa się z kilku osobnych warstw, może pojawić się pytanie, w jaki sposób połączyć te dane. Dla klasyfikatora wartość jasności piksela w każdym kanale spektralnym traktowane są jako liczby z pewnego zakresu i nic szczególnego je nie wyróżnia. W celu rozróżnienia informacji płynącej z różnych kanałów spektralnych, zaproponowano autorski algorytm filtrowania danych. Algorytm ten przekształca wartości jasności pikseli na uśrednione długości fal elektromagnetycznych (Tab. 7), w ustalonym dla każdego kanału zakresie jasności. Jeżeli dana jasność nie znajduje się w wyznaczonym przedziale, informacja płynąca z tego piksela jest zamieniana na wartość zero. W kolejnym podrozdziale zaprezentowano przykład działania algorytmu na sztucznie przygotowanym zestawie danych.

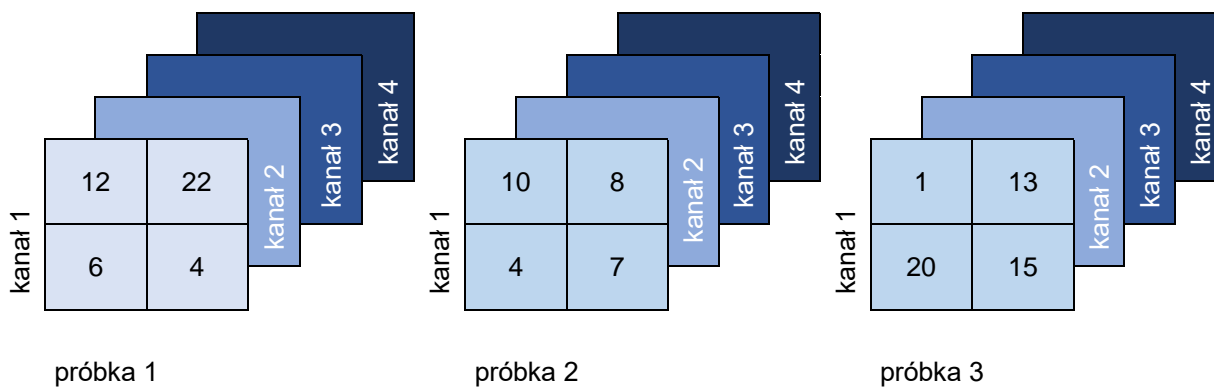
Tab. 7 Średnie długości fali reprezentujące poszczególne kanały spektralne (opracowanie własne)

Numer kanału	1	2	3	4	5	6	7	8
Średnia długość fali [nm]	425	480	545	605	660	725	832	950

6.3.1 Przykład funkcjonalności opracowanego w pracy algorytmu filtrowania kanałów spektralnych

Dla zaprezentowania opracowanego algorytmu filtrowania warstw spektralnych zostaną wykorzystane pewne uproszczone dane opracowane na rzecz tego przykładu. Załóżmy zatem, że pewien zestaw danych wejściowych składa się z trzech próbek obrazów o wymiarze 2 x 2 piksele. Każda próbka posiada pewne cztery kanały spektralne o kolejnych numerach 1, 2, 3, 4. Każdy piksel ma przypisaną wartość liczbową reprezentującą jasność. Dane takie możemy przedstawić za pomocą schematu przedstawionego na rysunku (Rys. 16). Na początku poszczególne obrazy poddane zostają wstępnemu przetwarzaniu polegającemu na wykonaniu kolejnych etapów:

- Etap 1: rozdzielenie obrazu multispektralnego na pojedyncze kanały.
- Etap 2: spłaszczenie pojedynczego obrazu z macierzy 2 x 2 do jednowymiarowej macierzy 1 x 4.
- Etap 3: utworzenie macierzy zbiorczych X_i dla poszczególnych kanałów spektralnych ($i = 1, 2, 3, 4$), gdzie dany wiersz zawiera wartości tej samej próbki we wszystkich macierzach (Tab. 8).



Rys. 16 Schemat danych wejściowych do przykładu obliczeniowego algorytmu filtrowania (opracowanie własne)

Tab. 8 Macierze X_i oraz macierz y dla przykładowych danych (opracowanie własne)

	X1				X2				X3				X4				y
próbka 1	12	22	6	4	1	33	20	3	60	63	90	55	23	26	56	70	1
próbka 2	10	8	4	7	19	40	17	8	56	53	54	43	44	32	1	23	0
próbka 3	1	13	20	15	30	35	2	60	99	57	51	47	65	58	1	32	0

Dodatkowo tworzona jest zero - jedynkowa macierz y , o wymiarze liczby próbek (3×1), która nie podlega modyfikacji algorytmem filtrowania. W macierzy tej zakodowana jest informacja o znanej klasyfikacji każdej próbki: 1 oznacza pokrycie azbestowe, 0 – inny rodzaj pokrycia dachowego. Należy zauważyć, że wiersze z macierzy X_i oraz y dotyczą tej samej próbki.

Wartości jasności w otrzymanych macierzach X_i w przypadku danych rzeczywistych z obrazu satelitarne są dodatkowo normalizowane, tak aby mieściły się w przedziale $[0, 1]$. Każda wartość z macierzy X_i jest dzielona przez element o maksymalnej wartości z tej macierzy. Działanie to ma na celu wyeliminowanie różnych zakresów wartości jasności w poszczególnych kanałach. Ze względu na przejrzystość opisu, podczas omawiania przykładu obliczeniowego działania algorytmu filtrowania pominięto normalizację. Dalsze rozważania będą prowadzone na wartościach rzeczywistych jasności pikseli, które podczas analiz są normalizowane.

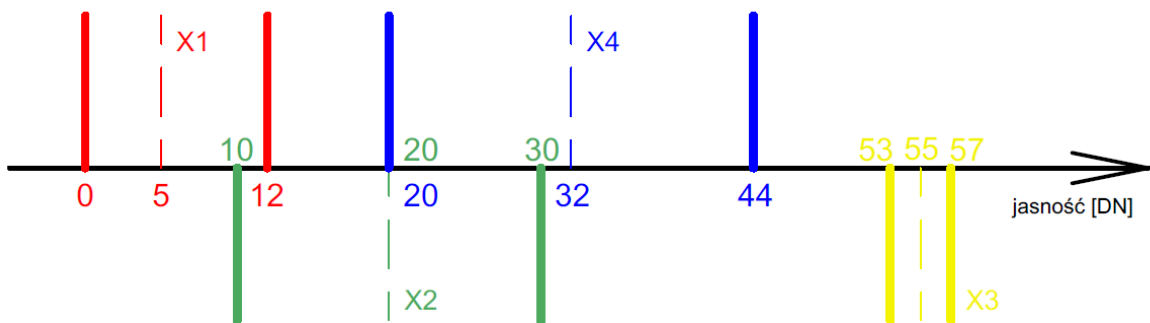
W wyniku procesu filtrowania macierze X_i przekształcane są w jedną macierz X , która wykorzystywana jest do klasyfikowania zestawu próbek z wykorzystaniem uczenia maszynowego. Przebieg filtrowania można podzielić na dwie fazy, opisane poniżej.

a) Faza 1: Ustalenie wartości dla filtra w postaci pewnej macierzy F.

Macierz F składa się z par liczb, które odnoszą się do poszczególnych kanałów spektralnych reprezentowanych przez macierze X_i (Rys. 17). W związku z tym, wymiar macierzy F w analizowanym przykładzie wynosi 1×8 . Pojedyncza para liczb określa przedział istotnych wartości dla danego kanału spektralnego podczas klasyfikacji próbek. Pierwsza liczba wyznacza środek przedziału, natomiast druga liczba definiuje odchylenie w jedną i w drugą stronę. Na rysunku (Rys. 18) przedstawiono zakresy w omawianym przykładzie. Należy dodać, że jeżeli dolna granica przedziału wypadnie poniżej zera, zamienia się ją na wartość zero, gdyż jasności pikseli są zawsze wartościami nieujemnymi.

$$\begin{bmatrix} X1 & X2 & X3 & X4 \\ (5, 7) & (20, 10) & (55, 2) & (32, 12) \end{bmatrix}$$

Rys. 17 Macierz F wykorzystywana w przykładzie obliczeniowym (opracowanie własne)



Rys. 18 Zakresy przedziałów filtrowania danych przedstawione na osi wzrostu wartości jasności pikseli (opracowanie własne)

Zastosowanie filtra polega na sprawdzeniu czy dana wartość z poszczególnych macierzy X_i mieści się w zdefiniowanym zakresie w macierzy F dla danego kanału spektralnego. W przypadku gdy dana wartość jasności z poszczególnych macierzy X_i spełnia opisany warunek, następuje zamiana tej wartości na średnią długość fali elektromagnetycznej rejestrowanej przez sensor dla tego kanału spektralnego (Tab. 9). W przeciwnym przypadku wartość jasności zastępuje się zerem. W tabeli (Tab. 10) zaprezentowano macierze X_i po przeprowadzeniu filtrowania.

Tab. 9 Zestawienie średniej długości fali elektromagnetycznej dla poszczególnych kanałów spektralnych (opracowanie własne)

Kanał spektralny	Oznaczenie macierzy X_i	Średnia długość fali [nm]
1	X_1	425
2	X_2	480
3	X_3	545
4	X_3	605

Tab. 10 Macierze X_i przefiltrowane macierzą F (opracowanie własne)

X1				X2				X3				X4			
425	0	425	425	0	0	480	0	0	0	0	545	605	605	0	0
425	425	425	425	480	0	480	0	545	545	545	0	605	605	0	605
425	0	0	0	480	0	0	0	0	545	0	0	0	0	0	605

b) Faza 2: Utworzenie macierzy wyjściowej X.

Drugim etapem przekształcenia danych jest połączenie wszystkich przefiltrowanych macierzy X_i w jedną macierz wyjściową X. W tym celu poszczególne elementy zbioru macierzy X_i są sumowane i dzielone przez liczbę elementów niezerowych, zgodnie ze wzorem (5). Dla lepszego zobrazowania wykonywanych obliczeń, w tabeli (Tab. 11) przedstawiono macierz NO, która zawiera zliczone niezerowe wartości w zestawie macierzy X_i . Końcowym wynikiem przeprowadzonych obliczeń wg wzoru (5) jest poszukiwana macierz X (Tab. 12). Podczas obliczeń może wystąpić szczególny przypadek, gdy dla danego elementu macierzy X, wszystkie odpowiadające mu elementy w zestawie macierzy X_i przyjmują wartość zero. W związku z tym, parametr NO również przyjmuje wartość zero, a wzór (5) jest niemożliwy do obliczenia. W tej sytuacji danemu elementowi przypisywana jest również wartość zero .

$$X_{a,b} = \frac{\sum_{i=1}^4 X_i^{a,b}}{NO} \quad (5)$$

gdzie: $a = 1, 2, 3$ (ilość próbek)
 $b = 1, 2, 3, 4$ (ilość kanałów spektralnych)
 NO – liczba niezerowych elementów $X_i^{a,b}$

Tab. 11 Macierz NO: zliczone niezerowe wartości w zestawie macierzy X_i (opracowanie własne)

próbka 1	2	1	2	2
próbka 2	4	3	3	2
próbka 3	2	1	0	1

Tab. 12 Macierz X dla przykładowych danych (opracowanie własne)

X

515	605	453	485
514	525	483	515
453	545	0	605

Otrzymana macierz wyjściowa zawiera zatem informację o uśrednionej długości fali dla każdego piksela przetwarzanego obrazu z uwzględnieniem tych pikseli, dla których wartość ich jasności zmieściła się w ustalonej przez filtr szerokości ścieżki przenikania informacji do macierzy wyjściowej X. Otrzymana macierz wyjściowa zawiera zatem informację o uśrednionej długości fali dla każdego piksela przetwarzanego obrazu z uwzględnieniem tych pikseli, dla których wartość ich jasności zmieściła się w ustalonej przez filtr szerokości ścieżki przenikania informacji do macierzy wyjściowej X.

6.4 ANALIZA STATYSTYCZNA KLASYFIKOWANYCH DANYCH

Zebrane dane treningowe (dla wsi Krzczonów) dla zbiorów danych 7x7 pikseli oraz 14x14 pikseli, w formie macierzy X, poddano podstawowej analizie statystycznej. Wyznaczono dla każdego kanału spektralnego następujące parametry statystyczne: minimum, maksimum, wartość średnią, medianę, odchylenie standardowe oraz współczynnik zmienności (Tab. 13; Tab. 14), na podstawie których dokonano porównania jasności pikseli dla azbestowych i innych pokryć dachowych.

Minimalna wartość jasności piksela dla wszystkich próbek wynosi 1, natomiast maksymalna 2047. Analizując średnie wartości jasności pikseli w poszczególnych kanałach spektralnych można zauważyć, zbliżone wartości w kanałach od 1 do 3, natomiast większe różnice występują w kanałach od 4 do 7 (Tab. 13; Tab. 14).

Tab. 13 Zestawienie podstawowych wartości statystycznych dla zestawu danych 7x7 pikseli (opracowanie własne)

Rodzaj pokrycia	Numer kanału spektralnego	Minimum	Maksimum	Średnia	Mediana	Odchylenie standardowe	Współczynnik zmienności [%]
azbestowe	1	177	395	209,76	208	17,34	8,27
inne		167	650	211,83	203	37,40	17,66
azbestowe	2	94	332	132,40	131	19,78	14,94
inne		84	640	132,96	123	40,67	30,59
azbestowe	3	58	475	148,28	145	34,99	23,60
inne		65	1024	149,38	134	65,30	43,71
azbestowe	4	89	759	208,97	201	62,27	29,80
inne		87	1860	241,82	211	122,39	50,61
azbestowe	5	41	519	135,56	130	47,76	35,23
inne		42	1472	172,12	142	103,35	60,05
azbestowe	6	65	701	202,11	191	70,80	35,03
inne		64	2047	260,24	208	160,77	61,78
azbestowe	7	43	580	185,22	177	72,21	38,98
inne		46	1766	218,42	175	136,19	62,35
azbestowe	8	44	588	198,86	189	79,33	39,89
inne		45	1780	221,92	181	139,43	62,83

Tab. 14 Zestawienie podstawowych wartości statystycznych dla zestawu danych 14x14 pikseli (opracowanie własne)

Rodzaj pokrycia	Numer kanału spektralnego	Minimum	Maksimum	Średnia	Mediana	Odchylenie standardowe	Współczynnik zmienności [%]
azbestowe	1	92	733	201,98	197	19,72	9,76
inne		167	793	205,02	196	32,86	16,03
azbestowe	2	85	469	124,88	120	22,07	17,67
inne		76	739	127,32	118	36,04	28,31
azbestowe	3	30	1299	138,14	132	39,39	28,51
inne		34	1086	142,57	130	59,15	41,49
azbestowe	4	1	1221	190,75	181	68,50	35,91
inne		87	1885	215,15	187	111,31	51,74
azbestowe	5	1	1209	123,87	116	54,34	43,87
inne		9	1526	148,24	122	92,18	62,18
azbestowe	6	1	1454	199,80	190	86,92	43,50
inne		57	2047	236,28	205	140,30	59,38
azbestowe	7	13	1009	190,68	179	93,33	48,95
inne		24	2003	213,74	183	128,50	60,12
azbestowe	8	1	1295	197,64	186	100,42	50,81
inne		30	2002	214,87	185	133,49	62,13

Kolejnym wyznaczonym wskaźnikiem była mediana, dzieląca zbiór danych w poszczególnych kanałach spektralnych na dwie równe części. Wartość mediany dla wszystkich kanałów w obu zestawach danych mieści się w przedziale od 116 do 211. Z tej informacji można wyciągnąć wniosek, iż 50% wszystkich jasności pikseli mieści w wartości do jasności 200. Im bardziej jasności przyjmują wartości zbliżone do siebie dla różnych pokryć dachowych, tym bardziej kłopotliwe będzie rozróżnienie ich w procesie klasyfikacji.

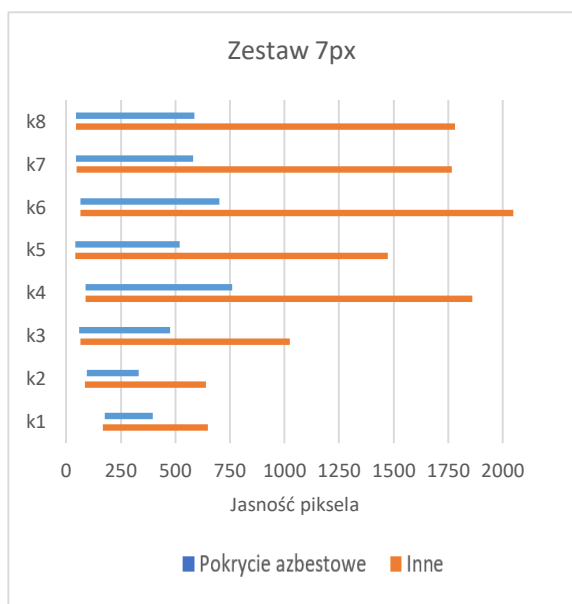
Odchylenie standardowe przedstawia rozrzut danych wokół średniej wartości. Im większe odchylenie standardowe, tym większy rozrzut danych, co wskazuje na większą zmienność i rozbieżność między poszczególnymi wartościami w danym kanale spektralnym. W prezentowanych wynikach najmniejsze odchylenie standardowe uzyskano dla 1 i 2 kanału spektralnego. Jednak jasności pikseli dla dachów azbestowych oraz innych pokryć dachowych są do siebie zbliżone.

Ostatnią wyznaczoną miarą jest współczynnik zmienności, określający względny rozrzutu danych. Jest to stosunek odchylenia standardowego do średniej wartości danych, wyrażony w procentach. Współczynnik zmienności pozwala porównać zmienność między zbiorami o różnych średnich wartościach, gdyż uwzględnia on skalę danych. Im większy współczynnik zmienności, tym większy względny rozrzut danych. W analizowanym przypadku, tylko dla 1 kanału spektralnego dla próbek azbestowych współczynnik zmienności nie przekracza 10%.

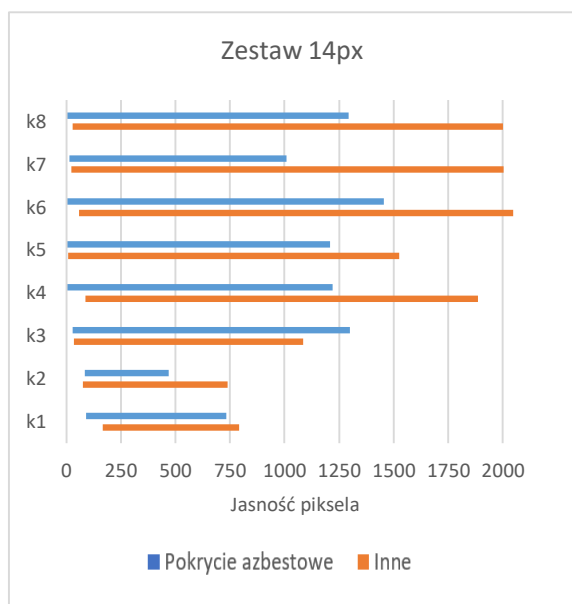
Na wykresie (Rys. 19) przedstawiono rozpiętość jasności (statystyczny rozstęp) pikseli dla próbek dachów azbestowych oraz innych pokryć dachowych w poszczególnych kanałach spektralnych.

Analizując wyłącznie zakres rozpiętości jasności pikseli w poszczególnych kanałach spektralnych można zauważyć występujące różnice (Rys. 19). Dla zestawu próbek 7x7 pikseli (Rys. 19A), jasności dla dachów azbestowych zlokalizowane są po lewej stronie wykresu, natomiast rozpiętość pozostałych dachów jest co najmniej dwukrotnie większa w porównaniu do azbestu. W przypadku próbek 14x14 pikseli (Rys. 19B) różnica w rozrzucie danych dla dachów azbestowych jest większa. Próbkę dachów przygotowano tak, aby na obrazach 7x7 pikseli przedstawić informacje jednorodną – wyłącznie odpowiedź spektralną pokrycia dachowego. Obrazy 14x14 pikseli zawierały odpowiedź spektralną pokrycia dachowego i jego najbliższego otoczenia. W związku z tak przygotowanymi danymi, po przeprowadzonych eksperymentach obliczeniowych, można sprawdzić, czy na dokładność klasyfikacji azbestowych pokryć dachowych ma wpływ zarejestrowana odpowiedź spektralna bliskiego otoczenia budynku.

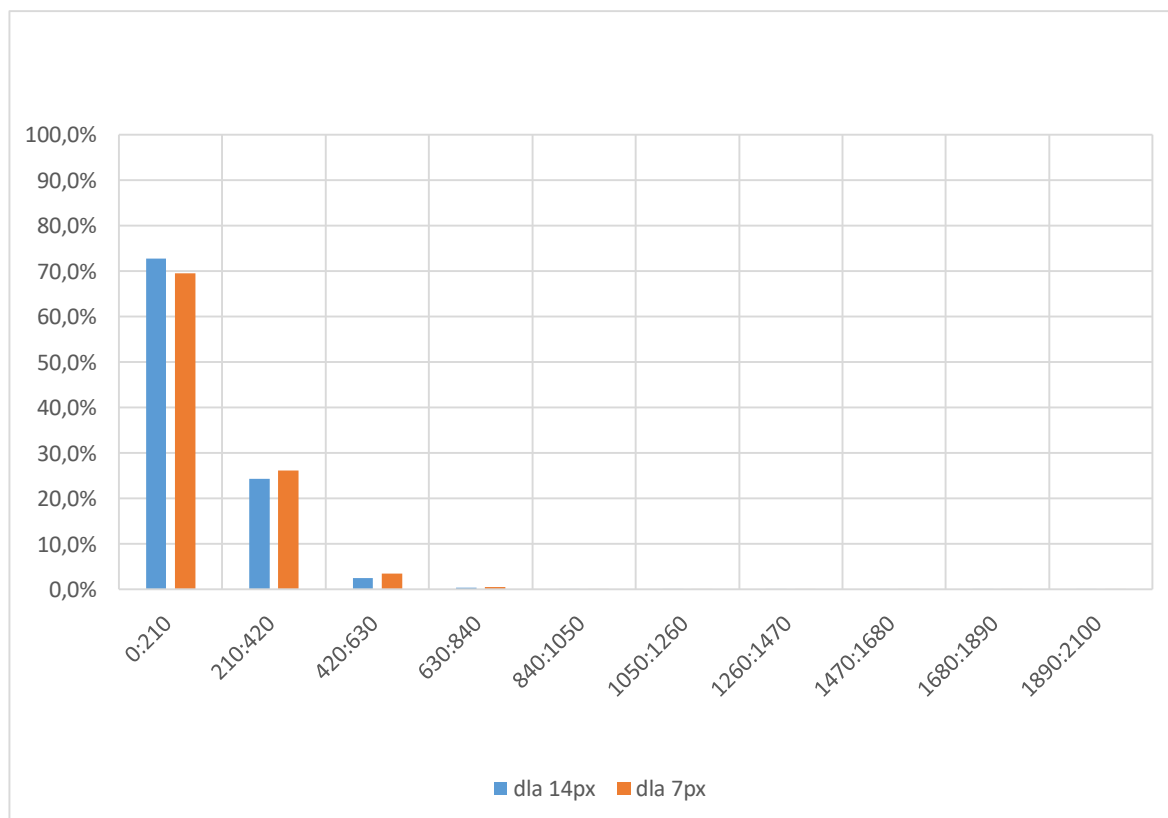
(A)



(B)



Rys. 19 Rozpiętość jasności pikseli dla próbek dachów azbestowych oraz innych pokryć dachowych w poszczególnych kanałach spektralnych (opracowanie własne)



Rys. 20 Rozkład jasności pikseli dla wszystkich kanałów spektralnych z rozróżnieniem zestawów próbek (opracowanie własne)

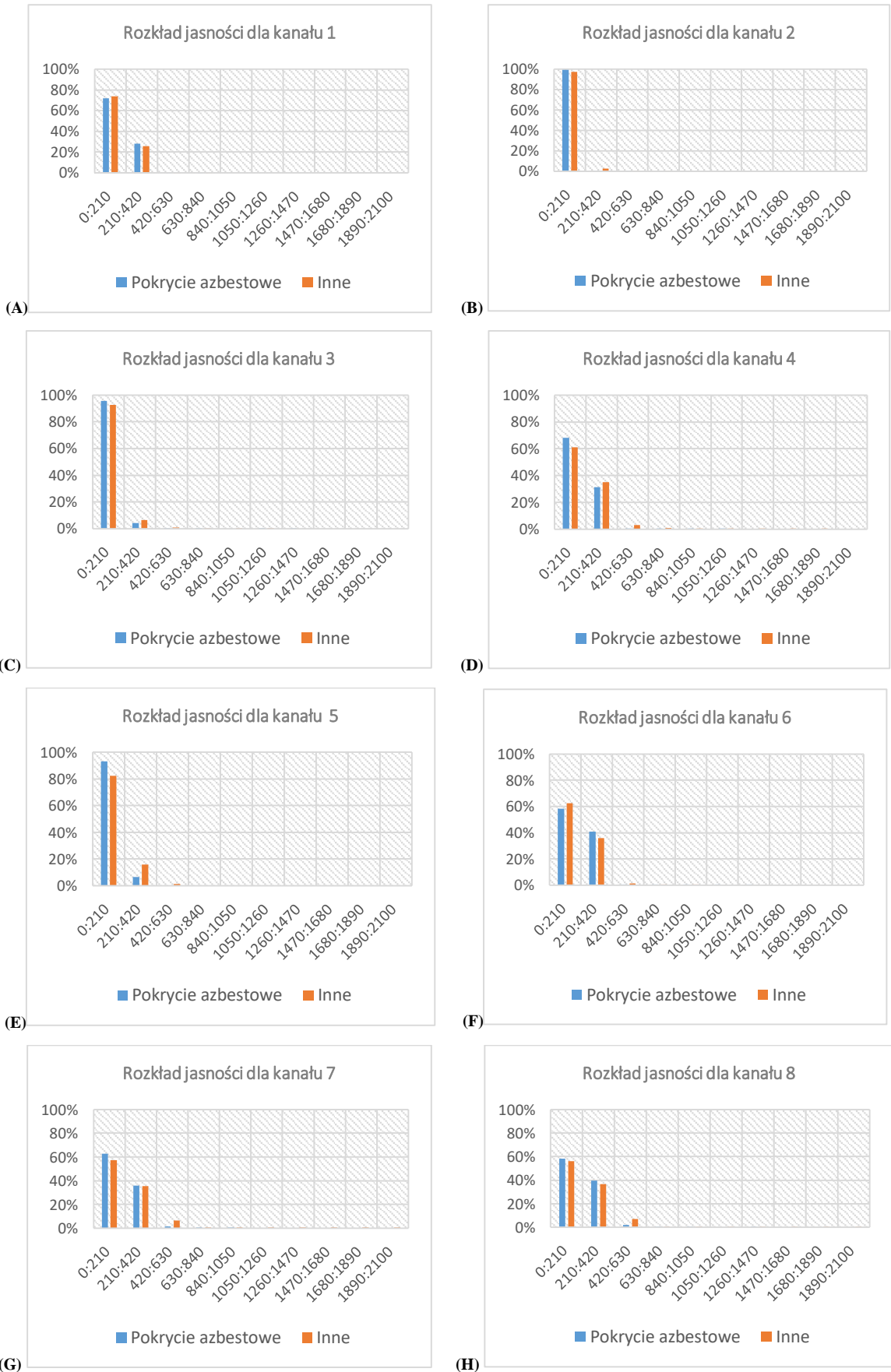
W celu wizualnego porównania jasności pikseli dla obu zbiorów danych, sporządzono wykres (Rys. 20). Zakresy jasności pikseli dla obu zestawów podzielono na dziesięć równych przedziałów. Następnie obliczono procentowy udział jasności w poszczególnych przedziałach. Taka metoda została zastosowana ze względu na zróżnicowaną liczbę pikseli w obu zestawach danych. Z rysunku (Rys. 20) wynika, że jasności pomiędzy 0 a 210 procentowo występują częściej na próbkach 14x14 pikseli. Natomiast dla zestawu 7x7 pikseli procentowo jest więcej jasności z przedziału od 210 do 630. Jasność o wartościach z przedziału od 630 do 2100 stanowi udział mniejszy niż 1% wszystkich pikseli dla obu zestawów danych.

Na końcu przygotowano wizualizacje danych w formie wykresów rozkładu jasności pikseli z podziałem na kanały spektralne (Rys. 21; Rys. 22). Można zauważyć podobieństwo rozkładu w kanałach 1 i 6. W przedziale jasności od 0 do 210 jest więcej pikseli należących do innych pokryć dachowych niż azbestu, natomiast w przedziale od 210 do 420 jest na odwrót (Rys. 21 A,F; Rys. 22A,F). W kanale 2 większość jasności znajduje się w pierwszym przedziale (Rys. 21B; Rys. 22B). Wykres kanału 3 wizualnie jest podobny do 2 kanału, jednak w drugim przedziale (od 210 do 420) znajduje się więcej jasności pikseli dla rozróżnianych klas (Rys. 21C; Rys. 22C). W kanale 4 w przypadku próbek 7x7 pikseli liczba jasności pikseli w pierwszych dwóch przedziałach jest podobna (Rys. 21D), jednak ta prawidłowość nie występuje w przypadku zestawu 14x14 pikseli (Rys. 22D). Słupki dla kanału 5 kształtem przypominają kanał 2 i 3, przy czym zmieniają się proporcje pierwszych dwóch zakresów jasności, pojawia się więcej pikseli o jasnościach od 210 do 420 (Rys. 21E; Rys. 22E). W ostatnich dwóch kanałach – 7 i 8, w przypadku obu zestawów danych, zauważalna jest tendencja do wzrostu liczby pikseli dla dachów nie azbestowych w trzecim przedziale od 420 do 630 (Rys. 21G,H; Rys. 22G,H).

Obliczone statystyki oraz ich wizualizacje nie dają jednoznacznej odpowiedzi, który kanał spektralny i jakie zakresy jasności dla zebranych danych są przydatne w klasyfikacji dachów azbestowych. Jasności pikseli są do siebie bardzo zbliżone, jednak pewne różnice są zauważalne. W związku z tym, w celu zmaksymalizowania informacji wynikającej z różnic jasności pikseli dla azbestowych dachów i innych pokryć dachowych, opracowano opisany w poprzednim rozdziale filtr, którego głównym zadaniem jest wzmacnianie informacji spektralnej.



Rys. 21 Rozkłady jasności pikseli w poszczególnych kanałach spektralnych dla próbek 7x7 pikseli (opracowanie własne)



Rys. 22 Rozkłady jasności pikseli w poszczególnych kanałach spektralnych dla próbek 14x14 pikseli (opracowanie własne)

6.5 PRZEBIEG EKSPERYMENTÓW OBLICZENIOWYCH

Eksperymenty obliczeniowe prowadzono wg trzech strategii. Strategia I oraz II wykonana została w formie numerycznej, gdzie efektem prowadzonych klasyfikacji były macierze filtrujące, wytrenowane klasyfikatory oraz miary oceny dokładności. W dalszych rozważaniach w tym temacie można uwzględnić przypisanie współrzędnych określających położenie poszczególnych dachów (np. centroidy wektorów) do macierzy, pozwoli to na późniejsze zlokalizowanie poszczególnych próbek na obrazie. Strategię III prowadzono w oprogramowaniu graficznym SAGA, więc efektem klasyfikacji były obrazy. Na końcu przetestowano klasyfikatory na przefiltrowanych (filtry wg strategii I oraz II) danych testowych (próbki dachów z miejscowości przyległych do Krzczonowa).

Do klasyfikacji wg strategii I i II zostały przygotowane dwa zestawy danych wejściowych, różniące się wymiarami macierzy (Tab. 15). Klasyfikacja prowadzona była niezależnie na obu zbiorach danych, przy użyciu klasyfikatorów binarnych.

Wybrane ustawienia algorytmu genetycznego wykorzystane dla strategii I oraz II opisano w rozdziale 3.5.1.

Końcowe eksperymenty obliczeniowe prowadzone wg strategii I i II poprzedzono optymalizacją hiperparametrów wybranych klasyfikatorów uczenia maszynowego. Zastosowano metodę przeszukiwania siatki (*ang. grid search*), dzięki której ustalono wartości hiperparametrów wykorzystywanych w obliczeniach.

Tab. 15 Wymiar macierzy wykorzystywanych w procesie klasyfikacji (opracowanie własne)

Nazwa zestawu danych	7pix		14pix	
	Liczba wierszy	Liczba kolumn	Liczba wierszy	Liczba kolumn
Macierz X	435	49	435	196
Macierz y	435	1	435	1

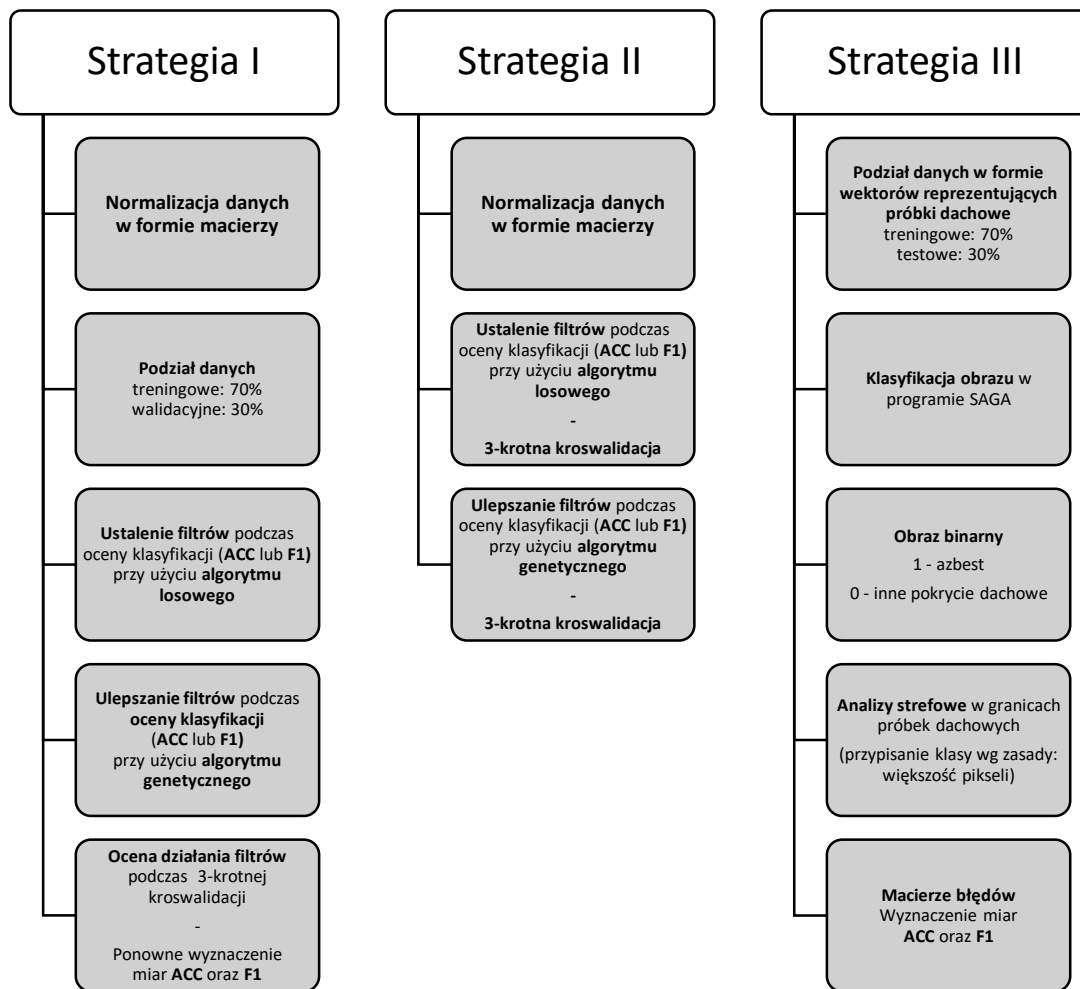
W prezentowanej pracy, jako miary oceny skuteczności klasyfikatora wybrano dokładność ACC oraz F1 dla klasy dachów azbestowych. Pierwsza strategia eksperymentów obliczeniowych przebiegała niezależnie dla obu metryk dokładności. Wyznaczano filtry przy pomocy algorytmu genetycznego, tak aby ACC lub F1 było jak najwyższe.

W drugiej części obliczeń dla strategii I zastosowano wyznaczone filtry do danych wejściowych. Ponownie uczono modele, korzystając z metody rotacyjnej, tzw. k-krokowej metody walidacji krzyżowej - kroswalidacji (*ang. k-fold cross-validation* [kCV]). Metoda ta polega na losowym podziale zbioru danych na k podzbiorów, przy czym k-1 zbiorów tworzy próbę uczącą, natomiast jeden podzbiór – próbę walidacyjną. Procedurę tę powtarza się k razy, tak aby każdy podzbiór wykorzystany był jako zbiór walidacyjny (Géron, 2020). W prezentowanej pracy, zbiór danych dzielono na trzy podzbiory, w uwagi na małą liczebność próbek. Przeprowadzone obliczenia wykonano w celu oceny skuteczności działania filtrów, gdyż za każdym przebiegiem algorytmu klasyfikator uczono i testowano na niezależnych danych (podzbiórach). Pozwoliło to na wyznaczenie średnich wartości miar ACC oraz F1.

Mając na względzie różne proporcje klas dachów azbestowych oraz innych pokryć dachowych, wykonano dwa rodzaje kroswalidacji. Kroswalidację stratyfikowaną, która polegała na podziale danych na trzy podzbiory (tzw. foldy) po 145 próbek dachowych. Każdy podzbiór zawierał reprezentatywny zestaw przykładów z obu klas (podzbiór 1: 41 próbek azbestu oraz 104 inne próbki, podzbiór 2 i 3: po 42 próbki azbestu oraz po 103 inne próbki). Dzięki temu można było zminimalizować efekt nieodpowiedniego podziału danych na zbiór treningowy i walidacyjny, co mogło prowadzić do uzyskania wyników obarczonych błędem. Kroswalidacja stratyfikowana jest szczególnie przydatna w przypadku niezbalansowanych zbiorów danych, gdzie proporcje klas są nierównomierne (Géron, 2020). Dla porównania wykonano również obliczenia standardowej kroswalidacji, gdzie nie ustalono proporcji podziału danych na podzbiory, tylko zastosowano metodę losową.

W obliczeniach dla strategii II wykorzystywano kroswalidację stratyfikowaną już w trakcie wyznaczania macierzy filtrującej. Ponownie wybranymi parametrami oceny była dokładność ACC oraz miara F1. W strategii III wykonano klasyfikację w ogólnodostępnym oprogramowaniu SAGA. Następnie wyznaczono ręcznie macierze błędów, z których policzono miary ACC oraz F1.

Prowadzone eksperymenty obliczeniowe przedstawiono na ogólnym schemacie (Rys. 23). Szczegółowy opis poszczególnych strategii umieszczono w kolejnych podrozdziałach.



Rys. 23 Schemat prowadzonych eksperymentów obliczeniowych (opracowanie własne)

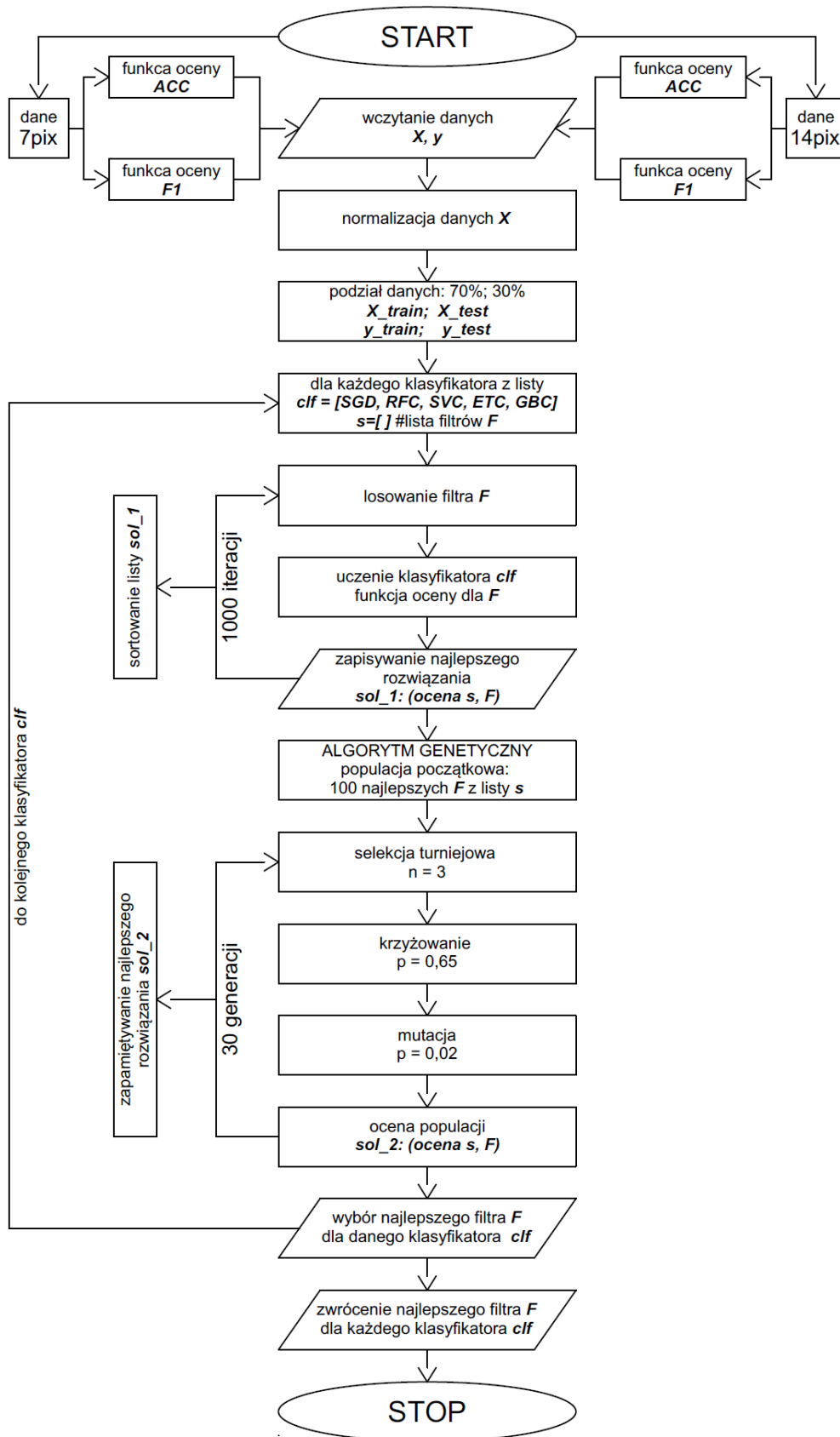
6.5.1 Strategia I

Eksperymenty obliczeniowe prowadzono w dwóch etapach. W pierwszym etapie ustalano parametry dla opracowanego filtra przy zastosowaniu wybranych klasyfikatorów binarnych, a przebieg obliczeń przedstawiono na opracowanym schemacie blokowym (Rys. 24). W drugim etapie przetestowano wyznaczone filtry, ucząc i oceniając klasyfikatory 3-krotną metodą sprawdzianu krzyżowego. Podział danych ograniczono do trzech podzbiorów z uwagi na nieliczny zestaw danych uczących (435 próbek).

Etap pierwszy eksperymentów obliczeniowych, zaprezentowany na schemacie blokowym (Rys. 24), przebiega wg następujących etapów:

1. Wczytanie danych wejściowych (zestaw 7 lub 14 pikseli).
2. Wskazanie funkcji oceny klasyfikatorów (ACC lub F1).

3. Normalizacja każdego elementu macierzy X_i , będącego jasnością pikseli na oryginalnym obrazie, dla każdego kanału spektralnego. Podczas normalizacji przyjęto zero jako wartość minimalną jasności we wszystkich kanałach, w związku z tym elementy macierzy X_i dzielono przez maksymalną wartość występującą w danym kanale spektralnym.
4. Podział danych na dane uczące: 70% oraz dane walidacyjne: 30%, z uwzględnieniem proporcji typów pokryć dachowych.
5. Wybór klasyfikatora ze zdefiniowanej listy klasyfikatorów clf oraz utworzenie pustej listy s , do której będą dodawane filtry F .
 - a. Losowanie filtra F (macierzy) z zastosowaniem heurystycznego losowego algorytmu.
 - b. Uczenie wybranego klasyfikatora clf na danych przefiltrowanych wg wylosowanego filtra F i ocenianego go wybranym w punkcie 2 parametrem oceny.
 - c. Zapisanie najlepszego rozwiązania do s : wartość oceny clf oraz filtr F .
 - d. Powtórzenie 1000-krotne czynności od podpunktu a wraz z równoczesnym sortowaniem listy najlepszych rozwiązań sol_1 .
6. Wybór 100 filtrów F , dla których funkcja oceny była najwyższa. Zestaw tych macierzy służy do stworzenia początkowej populacji chromosomów dla algorytmu genetycznego.
 - a. Selekcja turniejowa. Z populacji początkowej losowane są po trzy chromosomy (filtry F) i następuje trenowanie klasyfikatora. Filtr F , dla którego uzyskano najwyższą funkcję oceny zostaje dołączony do nowej populacji. Losowanie trwa do zapełnienia wszystkich 100 miejsc w nowej generacji.
 - b. Krzyżowanie z prawdopodobieństwem na poziomie 0,65.
 - c. Mutacja z prawdopodobieństwem na poziomie 0,02.
 - d. Ocena nowej populacji wg parametru wybranego w punkcie 2.
 - e. Powtórzenie 30-krotne czynności od podpunktu a z równoczesnym zapisaniem najlepszego rozwiązania sol_2 .
7. Wybór najlepszego filtra F dla danego klasyfikatora clf .
8. Powtórzenie czynności od punktu 5, gdzie następuje zmiana klasyfikatora.
9. Zwrócenie najlepszego filtra F dla każdego klasyfikatora clf i zapisanie wyników w pliku *csv*.



Rys. 24 Schemat blokowy eksperymentów obliczeniowych, pozwalający na wyznaczenie parametrów filtrowania danych (opracowanie własne)

6.5.2 Strategia II

Po przeprowadzeniu badań wg strategii I, zauważono spadek wartości metryk dokładności klasyfikacji po wykonaniu ponownej oceny klasyfikacji z wykorzystaniem krosvalidacji. W związku z tym, zmieniono sposób podziału danych uczących oraz walidacyjnych. W strategii I podział danych wykonywano jednokrotnie dla każdego zestawu próbek oraz metryk dokładności (parametr ACC lub F1). Natomiast w strategii II zastosowano 3-krotną walidację krzyżową. Przy każdej iteracji algorytmu losowego oraz dla każdej generacji algorytmu genetycznego dzielono dane wejściowe na trzy podzbiory i trzykrotnie uczono oraz oceniano klasyfikację dla danego filtra. Wyznaczone wartości dla filtra wg strategii II powinny być bardziej uniwersalne, a uzyskane w ten sposób modele klasyfikacji mniej przeuczone.

6.5.3 Strategia III

Celem strategii III było przeprowadzenie klasyfikacji za pomocą narzędzi aktualnie stosowanych w analizach teledetekcyjnych oraz porównanie otrzymanych wyników z wynikami uzyskanymi z wykorzystaniem proponowanego algorytmu genetycznego filtrującego dane wejściowe. Do analiz wykorzystano warstwy wektorowe określające dachy (rozmiar pojedynczej próbki: 7x7 lub 14x pikseli) oraz obraz satelitarny WorldView-3. Warstwy wektorowe podzielono losowo na dane uczące (70% - 304 próbki) oraz testowe (30% - 131 próbek) z zachowaniem proporcji klas. W przypadku obu zestawów danych (7x7 oraz 14x14 pikseli) zastosowano analogiczny podział danych wejściowych na dane uczące i testowe.

Klasyfikacje fragmentów obrazu WorldView-3 wykonano w programie SAGA 9.0.2 (<https://saga-gis.sourceforge.io/en/index.html>), z wykorzystaniem klasyfikatorów RFC oraz SVC z ustawieniami domyślnymi. Przeprowadzono próby modyfikacji hiperparametrów dla obu klasyfikatorów, jednak osiągnięte wyniki utrzymywały się na podobnym poziomie lub wykazywały pogorszenie dokładności klasyfikacji. Program SAGA nie posiada wbudowanego narzędzia przeszukiwania siatki hiperparametrów uczenia maszynowego.

W wyniku klasyfikacji otrzymywano obraz, na którym każdy piksel miał przypisaną klasę azbest (liczba 1) lub inne pokrycie dachowe (liczba 0). Z uwagi na klasyfikację metodą

pikselową, w granicy jednego pokrycia dachowego znajdowały się piksele z obu klas. Wcześniejsze strategie opierały się na klasyfikacji obiektowej – wg przygotowanych próbek dachów, w związku z tym również w tym przypadku postanowiono każdej próbce dachu przypisać klasę zgodną z klasą większości pikseli w granicach tego dachu. Następnie obliczono macierze błędów i wyznaczono parametry ACC oraz F1 dla klasy azbest.

6.5.4 Testowanie klasyfikatorów na przefiltrowanych danych

Po wyznaczeniu filtrów wg strategii I oraz II wykonano ocenę skuteczności klasyfikatorów trenowanych na przefiltrowanych danych. W tym celu przeprowadzono eksperymenty obliczeniowe wg następujących kroków:

1. Wczytanie danych treningowych (435 próbek z miejscowości Krzczonów) oraz testowych (70 próbek z miejscowości obrzeżnych).
2. Znormalizowanie danych.
3. Filtrowanie danych wg macierzy F wyznaczonych w strategii I oraz II.
4. Trenowanie klasyfikatorów na danych uczących.
5. Prognoza dla danych testowych.
6. Ocena klasyfikatorów: precyzja, czułość, F1, ACC.

Obliczenia wykonano dla każdego zestawu danych (7x7 pikseli oraz 14x14 pikseli) czterokrotnie: dla strategii I – filtry wyznaczone wg funkcji oceny ACC i F1 oraz dla strategii II – filtry również wyznaczone wg miar ACC i F1.

6.6 ŚRODOWISKO OBLICZENIOWE

Podczas przetwarzania obrazów oraz eksperymentów obliczeniowych wykorzystano: komputer z systemem Windows 10, z procesor Intel(R) Core(TM) i5-7200U CPU @ 2,50GHz, 2701 MHz; pamięcią RAM: 20GHz i procesor graficzny Intel(R) HD Graphics 620.

Proces klasyfikacji zestawów próbek dachowych prowadzony był w oparciu o algorytmy zaimplementowane w następujących bibliotekach:

- a) Scikit-learn – narzędzie zawierające algorytmy uczenia maszynowego.

- b) DEAP – narzędzie zawierające między innymi implementacje algorytmu genetycznego.
- c) OpenCV – biblioteka wykorzystywana w programie SAGA do klasyfikacji obrazów.

W prezentowanej pracy wszystkie obliczenia wykonano z wykorzystaniem języka programowania *Python* w wersji 3.9. Kod został napisany w środowisku Jupyter. Kod wykorzystanych funkcji oraz programów *Python*, niezbędny dla wykonania eksperymentów obliczeniowych dla strategii I oraz II został załączony do pracy jako **Załącznik 1**, **Załącznik 2**, **Załącznik 3**. Dodatkowo dane w formie macierzy X_i oraz y , jak również wykorzystywany kod znajdują się pod linkiem: <https://github.com/JudKsi/asbestos-roofs.git>.

7 WYNIKI KLASYFIKACJI

W rozdziale przedstawiono wyniki eksperymentów obliczeniowych prowadzonych wg strategii opisanych w rozdziale 6.5 oraz wyniki testowania klasyfikatorów wytrenowanych na podstawie przefiltrowanych danych. Rezultaty wykonanych obliczeń zestawiono w formie tabelarycznej oraz przedstawiono na wykresach.

7.1 WYZNACZENIE FILTRÓW – STRATEGIA I

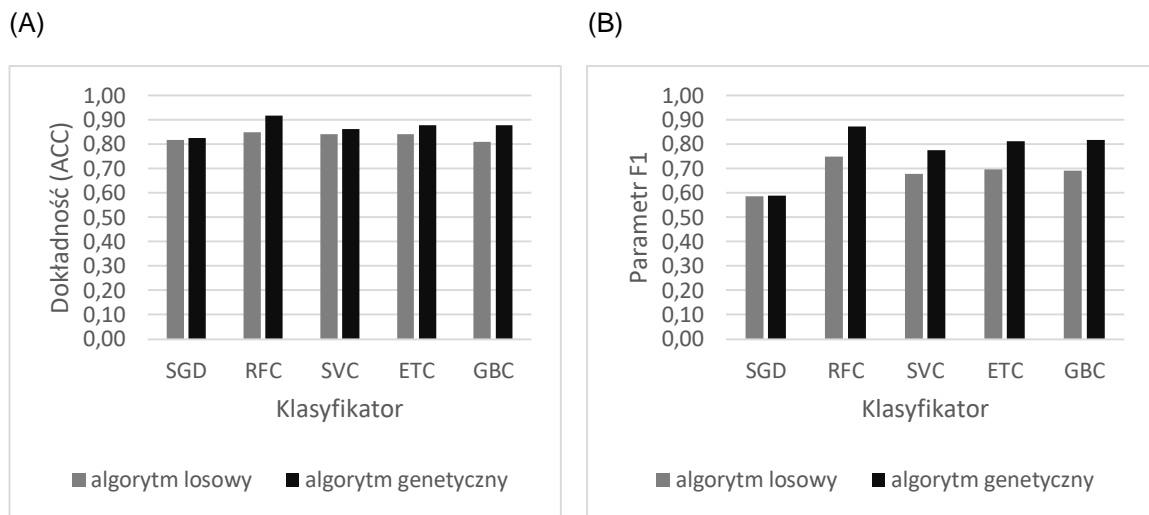
Otrzymane wyniki eksperymentów obliczeniowych zestawiono w tabeli (Tab. 16). Uzyskiwane dokładności klasyfikacji dla każdego wykorzystanego klasyfikatora, przekroczyły wartość 80%. W przypadku parametru F1 rozrzut otrzymywanych wartości wyników był większy i zależał od wykorzystywanego klasyfikatora.

Najwyższe wyniki dla obu parametrów oceny klasyfikacji uzyskiwano przy użyciu klasyfikatora RFC. Dokładność klasyfikacji wyniosła dla obu zestawów danych wejściowych ponad 90%, natomiast parametr F1 w przypadku próbek 7x7 pikseli nie odbiegał znacząco od tej wartości i wyniósł 87%. Biorąc pod uwagę parametr F1, wyniki dla klasyfikatora RFC zbliżone są z wynikami klasyfikatorów ETC oraz GBC, szczególnie dla próbek 14x14 pikseli. Najniższe wartości oceny klasyfikacji otrzymano dla klasyfikatora SGD przy próbkach 7x7 pikseli oraz dla klasyfikatora SVC przy próbkach 14x14 pikseli.

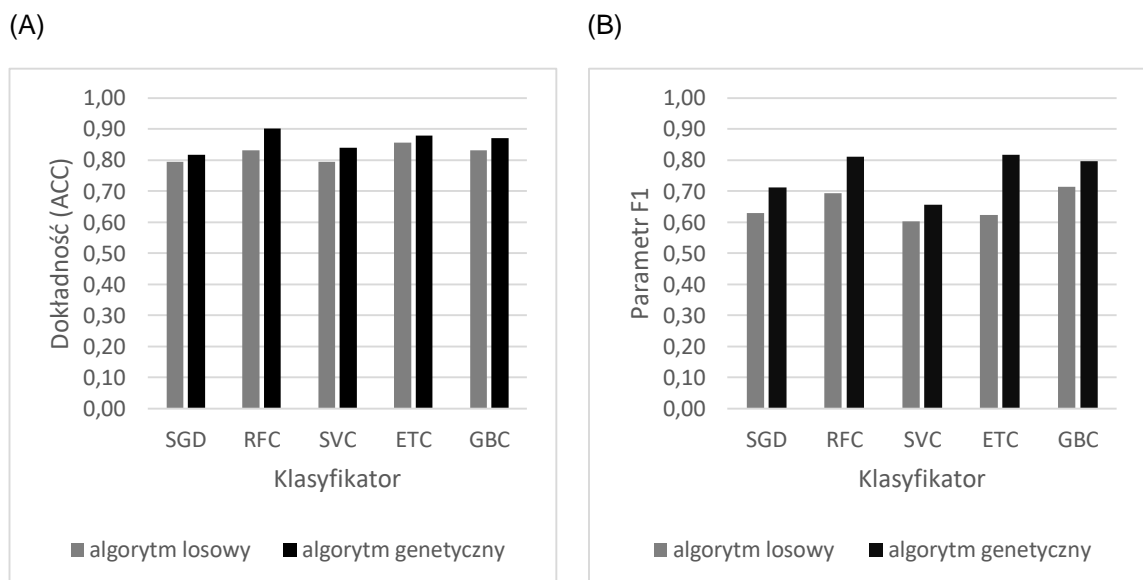
Tab. 16 Wyniki klasyfikacji z wykorzystaniem algorytmów poprawiających filtr danych (opracowanie własne)

Klasyfikator	Algorytm wyboru filtra	7 x 7 pikseli		14 x 14 pikseli	
		Parametr określający dokładność			
		ACC	F1	ACC	F1
SGD	losowy	0,81679	0,58462	0,79389	0,62963
	genetyczny	0,82443	0,58915	0,81679	0,71111
RFC	losowy	0,84733	0,75000	0,83206	0,69333
	genetyczny	0,91603	0,87324	0,90076	0,81013
SVC	losowy	0,83969	0,67692	0,79389	0,60317
	genetyczny	0,86260	0,77612	0,83969	0,65625
ETC	losowy	0,83969	0,69697	0,85496	0,62295
	genetyczny	0,87786	0,81159	0,87786	0,81690
GBC	losowy	0,80916	0,69136	0,83206	0,71429
	genetyczny	0,87786	0,81690	0,87023	0,79518

Na wykresach (Rys. 25; Rys. 26) można zauważyć wzrost metryk dokładności po zastosowaniu algorytmu genetycznego. Dokładność ACC dla różnych klasyfikatorów poprawionych algorytmem genetycznym nie różnią się znacząco (Rys. 25A; Rys. 26A). Natomiast, w przypadku F1 można zauważyć wzrost dokładności po wykorzystaniu algorytmu genetycznego, z wyjątkiem klasyfikatora SGD dla próbek 7x7 pikseli (Rys. 25B; Rys. 26B).



Rys. 25 Wyniki klasyfikacji dla próbek 7x7 pikseli z podziałem na klasyfikatory (A) metryka: dokładność ACC; (B) metryka: parametr F1 (opracowanie własne)



Rys. 26 Wyniki klasyfikacji dla próbek 14x14 pikseli z podziałem na klasyfikatory (A) metryka: dokładność ACC; (B) metryka: parametr F1 (opracowanie własne)

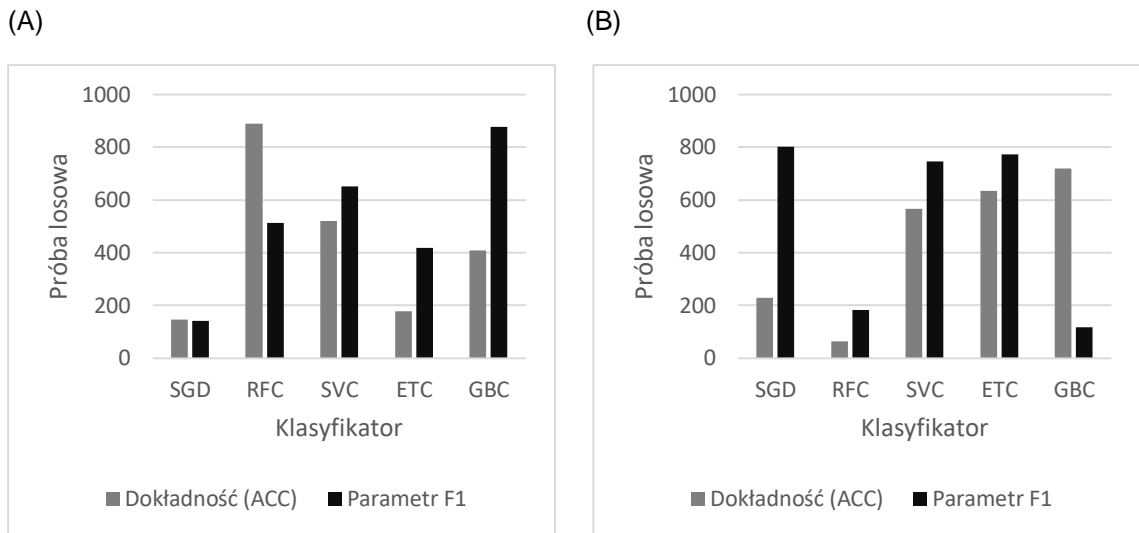
W czasie obliczeń zbierano również informacje na temat numeru próby losowej oraz numeru generacji algorytmu genetycznego, dla których klasyfikatory uzyskiwały najwyższe rezultaty wybranych parametrów oceny dokładności (Tab. 17).

Rozkład numerów próby losowej przedstawiono na rysunku (Rys. 27). W przypadku próbek 7x7 pikseli, jak również 14x14 pikseli nie ma żadnej zależności, co jest zrozumiałe przy zastosowaniu algorytmu opartego na losowaniu.

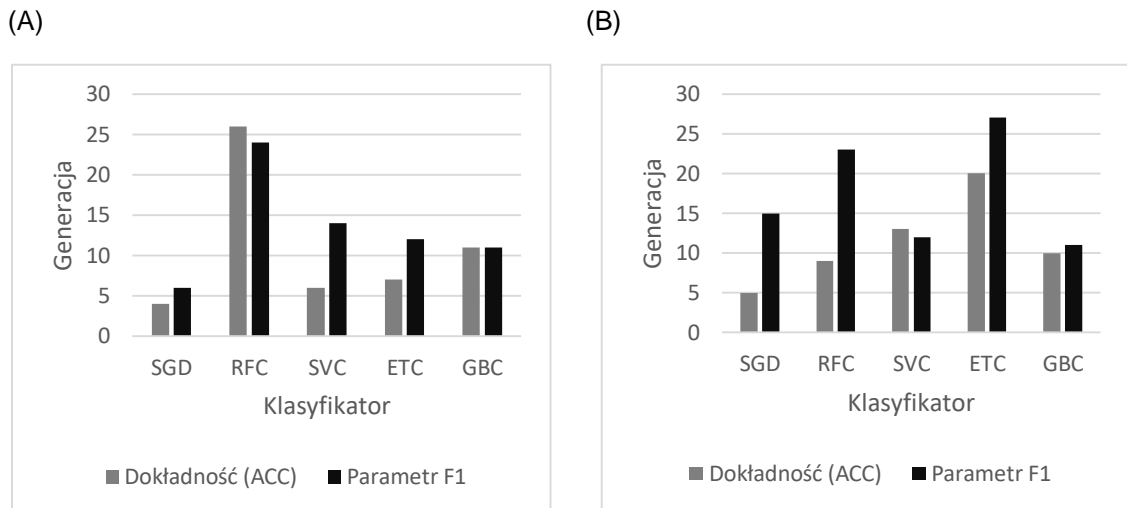
Ewolucja macierzy filtrowania pozwalająca na uzyskanie najwyższego wyniku funkcji celu przebiegała w mniejszej liczbie generacji dla parametru dokładność (ACC), niż dla parametru F1 (Rys. 28). Wystąpiły dwa wyjątki, dla próbek 7x7 pikseli: klasyfikator RFC (Rys. 28A) oraz klasyfikator SVC dla próbek 14x14 pikseli (Rys. 28B).

Tab. 17 Numer próby losowej i numer generacji algorytmu genetycznego, dla których wyznaczono najwyższą wartość parametru oceny dokładności klasyfikacji (opracowanie własne)

Klasyfikator	Parametr określający dokładność	7x7 pikseli		14x14 pikseli	
		Numer próby losowej	Numer generacji	Numer próby losowej	Numer generacji
SGD	ACC	145	4	228	5
	F1	141	6	802	15
RFC	ACC	890	26	64	9
	F1	513	24	182	23
SVC	ACC	519	6	565	13
	F1	651	14	746	12
ETC	ACC	177	7	635	20
	F1	418	12	773	27
GBC	ACC	408	11	718	10
	F1	876	11	116	11



Rys. 27 Numer próby losowej dla której klasyfikator uzyskał najwyższy wynik metryk dokładności (A) próbki 7x7 pikseli; (B) próbki 14x14 pikseli (opracowanie własne)



Rys. 28 Numer generacji dla której klasyfikator uzyskał najwyższy wynik metryk dokładności (A) próbki 7x7 pikseli; (B) próbki 14x14 pikseli (opracowanie własne)

W tabelach (Tab. 18; Tab. 19) zestawiono wszystkie wartości środkowe oraz odchylenia dla poszczególnych kanałów spektralnych, wyznaczone przez algorytm genetyczny. Dla lepszego zwizualizowania otrzymanych zakresów filtra sporządzono wykresy (Rys. 29; Rys. 30). Celem działania filtra jest wzmocnienie informacji obrazowej, a dobór jego wartości jest realizowany przez sztuczną inteligencję (AI) przy użyciu algorytmu genetycznego.

Większość informacji spektralnej (DN, *ang. Digital Number*) znajduje się w przedziale od 0 do 420 (Rys. 21; Rys. 22). Po normalizacji dane te przyjmują wartości od

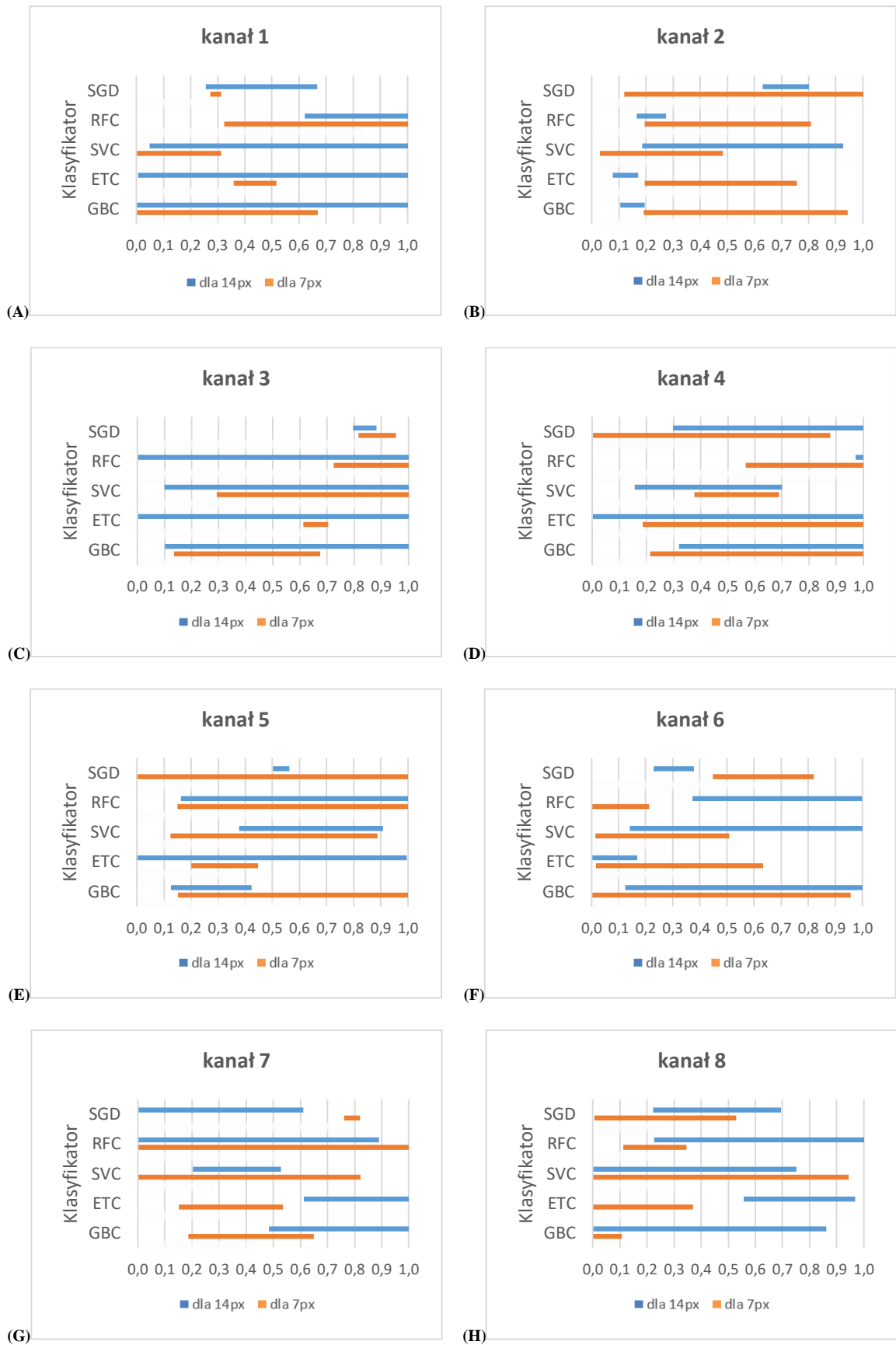
0,0 do 0,2. W związku z tym, przedziały filtrowania rozpoczynające się od 0,2 powodują odrzucanie znacznej liczby danych wejściowych.

Tab. 18 Wartość środkowa i odchylenie (parametry filtra) dla poszczególnych kanałów spektralnych wyznaczone przez algorytm genetyczny wykorzystujący jako funkcje oceny dokładność ACC (opracowanie własne)

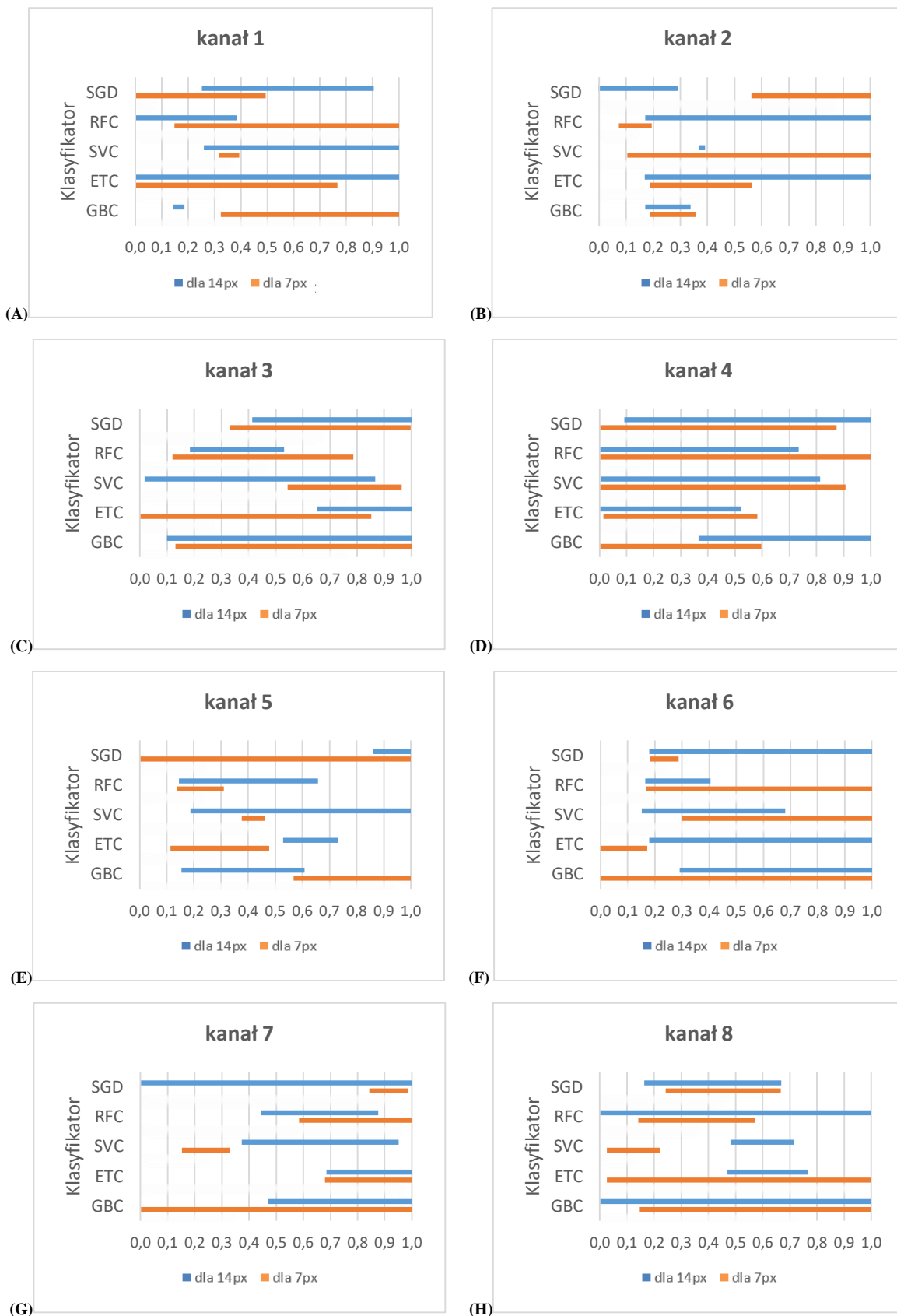
Zestaw próbek	Klasyfikator	Kanał 1		Kanał 3		Kanał 3		Kanał 4		Kanał 5		Kanał 6		Kanał 7		Kanał 8	
		Wartość środkowa	Odchylenie	Wartość środkowa	Odchylenie	Wartość środkowa	Odchylenie	Wartość środkowa	Odchylenie	Wartość środkowa	Odchylenie	Wartość środkowa	Odchylenie	Wartość środkowa	Odchylenie	Wartość środkowa	Odchylenie
7px	SGD	0,29	0,02	0,73	0,61	0,89	0,07	0,39	0,49	0,23	0,96	0,63	0,19	0,79	0,03	0,27	0,26
14px		0,46	0,21	0,72	0,09	0,84	0,04	0,93	0,63	0,53	0,03	0,30	0,07	0,14	0,47	0,46	0,24
7px	RFC	0,82	0,50	0,50	0,31	0,96	0,23	0,92	0,35	0,85	0,70	0,10	0,12	0,57	0,65	0,23	0,12
14px		0,96	0,33	0,22	0,05	0,88	1,00	1,00	0,03	0,76	0,60	0,69	0,31	0,18	0,71	0,77	0,54
7px	SVC	0,09	0,22	0,26	0,23	0,84	0,55	0,53	0,16	0,51	0,38	0,26	0,25	0,27	0,56	0,30	0,64
14px		0,90	0,85	0,56	0,37	0,63	0,53	0,43	0,27	0,64	0,27	0,73	0,58	0,37	0,16	0,29	0,46
7px	ETC	0,44	0,08	0,48	0,28	0,66	0,05	0,72	0,53	0,32	0,12	0,33	0,31	0,34	0,19	0,12	0,25
14px		0,85	0,84	0,12	0,05	0,81	0,81	0,61	0,94	0,42	0,58	0,05	0,12	0,90	0,29	0,76	0,21
7px	GBC	0,30	0,37	0,57	0,38	0,40	0,27	0,71	0,50	0,66	0,51	0,20	0,76	0,42	0,23	0,04	0,06
14px		0,94	0,94	0,15	0,05	0,76	0,66	0,91	0,59	0,27	0,15	0,67	0,55	0,89	0,41	0,02	0,85

Tab. 19 Wartość środkowa i odchylenie (parametry filtra) dla poszczególnych kanałów spektralnych wyznaczone przez algorytm genetyczny wykorzystujący jako funkcje oceny parametr F1 (opracowanie własne)

Zestaw próbek	Klasyfikator	Kanał 1		Kanał 3		Kanał 3		Kanał 4		Kanał 5		Kanał 6		Kanał 7		Kanał 8	
		Wartość środkowa	Odchylenie	Wartość środkowa	Odchylenie	Wartość środkowa	Odchylenie	Wartość środkowa	Odchylenie	Wartość środkowa	Odchylenie	Wartość środkowa	Odchylenie	Wartość środkowa	Odchylenie	Wartość środkowa	Odchylenie
7px	SGD	0,07	0,42	0,86	0,30	0,67	0,33	0,31	0,57	0,23	0,82	0,24	0,05	0,92	0,07	0,45	0,21
14px		0,58	0,33	0,06	0,23	0,98	0,57	0,81	0,72	0,97	0,11	0,95	0,78	0,35	0,92	0,42	0,25
7px	RFC	1,00	0,85	0,13	0,06	0,45	0,33	0,18	0,92	0,22	0,09	0,68	0,52	0,96	0,37	0,36	0,22
14px		0,07	0,32	0,73	0,56	0,36	0,17	0,33	0,40	0,40	0,26	0,28	0,12	0,66	0,22	0,36	0,92
7px	SVC	0,35	0,04	0,68	0,58	0,75	0,21	0,17	0,73	0,42	0,04	0,97	0,67	0,24	0,09	0,12	0,10
14px		1,00	0,74	0,38	0,01	0,44	0,42	0,41	0,41	0,97	0,78	0,42	0,26	0,66	0,29	0,60	0,12
7px	ETC	0,31	0,46	0,38	0,19	0,26	0,59	0,30	0,28	0,30	0,18	0,02	0,15	0,96	0,28	0,92	0,89
14px		0,78	0,97	0,76	0,59	0,83	0,18	0,04	0,48	0,63	0,10	0,94	0,76	0,90	0,21	0,62	0,15
7px	GBC	0,80	0,48	0,27	0,09	0,75	0,62	0,03	0,56	0,92	0,35	0,77	0,81	0,34	0,67	0,72	0,57
14px		0,16	0,02	0,25	0,08	0,69	0,59	0,97	0,61	0,38	0,23	0,83	0,54	0,94	0,47	0,77	0,95



Rys. 29 Zakresy przedziałów filtrowania dla poszczególnych kanałów spektralnych wyznaczone przez algorytm genetyczny wykorzystujący jako funkcje oceny dokładność ACC (opracowanie własne)



Rys. 30 Zakresy przedziałów filtrowania dla poszczególnych kanałów spektralnych wyznaczone przez algorytm genetyczny wykorzystujący jako funkcje oceny parametr F1 (opracowanie własne)

7.2 OCENA SKUTECZNOŚCI FILTRÓW – STRATEGIA I

Wyniki niezależnej oceny jakości wyznaczonych filtrów F wg strategii I zaprezentowano w dwóch tabelach (Tab. 20; Tab. 21).

Tab. 20 Zestawienie wyników oceny klasyfikatora przy użyciu krosvalidacji, dla próbek 7x7 pikseli (opracowanie własne)

Klasyfikator	Krosvalidacja stratyfikowana				Krosvalidacja			
	Podzbiór			Średnia	Podzbiór			Średnia
	1	2	3		1	2	3	
	ACC							
SGD	0,80	0,71	0,72	0,74	0,71	0,47	0,72	0,63
RFC	0,85	0,85	0,83	0,84	0,86	0,87	0,80	0,84
SVC	0,81	0,79	0,77	0,79	0,79	0,79	0,77	0,78
ETC	0,77	0,78	0,77	0,77	0,79	0,82	0,76	0,79
GBC	0,84	0,79	0,83	0,82	0,75	0,81	0,80	0,79
	F1							
SGD	0,71	0,71	0,72	0,71	0,70	0,47	0,72	0,63
RFC	0,86	0,80	0,74	0,81	0,78	0,85	0,81	0,81
SVC	0,83	0,78	0,75	0,79	0,78	0,80	0,76	0,78
ETC	0,74	0,83	0,80	0,79	0,80	0,85	0,81	0,82
GBC	0,82	0,80	0,81	0,81	0,74	0,79	0,78	0,77

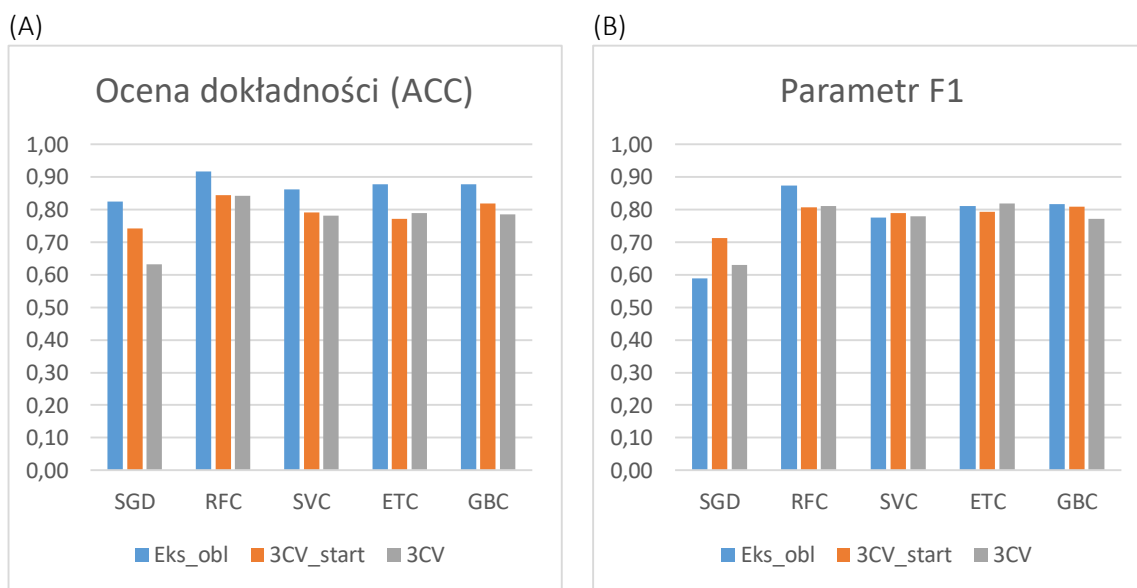
Tab. 21 Zestawienie wyników oceny klasyfikatora przy użyciu krosvalidacji, dla próbek 14x14 pikseli (opracowanie własne)

Klasyfikator	Krosvalidacja stratyfikowana				Krosvalidacja			
	Podzbiór			Średnia	Podzbiór			Średnia
	1	2	3		1	2	3	
	ACC							
SGD	0,46	0,71	0,72	0,63	0,71	0,71	0,72	0,71
RFC	0,75	0,69	0,67	0,70	0,68	0,68	0,71	0,69
SVC	0,80	0,75	0,72	0,76	0,75	0,75	0,72	0,74
ETC	0,74	0,67	0,73	0,71	0,67	0,70	0,72	0,70
GBC	0,83	0,68	0,72	0,74	0,75	0,75	0,71	0,74
	F1							
SGD	0,71	0,39	0,72	0,60	0,71	0,71	0,72	0,71
RFC	0,76	0,66	0,72	0,71	0,71	0,71	0,72	0,71
SVC	0,71	0,71	0,71	0,71	0,71	0,71	0,71	0,71
ETC	0,67	0,74	0,72	0,71	0,61	0,75	0,73	0,70
GBC	0,66	0,64	0,63	0,64	0,64	0,63	0,70	0,66

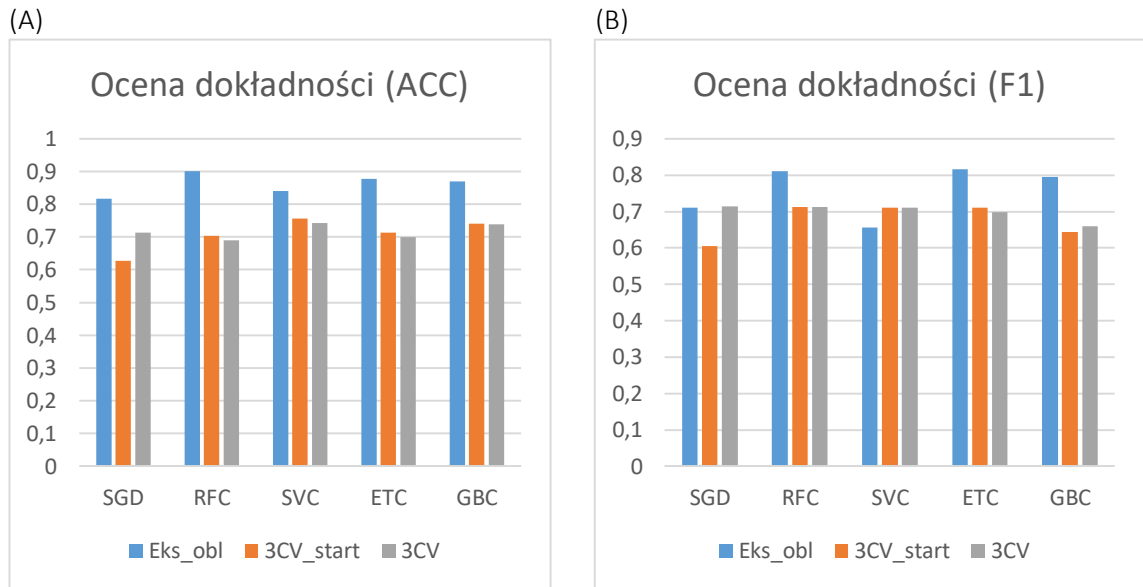
Wyznaczone wartości miar oceny dokładności z krosvalidacji losowej oraz stratyfikowanej spadły w porównaniu z wynikami oceny uzyskiwanymi podczas wyznaczania filtrów F (Rys. 31; Rys. 32). Nie ma znaczących różnic pomiędzy wynikami obu krosvalidacji, jednak z uwagi na nierówną liczebność klas, bardziej wiarygodną oceną wydaje się krosvalidacja stratyfikowana.

W przypadku próbek 7x7 pikseli, ocena klasyfikatora RFC pozostała na stosunkowo wysokim poziomie - ponad 80%. Ciekawe jest też zachowanie parametru F1 dla klasyfikatora SGD, gdyż parametr ten wzrósł z 59% podczas wyznaczania filtra do 71% dla krosvalidacji stratyfikowanej.

W przypadku próbek 14x14 pikseli zauważalny jest większy spadek wartości oceny klasyfikatorów. Możliwe, że podczas uczenia doszło do przetrenowania modelu z uwagi na różnorodność jasności pikseli znajdujących się w otoczeniu budynków.



Rys. 31 Wyniki średniej oceny klasyfikatora, dla próbek 7x7 pikseli, otrzymane podczas: wyznaczania filtrów (Eks_obl), stratyfikowanej walidacji krzyżowej (3CV_strat), walidacji krzyżowej (3CV) (opracowanie własne)



Rys. 32 Wyniki średniej oceny klasyfikatora, dla próbek 14x14 pikseli, otrzymane podczas: wyznaczania filtrów (Eks_obl), stratyfikowanej walidacji krzyżowej (3CV_strat), walidacji krzyżowej (3CV) (opracowanie własne)

7.3 WYZNACZENIE FILTRÓW – STRATEGIA II

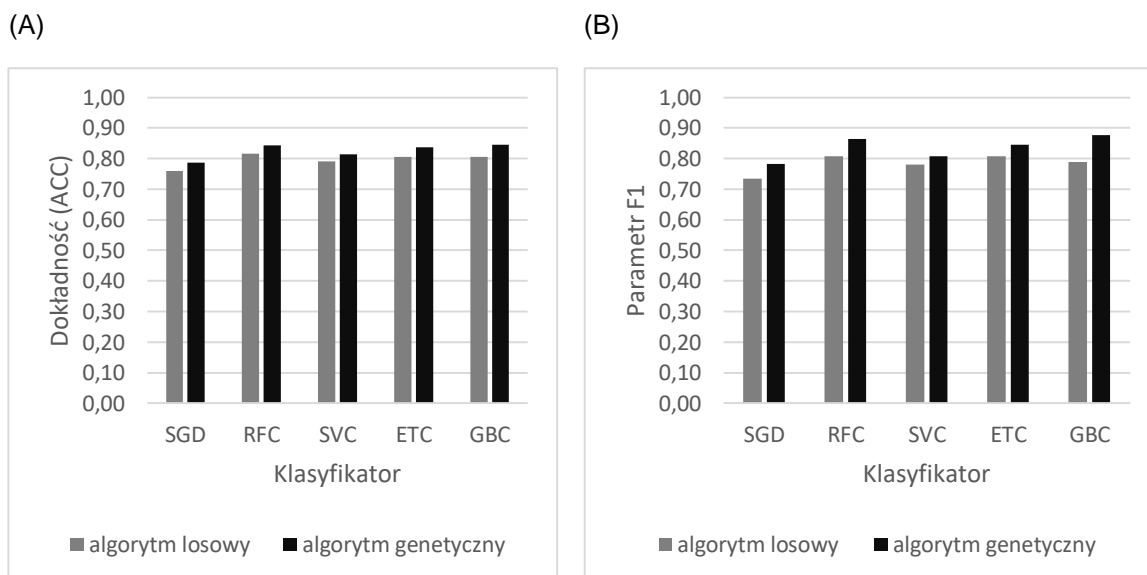
Otrzymane średnie wyniki eksperymentów obliczeniowych dla strategii II zestawiono w tabeli (Tab. 22). Uzyskiwane dokładności klasyfikacji dla każdego wykorzystanego klasyfikatora, przekroczyły wartość 70%.

Najwyższe wyniki dla obu parametrów oceny klasyfikacji uzyskiwano przy użyciu klasyfikatorów opartych na drzewach decyzyjnych: RFC, ETC, GBC. Najsłabszym klasyfikatorem dla analizowanego problemu okazał się klasyfikator SGD, choć wyniki dla klasyfikatora SVC były wyższe zaledwie o 2%.

Tab. 22 Średnie wartości miar oceny dokładności podczas klasyfikacji z wykorzystaniem algorytmów poprawiających filtr danych przy zastosowaniu krosvalidacji (opracowanie własne)

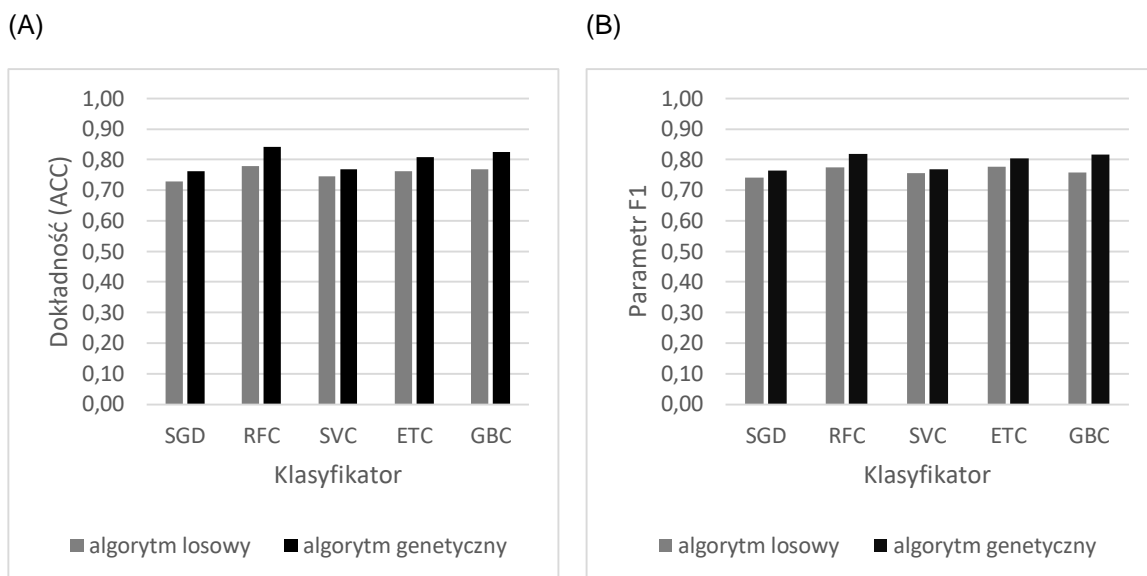
Klasyfikator	Algorytm wyboru filtra	7 x 7 pikseli		14 x 14 pikseli	
		Parametr określający dokładność			
		ACC	F1	ACC	F1
SGD	losowy	0,75862	0,73333	0,72874	0,74023
	genetyczny	0,78621	0,78161	0,76092	0,76322
RFC	losowy	0,81609	0,80690	0,77931	0,77471
	genetyczny	0,84368	0,86437	0,84138	0,81839
SVC	losowy	0,79080	0,77931	0,74483	0,75632
	genetyczny	0,81379	0,80690	0,76782	0,76782
ETC	losowy	0,80460	0,80690	0,76092	0,77701
	genetyczny	0,83678	0,84598	0,80690	0,80460
GBC	losowy	0,80460	0,78851	0,76782	0,75862
	genetyczny	0,84598	0,87586	0,82529	0,81609

Na wykresach (Rys. 33; Rys. 34) można zauważyć wzrost metryk dokładności po zastosowaniu algorytmu genetycznego. Wartości dokładności ACC oraz F1 dla wszystkich klasyfikatorów wzrosły o kilka procent.



Rys. 33 Wyniki klasyfikacji dla próbek 7x7 pikseli z podziałem na klasyfikatory przy zastosowaniu krosvalidacji

(A) metryka: dokładność ACC; (B) metryka: parametr F1
(opracowanie własne)



Rys. 34 Wyniki klasyfikacji dla próbek 14x14 pikseli z podziałem na klasyfikatory przy zastosowaniu krosvalidacji
(A) metryka: dokładność ACC; (B) metryka: parametr F1
(opracowanie własne)

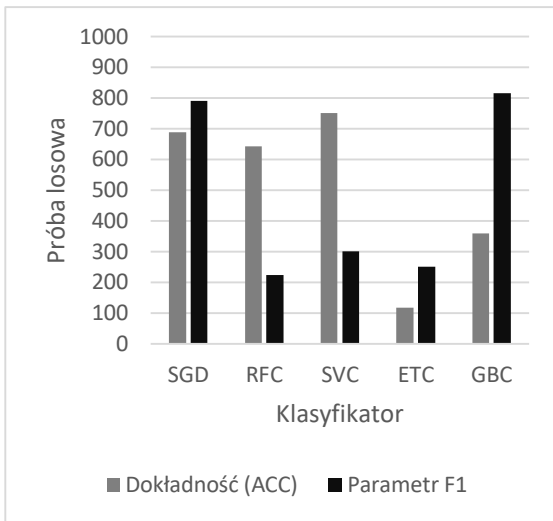
W czasie obliczeń ponownie zbierano informacje na temat numeru próby losowej oraz numeru generacji algorytmu genetycznego, dla których klasyfikatory uzyskiwały najwyższe uśrednione rezultaty wybranych miar oceny dokładności (Tab. 23).

Na wykresach (Rys. 35; Rys. 36) przedstawiono rozkład numerów próby losowej oraz numer generacji dla której klasyfikator uzyskał najwyższy średni wynik parametru oceny dokładności.

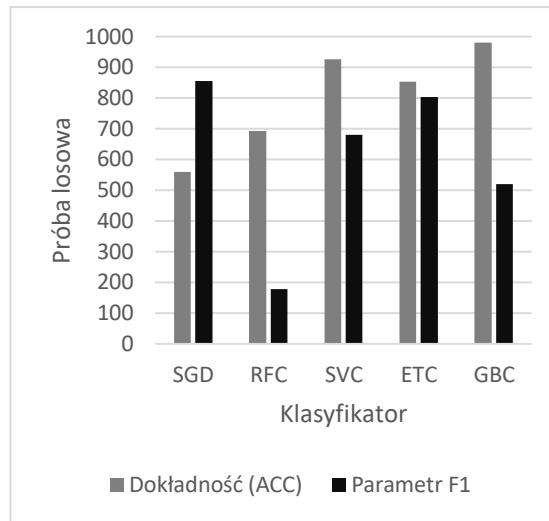
Tab. 23 Numer próby losowej i numer generacji algorytmu genetycznego, dla których wyznaczono najwyższą średnią wartość parametru oceny dokładności klasyfikacji (opracowanie własne)

Klasyfikator	Parametr określający dokładność	7x7 pikseli		14x14 pikseli	
		Numer próby losowej	Numer generacji	Numer próby losowej	Numer generacji
SGD	ACC	687	23	558	24
	F1	789	17	854	28
RFC	ACC	643	28	693	19
	F1	223	46	178	13
SVC	ACC	750	6	925	9
	F1	301	49	680	10
ETC	ACC	117	26	852	12
	F1	250	17	803	12
GBC	ACC	359	22	980	20
	F1	815	11	519	26

(A)

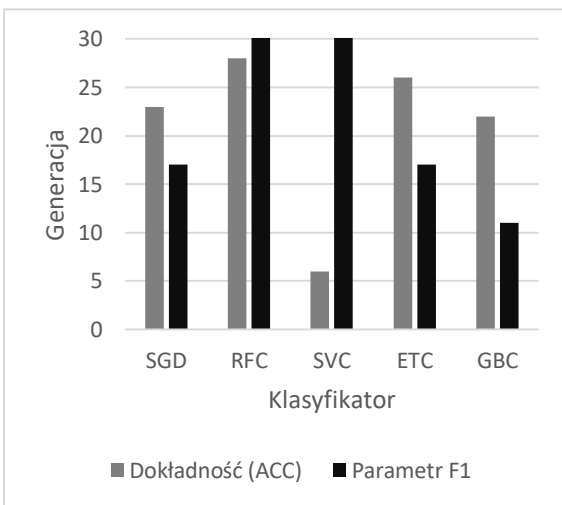


(B)

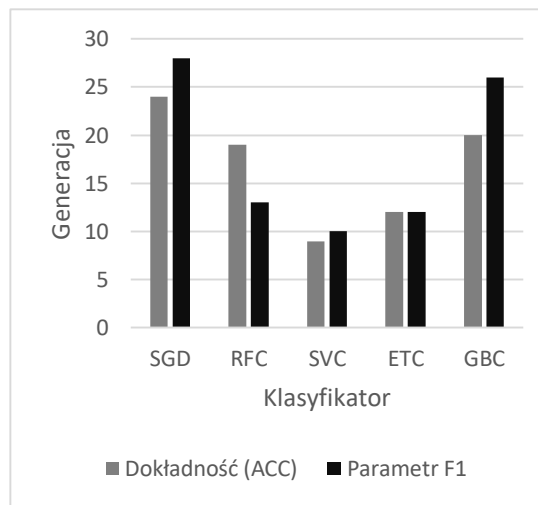


Rys. 35 Numer próby losowej dla której klasyfikator uzyskał najwyższy średni wynik metryk dokładności (A) próbki 7x7 pikseli; (B) próbki 14x14 pikseli (opracowanie własne)

(A)



(B)



Rys. 36 Numer generacji dla której klasyfikator uzyskał najwyższy średni wynik metryk dokładności (A) próbki 7x7 pikseli; (B) próbki 14x14 pikseli (opracowanie własne)

W dwóch tabelach (Tab. 24; Tab. 25) zestawiono wartości środkowe oraz odchylenia dla poszczególnych kanałów spektralnych, wyznaczone przez algorytm genetyczny. W celu zwizualizowania otrzymanych zakresów filtra sporządzono dwa rysunki (Rys. 37; Rys. 38).

W dwóch przypadkach dla miary F1 zakres filtrowania kanału spektralnego został ograniczony do jednej konkretnej liczby – szerokość przedziału wyniosła zero. Był to kanał

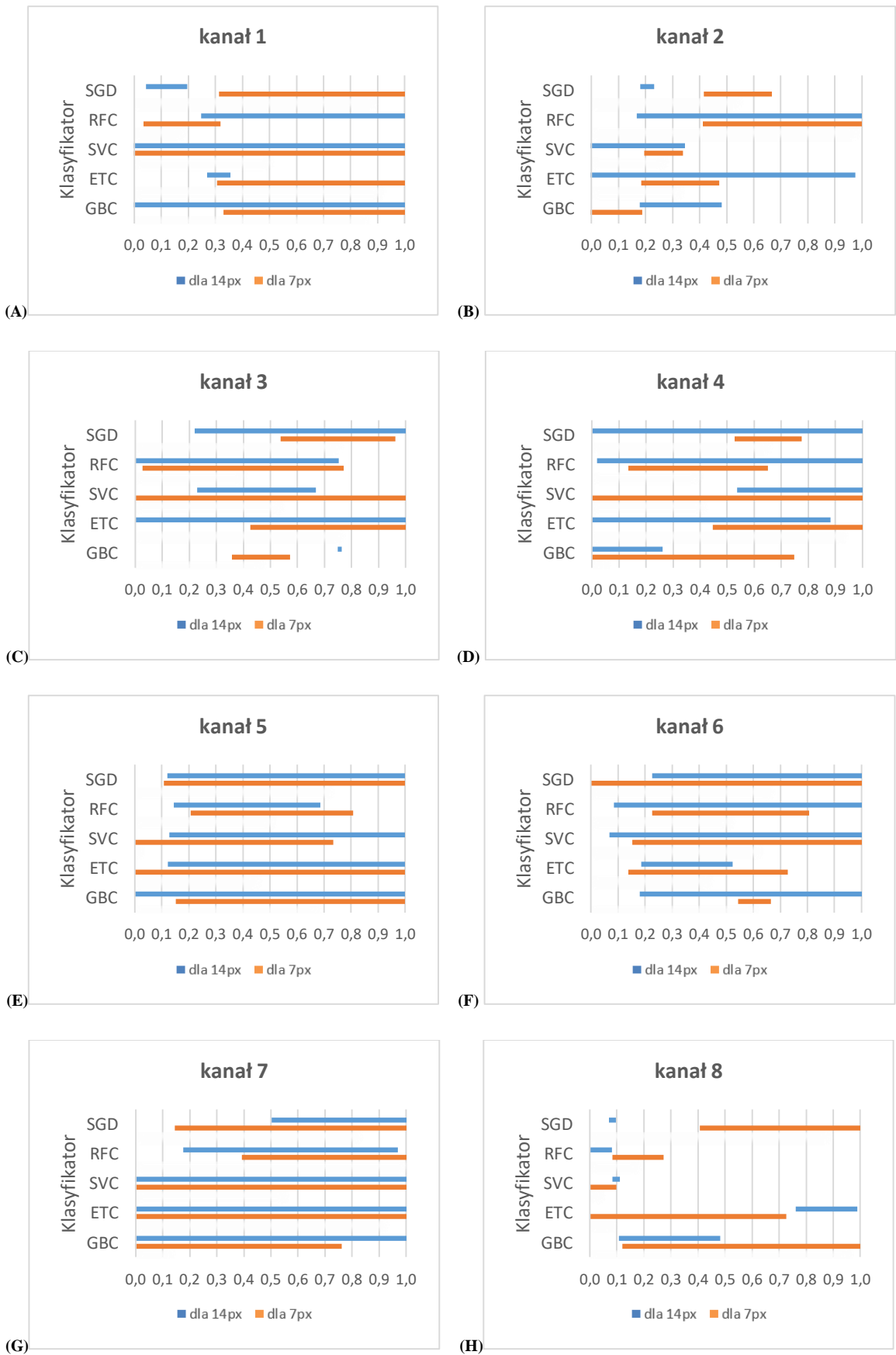
5 dla SVC przy próbkach 7x7 pikseli oraz kanał 8 dla ETC przy próbkach 14x14 pikseli (Tab. 25).

Tab. 24 Wartość środkowa i odchylenie (parametry filtra) dla poszczególnych kanałów spektralnych wyznaczone przez algorytm genetyczny wykorzystujący jako funkcje oceny dokładność ACC (opracowanie własne)

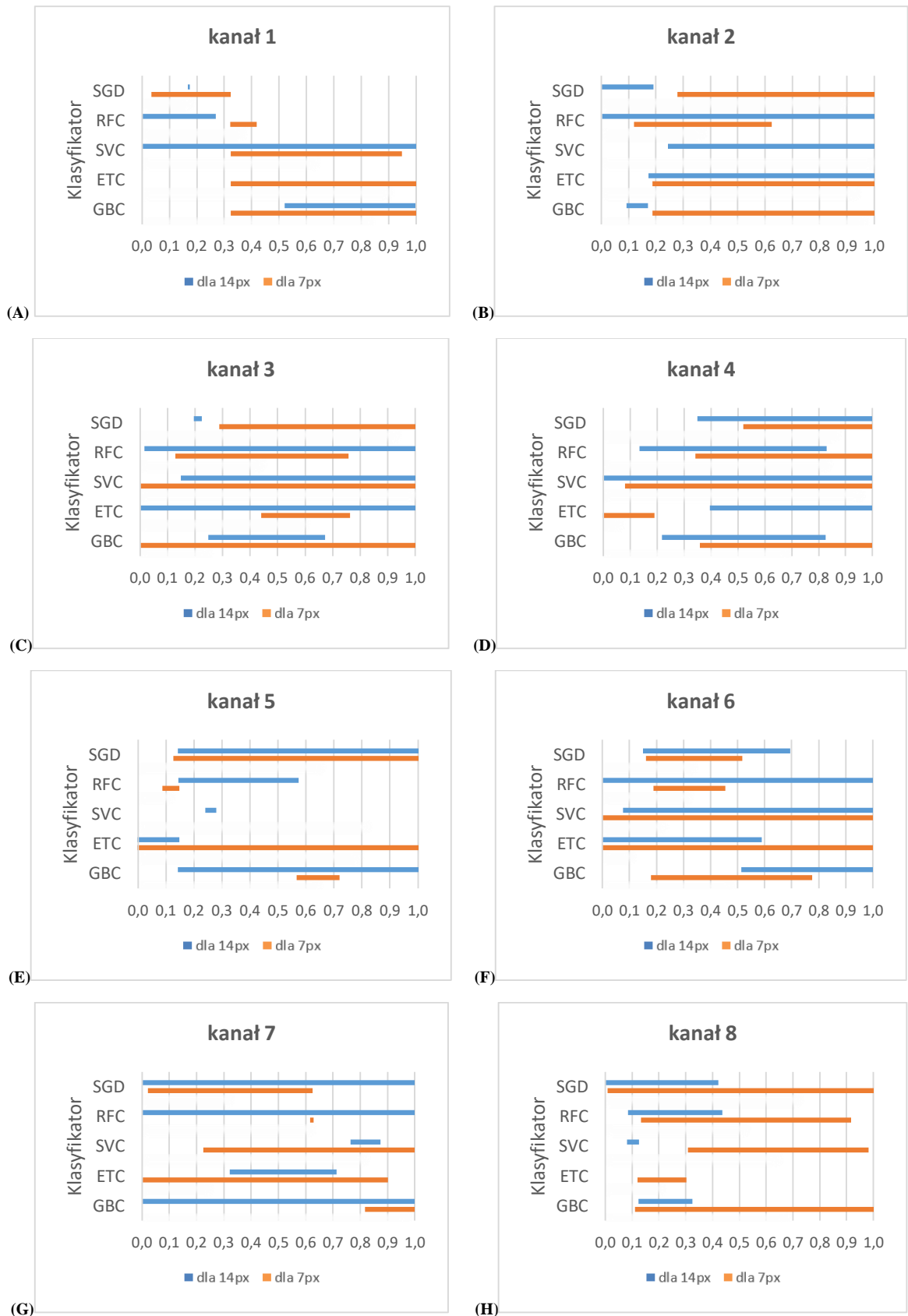
Zestaw próbek	Klasyfikator	Kanał 1		Kanał 3		Kanał 3		Kanał 4		Kanał 5		Kanał 6		Kanał 7		Kanał 8	
		Wartość środkowa	Odchylenie	Wartość środkowa	Odchylenie	Wartość środkowa	Odchylenie	Wartość środkowa	Odchylenie	Wartość środkowa	Odchylenie	Wartość środkowa	Odchylenie	Wartość środkowa	Odchylenie	Wartość środkowa	Odchylenie
7px	SGD	0,88	0,57	0,54	0,13	0,75	0,21	0,65	0,12	0,72	0,61	0,66	0,67	0,82	0,68	0,88	0,47
14px		0,12	0,08	0,21	0,02	0,76	0,54	0,27	0,86	0,89	0,77	0,75	0,52	0,95	0,45	0,08	0,01
7px	RFC	0,18	0,14	0,98	0,57	0,40	0,37	0,39	0,26	0,51	0,30	0,52	0,29	1,00	0,61	0,18	0,09
14px		0,89	0,64	0,77	0,60	0,11	0,64	0,77	0,75	0,42	0,27	0,86	0,77	0,57	0,40	0,02	0,06
7px	SVC	0,50	0,60	0,27	0,07	0,54	0,99	0,41	0,73	0,02	0,71	0,62	0,47	0,55	0,66	0,04	0,06
14px		0,19	0,89	0,01	0,33	0,45	0,22	0,82	0,28	0,74	0,61	0,79	0,72	0,45	0,61	0,10	0,01
7px	ETC	0,91	0,61	0,33	0,14	0,76	0,33	0,93	0,48	0,45	0,75	0,43	0,29	0,92	0,92	0,09	0,64
14px		0,31	0,04	0,19	0,78	0,67	0,78	0,27	0,61	0,76	0,64	0,35	0,17	0,67	0,66	0,88	0,11
7px	GBC	0,85	0,52	0,00	0,19	0,47	0,11	0,05	0,70	0,67	0,52	0,60	0,06	0,13	0,63	0,96	0,84
14px		0,14	1,00	0,33	0,15	0,76	0,01	0,06	0,20	0,39	0,90	0,93	0,75	0,34	0,68	0,29	0,19

Tab. 25 Wartość środkowa i odchylenie (parametry filtra) dla poszczególnych kanałów spektralnych wyznaczone przez algorytm genetyczny wykorzystujący jako funkcje oceny parametr F1 (opracowanie własne)

Zestaw próbek	Klasyfikator	Kanał 1		Kanał 3		Kanał 3		Kanał 4		Kanał 5		Kanał 6		Kanał 7		Kanał 8	
		Wartość środkowa	Odchylenie	Wartość środkowa	Odchylenie	Wartość środkowa	Odchylenie	Wartość środkowa	Odchylenie	Wartość środkowa	Odchylenie	Wartość środkowa	Odchylenie	Wartość środkowa	Odchylenie	Wartość środkowa	Odchylenie
7px	SGD	0,18	0,15	0,89	0,62	0,93	0,65	0,87	0,35	0,65	0,53	0,34	0,18	0,32	0,30	0,72	0,71
14px		0,17	0,00	0,00	0,19	0,21	0,01	0,82	0,47	1,00	0,86	0,42	0,27	0,37	0,68	0,08	0,34
7px	RFC	0,37	0,05	0,37	0,25	0,44	0,31	0,83	0,49	0,12	0,03	0,32	0,13	0,62	0,01	0,52	0,39
14px		0,13	0,14	0,71	0,71	0,91	0,89	0,48	0,35	0,36	0,21	0,68	0,79	0,32	0,86	0,26	0,18
7px	SVC	0,64	0,31	1,00	0,00	1,00	1,00	0,96	0,87	0,82	0,00	0,23	0,83	0,82	0,59	0,64	0,34
14px		0,55	0,64	0,90	0,65	0,89	0,74	0,63	0,76	0,26	0,02	0,61	0,54	0,82	0,05	0,10	0,02
7px	ETC	0,90	0,58	0,94	0,75	0,60	0,16	0,06	0,13	0,79	0,97	0,11	0,94	0,25	0,65	0,21	0,09
14px		0,58	0,00	0,74	0,56	0,35	0,90	0,84	0,45	0,03	0,12	0,01	0,58	0,52	0,20	0,71	0,00
7px	GBC	0,83	0,51	0,97	0,78	0,25	0,82	0,69	0,33	0,64	0,08	0,48	0,30	0,97	0,15	0,67	0,56
14px		0,76	0,24	0,13	0,04	0,46	0,21	0,52	0,30	0,60	0,46	0,93	0,41	0,23	1,00	0,22	0,10



Rys. 37 Zakresy przedziałów filtrowania dla poszczególnych kanałów spektralnych wyznaczone przez algorytm genetyczny wykorzystujący jako funkcje oceny dokładność ACC (opracowanie własne)



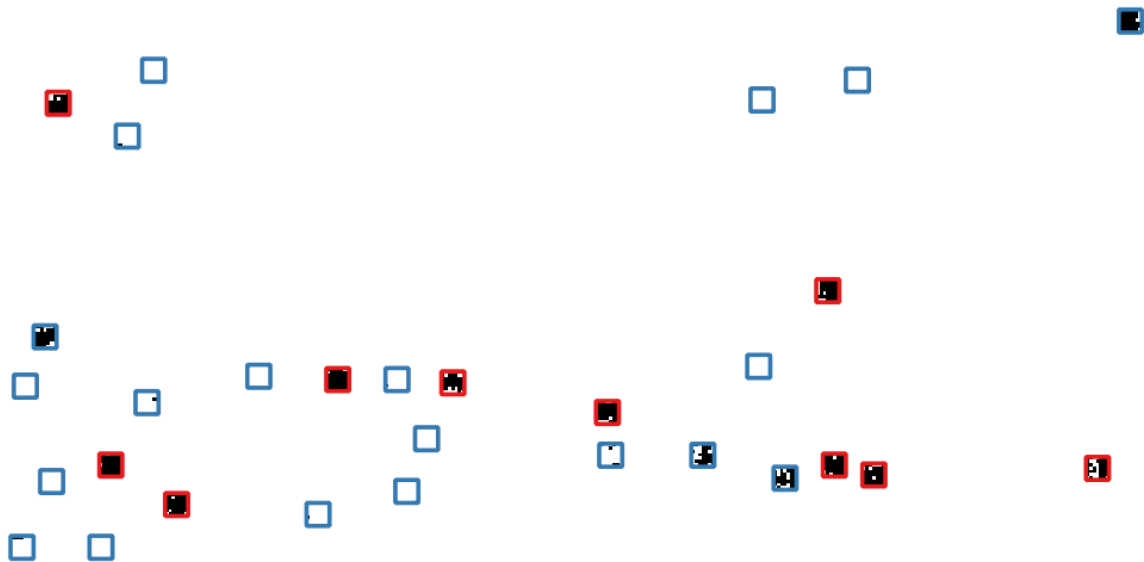
Rys. 38 Zakresy przedziałów filtrowania dla poszczególnych kanałów spektralnych wyznaczone przez algorytm genetyczny wykorzystujący jako funkcje oceny parametr F1 (opracowanie własne)

7.4 KLASYFIKACJA W PROGRAMIE SAGA – STRATEGIA III

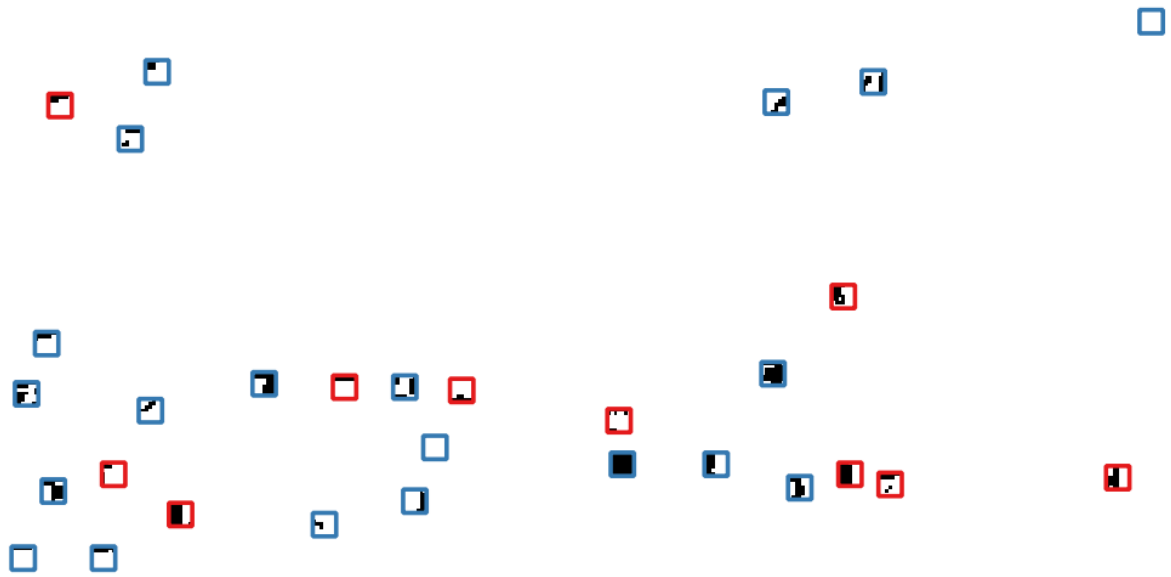
Fragmenty obrazów klasyfikacji z zaznaczonymi próbkami dachowymi z oprogramowania SAGA przedstawiono na rysunkach (Rys. 39; Rys. 40; Rys. 41; Rys. 42). Otrzymane macierze błędów oraz wyznaczone miary dokładności zaprezentowano w tabelach (Tab. 26; Tab. 27).

Wyniki oceny klasyfikacji wykonane przy użyciu oprogramowania SAGA są znacząco niższe w porównaniu do klasyfikacji danych przefiltrowanych. Wyjątek stanowi klasyfikacja próbek 7x7 pikseli klasyfikatorem RFC, dla której miary oceny klasyfikacji są na równie wysokim poziomie.

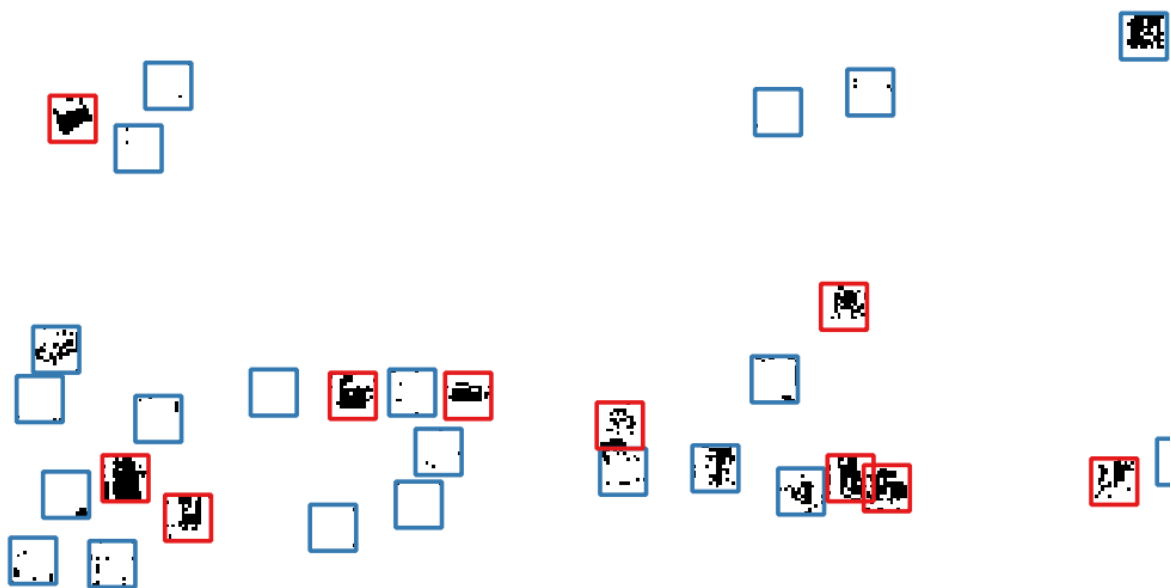
W przypadku obu zestawów danych (7x7 oraz 14x14 pikseli) zastosowano analogiczny podział danych wejściowych na dane uczące i testowe. Można więc przyrównać bezpośrednio wyniki klasyfikacji dla próbek przedstawiających wyłącznie fragmenty dachów (7x7 pikseli) oraz próbek o bardziej zróżnicowanej informacji spektralnej, przedstawiającej również obszar wokół budynku (14x14 pikseli). Klasyfikator RFC poradził sobie zdecydowanie lepiej przy próbkach o mniejszym zróżnicowaniu spektralnym, natomiast klasyfikator SVC uzyskał wyższe wyniki dokładności dla próbek 14x14 pikseli.



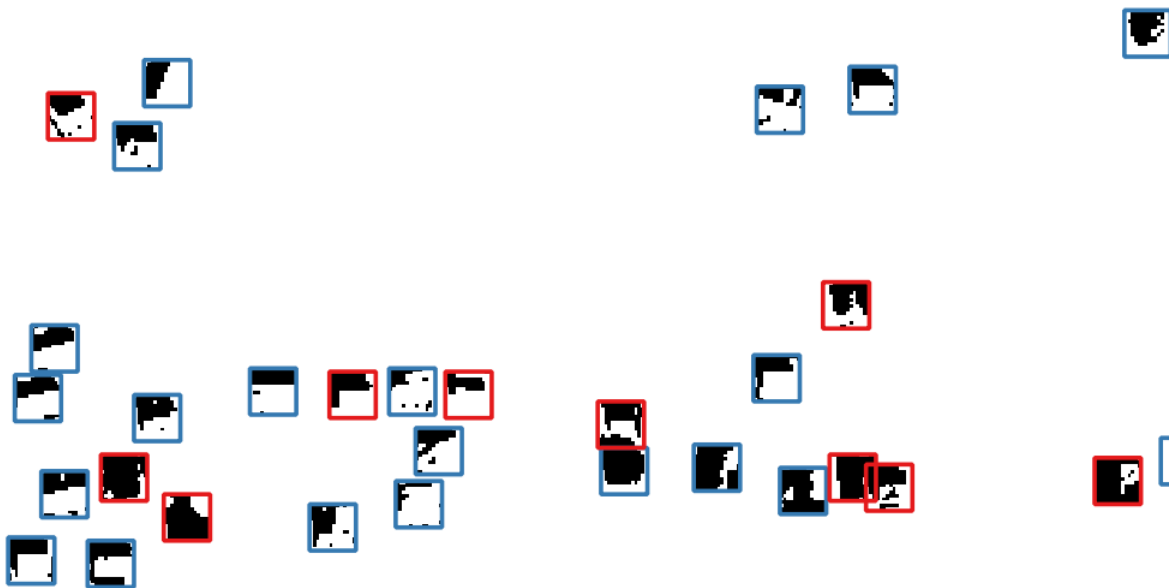
Rys. 39 Fragment wyników klasyfikacji RFC z zaznaczonymi próbkami dachowymi 7x7 pikseli
sklasyfikowane piksele: kolor czarny – azbest, kolor biały – inne pokrycia dachowe;
rzeczywista próbka dachu: ramka czerwona – azbest, ramka niebieska – inne pokrycie dachowe
(opracowanie własne)



Rys. 40 Fragment wyników klasyfikacji SVC z zaznaczonymi próbkami dachowymi 7x7 pikseli
 sklasyfikowane piksele: kolor czarny – azbest, kolor biały – inne pokrycia dachowe;
 rzeczywista próbka dachu: ramka czerwona – azbest, ramka niebieska – inne pokrycie dachowe
 (opracowanie własne)



Rys. 41 Fragment wyników klasyfikacji RFC z zaznaczonymi próbkami dachowymi 14x14 pikseli
 sklasyfikowane piksele: kolor czarny – azbest, kolor biały – inne pokrycia dachowe;
 rzeczywista próbka dachowa: ramka czerwona – azbest, ramka niebieska – inne pokrycie dachowe
 (opracowanie własne)



Rys. 42 Fragment wyników klasyfikacji SVC z zaznaczonymi próbkami dachowymi 14x14 pikseli
 sklasyfikowane piksele: kolor czarny – azbest, kolor biały – inne pokrycia dachowe;
 rzeczywista próbka dachowa: ramka czerwona – azbest, ramka niebieska – inne pokrycie dachowe
 (opracowanie własne)

Tab. 26 Macierze błędów dla klasyfikacji wykonanej w oprogramowaniu SAGA (opracowanie własne)

(A)		7x7 pikseli, RFC			(B)		14x14 pikseli, RFC		
Prognoza		Klasa	Rzeczywistość		Prognoza		Klasa	Rzeczywistość	
			1	0				1	0
	1		29	11		1	2	0	
	0		9	82		0	36	93	

(C)		7x7 pikseli, SVC			(D)		14x14 pikseli, SVC		
Prognoza		Klasa	Rzeczywistość		Prognoza		Klasa	Rzeczywistość	
			1	0				1	0
	1		4	46		1	19	41	
	0		34	47		0	19	52	

Tab. 27 Wyniki oceny klasyfikacji w oprogramowaniu SAGA (opracowanie własne)

Klasyfikator	Parametr określający dokładność	7 x 7 pikseli	14 x 14 pikseli
RFC	ACC	0,84733	0,72519
	F1	0,74359	0,10000
SVC	ACC	0,38931	0,54198
	F1	0,09091	0,38776

7.5 WYNIKI OCENY KLASYFIKACJI NA DANYCH TESTOWYCH

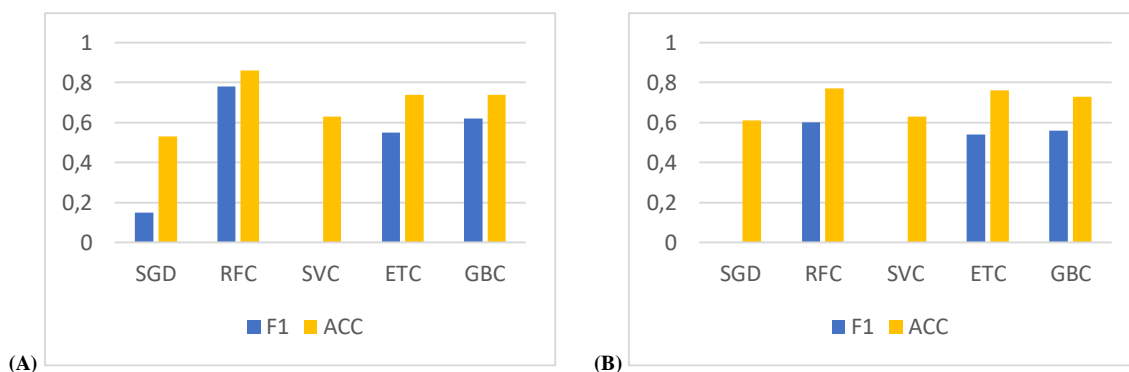
Skuteczność proponowanego podejścia zweryfikowano na niezależnych danych testowych, które nie brały udziału w eksperymentach opisanych w poprzednim rozdziale (6.5.1; 6.5.2; Rys. 24). Do oceny dokładności wykorzystano przefiltrowane dane testowe wg macierzy filtrujących wyznaczonych przez algorytm genetyczny wg strategii I (Tab. 18; Tab. 19) oraz strategii II (Tab. 24; Tab. 25). Otrzymane wartości miar oceny dokładności dla próbek 7x7 pikseli zestawiono w tabeli (Tab. 28). Miary precyzji, czułości oraz F1 odnoszą się do klasy dachów azbestowych.

Można zauważyć, że zadowalające wartości wszystkich miar otrzymano tylko dla dwóch klasyfikatorów: RFC oraz GBC. Klasyfikator RFC był skuteczny dla dwóch filtrów, natomiast GBC tylko dla jednego (wartości wyróżnione w Tab. 28). Pozostałe filtry nie sprawdziły się podczas klasyfikacji danych testowych. Najniższe wartości miar uzyskano dla klasyfikatora SVC, który był całkowicie niewrażliwy na zmianę filtru danych źródłowych.

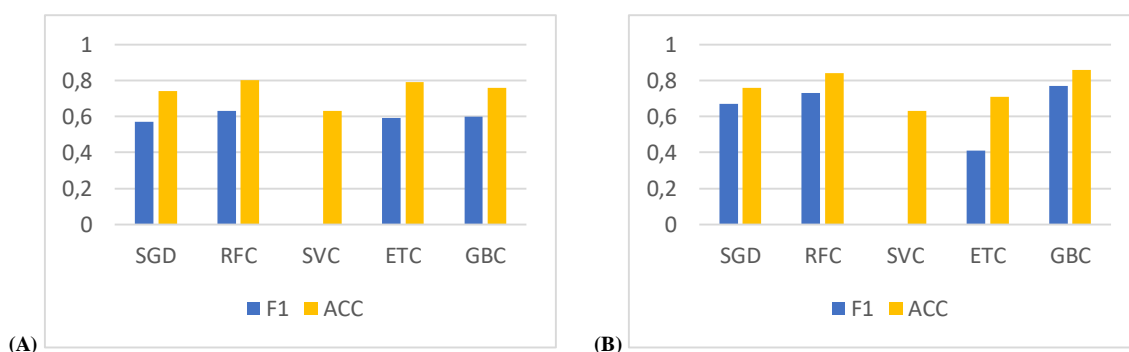
W celu zobrazowania otrzymanych wartości miar oceny dokładności wykonano wykresy (Rys. 43; Rys. 44). Dodatkowo w tabeli (Tab. 29) przedstawiono macierze błędów oraz miary oceny dokładności dla obu klas (dachy azbestowe: 1, inne pokrycia dachowe: 0).

Tab. 28 Ocena dokładności klasyfikacji (7x7 pikseli) na danych testowych z wykorzystaniem uzyskanych filtrów (opracowanie własne)

Ustalenie filtru wg strategii	Funkcja oceny	Klasyfikator	Miary oceny dokładności			
			Precyzja	Czułość	F1	ACC
I	ACC	SGD	0.23	0.12	0.15	0.53
		RFC	0.90	0.69	0.78	0.86
		SVC	1	0	0	0.63
		ETC	0.79	0.42	0.55	0.74
		GBC	0.68	0.58	0.62	0.74
	F1	SGD	0	0	0	0.61
		RFC	0.86	0.46	0.60	0.77
		SVC	1	0	0	0.63
		ETC	0.91	0.38	0.54	0.76
		GBC	0.71	0.46	0.56	0.73
II	ACC	SGD	0.75	0.46	0.57	0.74
		RFC	1	0.46	0.63	0.80
		SVC	1	0	0	0.63
		ETC	1	0.42	0.59	0.79
		GBC	0.76	0.50	0.60	0.76
	F1	SGD	0.68	0.65	0.67	0.76
		RFC	1	0.58	0.73	0.84
		SVC	1	0	0	0.63
		ETC	0.88	0.27	0.41	0.71
		GBC	0.94	0.65	0.77	0.86



Rys. 43 Wartości miar oceny dokładności dla próbek 7x7 pikseli z wykorzystaniem filtrów wg strategii I (A) funkcja oceny AG: ACC; (B) funkcja oceny AG: F1 (opracowanie własne)



Rys. 44 Wartości miar oceny dokładności dla próbek 7x7 pikseli z wykorzystaniem filtrów wg strategii II (A) funkcja oceny AG: ACC; (B) funkcja oceny AG: F1 (opracowanie własne)

Tab. 29 Zestawienie wybranych macierzy błędów oraz miar oceny dokładności dla próbek 7x7 pikseli (opracowanie własne)

Filtr wg strategii I dla klasyfikatora RFC, przy funkcji oceny ACC				
Macierz błędów	Predykcja			
	Klasa	0	1	
	Realnie			
	0	42	2	
	1	8	18	
Klasa	Miara oceny dokładności			
	Precyzja	Czułość	F1	ACC
0	0,84	0,95	0,89	0,86
1	0,90	0,69	0,78	

Filtr wg strategii II dla klasyfikatora RFC, przy funkcji oceny F1				
Macierz błędów	Predykcja			
	Klasa	0	1	
	Realnie			
	0	44	0	
	1	11	15	
Klasa	Miara oceny dokładności			
	Precyzja	Czułość	F1	ACC
0	0,80	1	0,89	0,84
1	1	0,58	0,73	

Filtr wg strategii II dla klasyfikatora GBC, przy funkcji oceny F1				
Macierz błędów	Predykcja			
	Klasa	0	1	
	Realnie			
	0	43	1	
	1	9	17	
Klasa	Miara oceny dokładności			
	Precyzja	Czułość	F1	ACC
0	0,83	0,98	0,90	0,86
1	0,94	0,65	0,77	

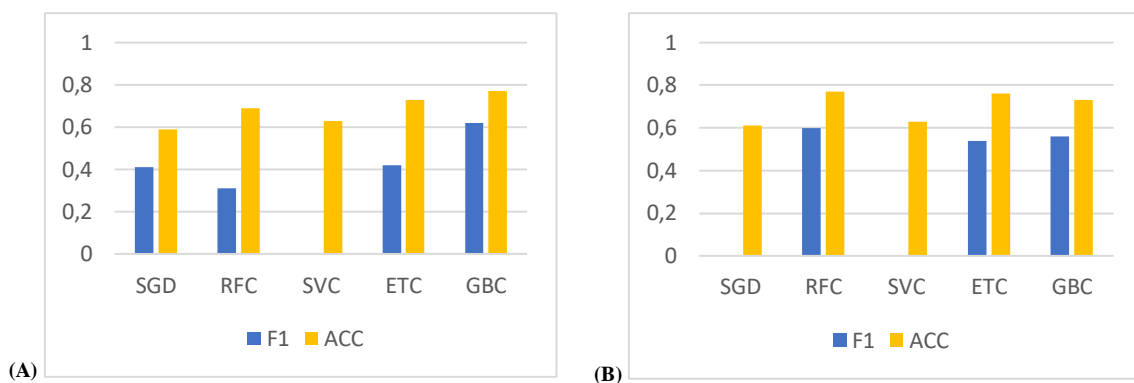
Miary oceny dokładności klasyfikacji wykonanych na przefiltrowanych danych 14x14 pikseli wg macierzy uzyskanych podczas wcześniejszych badań zestawiono w tabeli (Tab. 30). Ponownie precyzja, czułość oraz F1 dotyczyły klasy dachów azbestowych.

Najwyższe wartości miar otrzymano również dla klasyfikacji wykonanej klasyfikatorami RFC oraz GBC. Dla próbek obejmujących otoczenie budynków tylko dwa filtry pozwoliły na efektywną klasyfikację (Tab. 30). Oba filtry były wyznaczone podczas krosvalidacji (strategia II), gdzie funkcją oceny klasyfikacji była miara F1 dla dachów azbestowych. Powtórnie wyniki oceny dokładności dla klasyfikatora SVC były najniższe. Wyjątkowo wysokie miary oceny dokładności otrzymano dla klasyfikatora SGD dla filtrów pozyskanych podczas II strategii wg F1.

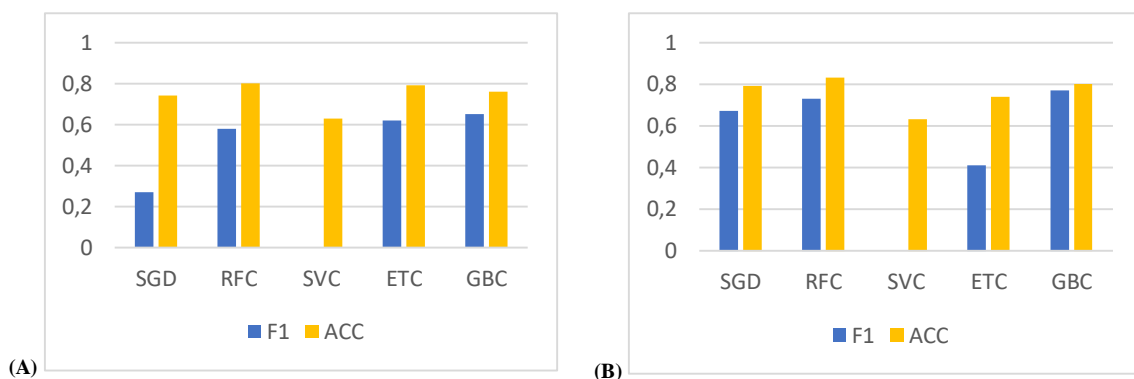
Wykresy (Rys. 47; Rys. 48) przedstawiają uzyskane wartości miar oceny dokładności. Dodatkowo w tabeli (Tab. 31) przedstawiono macierze błędów oraz miary oceny dokładności dla obu klas (dachy azbestowe: 1, inne pokrycia dachowe: 0) wyznaczone dla najskuteczniejszych klasyfikatorów.

Tab. 30 Ocena dokładności klasyfikacji (14x14 pikseli) na danych testowych z wykorzystaniem uzyskanych filtrów (opracowanie własne)

Ustalenie filtru wg strategii	Funkcja oceny	Klasyfikator	Miary oceny dokładności			
			Precyzja	Czułość	F1	ACC
I	ACC	SGD	0,43	0,38	0,41	0,59
		RFC	0,83	0,19	0,31	0,69
		SVC	1	0	0	0,63
		ETC	1	0,27	0,42	0,73
		GBC	0,81	0,50	0,62	0,77
	F1	SGD	0,3	0,12	0,17	0,57
		RFC	1	0,27	0,42	0,73
		SVC	1	0	0	0,63
		ETC	0,80	0,31	0,44	0,71
		GBC	0,69	0,35	0,46	0,70
II	ACC	SGD	0,45	0,19	0,27	0,61
		RFC	0,92	0,42	0,58	0,77
		SVC	1	0	0	0,63
		ETC	0,92	0,46	0,62	0,79
		GBC	0,82	0,54	0,65	0,79
	F1	SGD	0,87	0,50	0,63	0,79
		RFC	1	0,54	0,70	0,83
		SVC	1	0	0	0,63
		ETC	0,90	0,35	0,50	0,74
		GBC	0,77	0,65	0,71	0,80



Rys. 45 Wartości miar oceny dokładności dla próbek 7x7 pikseli z wykorzystaniem filtrów wg strategii I (A) funkcja oceny AG: ACC; (B) funkcja oceny AG: F1 (opracowanie własne)



Rys. 46 Wartości miar oceny dokładności dla próbek 7x7 pikseli z wykorzystaniem filtrów wg strategii II (A) funkcja oceny AG: ACC; (B) funkcja oceny AG: F1 (opracowanie własne)

Tab. 31 Zestawienie wybranych macierzy błędów oraz miar oceny dokładności dla próbek 14x14 pikseli (opracowanie własne)

Filtr wg strategii II dla klasyfikatora RFC , przy funkcji oceny F1					
Macierz błędów	Predykcja				
	Klasa	0	1		
	Realnie	0	44	0	
	1	12	14		
Klasa	Miara oceny dokładności				
	Precyzja	Czułość	F1	ACC	
	0	0,79	1	0,88	0,83
	1	1	0,54	0,70	

Filtr wg strategii II dla klasyfikatora GBC , przy funkcji oceny F1					
Macierz błędów	Predykcja				
	Klasa	0	1		
	Realnie	0	39	5	
	1	9	17		
Klasa	Miara oceny dokładności				
	Precyzja	Czułość	F1	ACC	
	0	0,81	0,89	0,85	0,80
	1	0,77	0,65	0,71	

8 ANALIZA I DYSKUSJA WYNIKÓW

Rozdział składa się z trzech części. W pierwszej dokonano porównania wyników klasyfikacji otrzymanych podczas eksperymentów, w czasie których wyznaczano filtry. W drugiej części porównano miary oceny dokładności klasyfikacji dla danych walidacyjnych oraz danych testowych. Na końcu odniesiono się do rezultatów opisanych w literaturze przedmiotu.

8.1 PORÓWNANIE WYNIKÓW EKSPERYMENTÓW OBLICZENIOWYCH

Wyniki badań prowadzonych wg wszystkich trzech strategii zestawiono w tabelach (Tab. 32; Tab. 33). Dla strategii I pokazano wartości parametrów uzyskane podczas uczenia klasyfikatorów oraz podczas ponownej niezależnej oceny klasyfikacji przy użyciu krosvalidacji stratyfikowanej.

Tab. 32 Zestawienie oceny dokładności klasyfikacji wszystkich strategii dla próbek 7x7 pikseli (opracowanie własne)

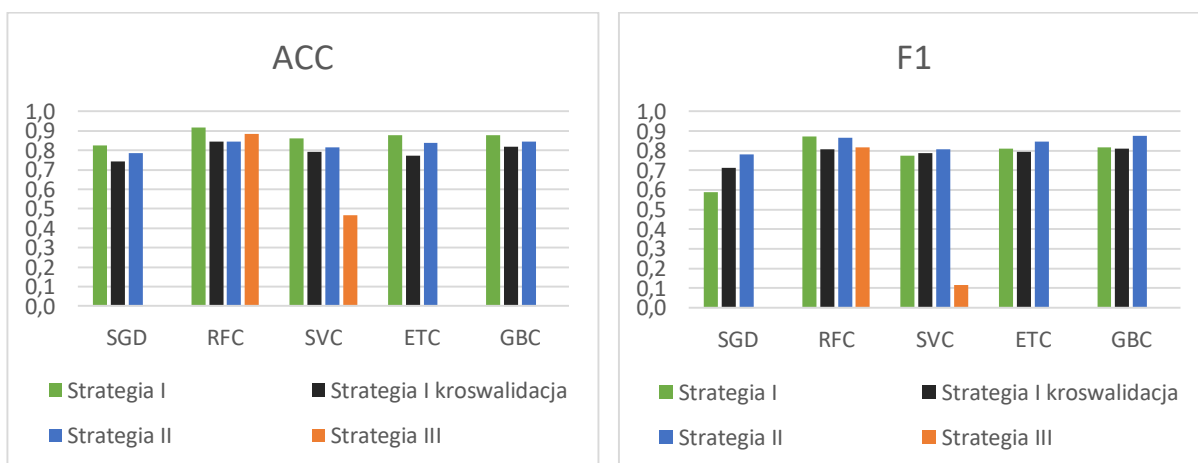
Klasyfikator	Strategia I		Strategia I krosvalidacja		Strategia II		Strategia III	
	Parametr określający dokładność							
	ACC	F1	ACC	F1	ACC	F1	ACC	F1
SGD	0,82443	0,58915	0,74253	0,71264	0,78621	0,78161		
RFC	0,91603	0,87324	0,84368	0,80689	0,84368	0,86437	0,84733	0,74359
SVC	0,86260	0,77612	0,79080	0,78851	0,81379	0,80690	0,38931	0,09091
ETC	0,87786	0,81159	0,77241	0,79310	0,83678	0,84598		
GBC	0,87786	0,81690	0,81839	0,80920	0,84598	0,87586		

Tab. 33 Zestawienie oceny dokładności klasyfikacji wszystkich strategii dla próbek 14x14 pikseli (opracowanie własne)

Klasyfikator	Strategia I		Strategia I krosvalidacja		Strategia II		Strategia III	
	Parametr określający dokładność							
	ACC	F1	ACC	F1	ACC	F1	ACC	F1
SGD	0,81679	0,71111	0,62759	0,60460	0,76092	0,76322		
RFC	0,90076	0,81013	0,70345	0,71264	0,84138	0,81839	0,72519	0,10000
SVC	0,83969	0,65625	0,75632	0,71034	0,76782	0,76782	0,54198	0,38776
ETC	0,87786	0,81690	0,71264	0,71034	0,80690	0,80460		
GBC	0,87023	0,79518	0,74023	0,64368	0,82529	0,81609		

Do wykonania oceny skuteczność klasyfikatorów wybrano dwie miary: dokładność klasyfikacji ACC oraz miara F1 dla klasy azbestowej. Z uwagi na nie zrównoważone zbiory danych wejściowych (125 próbek azbestowych, 310 próbek innych dachów) dokładność ACC może nie zawsze dobrze oceniać zachowanie klasyfikatora względem wykrywania dachów azbestowych. Z uwagi na temat pracy, w którym istotne jest wykrywanie pokryć azbestowych, lepszą miarą oceny było F1 dla dachów azbestowych. Parametr ten łączy dwie miary oceny klasyfikacji binarnej: precyzję oraz czułość. Precyzja wskazuje jak wiele z przewidzianych dachów azbestowych jest faktycznie dachami azbestowymi. Natomiast czułość określa jak wiele rzeczywistych pokryć azbestowych zostało poprawnie sklasyfikowanych przez klasyfikator.

Podczas analiz wyników uzyskanych z wykorzystaniem danych filtrowanych za bardziej wiarygodny można uznać wynik uzyskany podczas obliczeń wykonanych zgodnie z przyjętą strategią II, gdzie podczas ustalania wartości dla filtra wykorzystywano 3-krotną krosvalidację. Krosvalidacja umożliwia lepsze oszacowanie skuteczności klasyfikatora niż tradycyjne podział na zbiór treningowy i walidacyjny, ponieważ wykorzystuje wszystkie dostępne dane do trenowania i testowania modelu (Géron, 2020, Harrison, 2020). W trakcie obliczeń dla próbek 7x7 pikseli otrzymano średnią wartość F1 na poziomie 86% dla klasyfikatora RFC oraz 87% dla klasyfikatora GBC. Podczas gdy dla tych samych próbek bez filtrowania danych spektralnych, uzyskano F1 o wartości 74% dla RFC (Rys. 47). Dla próbek 14x14 pikseli średnia wartość parametru F1 spadła, chociaż dla klasyfikacji wykorzystującej filtr danych źródłowych, miara F1 pozostała na poziomie ponad 81%, natomiast dla klasyfikacji opartej wyłącznie na danych spektralnych spadła do zaledwie 10% dla RFC (strategia III) (Rys. 48).

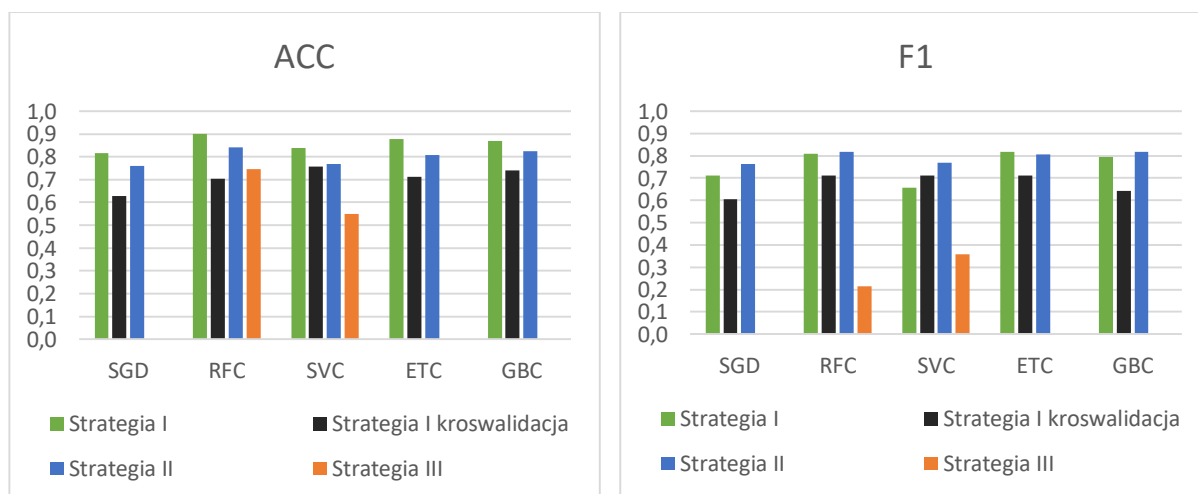


Rys. 47 Wartości miar oceny dokładności klasyfikatorów dla próbek 7x7 pikseli (opracowanie własne)

W przypadku próbek 7x7 pikseli najwyższe wyniki oceny klasyfikacji uzyskano dla klasyfikatora RFC (Rys. 47). Również warto zwrócić uwagę na klasyfikator GBC, gdyż przeprowadzona klasyfikacja dla strategii II była na równie wysokim poziomie co RFC. Niestety w programie SAGA klasyfikator GBC nie jest dostępny i nie można dokonać porównania wyników.

Dla próbek 14x14 pikseli otrzymywane wyniki oceny klasyfikacji były niższe niż dla próbek 7x7 pikseli (Tab. 32; Tab. 33). Można wnioskować, że zarejestrowane otoczenie wokół dachu ma negatywny wpływ na przebieg klasyfikacji. Można wskazać jeden wyjątek, klasyfikator SVC zachował się w odwrotny sposób dla strategii III (Rys. 47; Rys. 48). Jednak wyniki działania klasyfikatora SVC dla obu zestawów próbek były na niskim poziomie. W przypadku klasyfikacji z użyciem filtrowania danych spadek parametrów oceny dokładności dla próbek 14x14 pikseli jest widoczny, przy czym otrzymane wyniki dokładności ACC oraz miary F1 nadal są na zadowalającym poziomie (Rys. 47; Rys. 48).

Dla próbek 7x7 pikseli dla wszystkich trzech strategii działanie klasyfikatora RFC jest na akceptowalnym poziomie (Rys. 47; Rys. 48). Natomiast w przypadku próbek 14x14 pikseli klasyfikatory działają na odpowiednim poziomie dokładności wyłącznie przy zastosowaniu filtrowania danych wejściowych.



Rys. 48 Wartości miar oceny dokładności klasyfikatorów dla próbek 14x14 pikseli (opracowanie własne)

Podczas oceny klasyfikatorów wykorzystujących filtry wyznaczone w strategii I wykorzystano standardową oraz stratyfikowaną walidację krzyżową. Można zauważyć, że dla większości klasyfikatorów krosvalidacja z losowym podziałem danych uzyskiwała

niższe lub na zbliżonym poziomie wartości paramentów oceny klasyfikatorów jak krosvalidacja stratyfikowana. Wyjątkiem jest klasyfikator SGD, dla którego wartości parametrów dla próbek 7x7 pikseli oraz 14x14 pikseli prezentowały się w odwrotny sposób, tzn. dla próbek 7x7 pikseli wyższe wartości miar ACC oraz F1 otrzymano dla krosvalidacji stratyfikowanej, natomiast dla próbek 14x14 pikseli wyższe wartości miar ACC oraz F1 otrzymano dla krosvalidacji losowej. Dodatkowo dla klasyfikatora ETC (zestaw 7x7 pikseli) wartości ACC oraz F1 są nieznacznie wyższe przy krosvalidacji losowej. W związku z tym, można stwierdzić, że stosowanie stratyfikowanej walidacji krzyżowej dla nie zrównoważonych zbiorów danych wejściowych przeważnie pozwala uzyskać wyższe wyniki oceny dokładności klasyfikatorów.

W celu doboru odpowiedniego filtru stosowano algorytm genetyczny, którego zadaniem było ustalenie skutecznych wartości filtru. W przypadku eksperymentów obliczeniowych, gdzie dane źródłowe podzielono jednokrotnie na dane uczące i walidacyjne (strategia I), algorytm genetyczny pozwalał dopasować filtry, dla których parametry oceny dokładności wzrastały o kilka punktów procentowych względem filtrów losowanych. Podczas trenowania klasyfikatorów przy użyciu 3 – krotnej krosvalidacji (strategia II) również występował przyrost wartości parametrów oceny dokładności, jednak był on na nieco niższym poziomie. Można domniemywać, że w przypadku strategii I wyższy przyrost parametrów oceny dokładności wynikał z dopasowywania filtrów do stałego zestawu trenującego i oceniającego. W związku z wykorzystaniem walidacji krzyżowej przy strategii II, podczas poszukiwania odpowiednich macierzy filtrujących, uwzględniano wszystkie dane źródłowe. Na pewno pozwoliło to otrzymać bardziej uniwersalne i mniej przeuczone klasyfikatory. Po uwzględnieniu powyższych rozważań można stwierdzić, że zastosowanie algorytmu genetycznego jest działaniem efektywnym i przynosi dobre rezultaty.

Dodatkowo, algorytm genetyczny przekształca filtr, tak aby uzyskane dane były jak najbardziej odpowiednie do klasyfikacji wybranym klasyfikatorem. W efekcie otrzymane wyniki wartości ACC oraz F1 dla strategii I oraz II dla wszystkich klasyfikatorów nie różnią się znacząco (Rys. 47; Rys. 48).

Dla obu zestawów próbek najniższe wyniki otrzymano przy stosowaniu klasyfikatora SGD oraz SVC. Natomiast klasyfikatory oparte na losowych drzewach decyzyjnych uzyskiwały zadowalające wyniki.

8.2 PORÓWNANIE MIAR OCENY DOKŁADNOŚCI NA DANYCH WALIDACYJNYCH ORAZ TESTOWYCH

W czasie eksperymentów obliczeniowych wg strategii I oraz II wyznaczano na danych walidacyjnych wartości miar ACC oraz F1 niezależnie dla różnych filtrów. Natomiast testowanie klasyfikatorów opierało się na ponownie wytrenowanych modelach na danych treningowych (435 próbek dla miejscowości Krzczonów) przefiltrowanych wg otrzymanych macierzy filtrujących. Następnie prowadzono predykcję wyników dla danych testowych (70 próbek z miejscowości przyległych), na podstawie której wyznaczano miary oceny dokładności. W związku z tym, dla konkretnego filtru i klasyfikatora można porównać wartości pojedynczych miar oceny dokładności (ACC lub F1) na danych walidacyjnych i testowych.

W tabelach (Tab. 34; Tab. 35) zestawiono wyniki otrzymane dla najskuteczniejszych klasyfikatorów na danych testowych. W przypadku obu zestawów próbek (7x7 oraz 14x14 pikseli) były to klasyfikatory RFC oraz GBC. Wyznaczone macierze filtrujące dane wskazano czterokrotnie na podstawie miary F1 podczas strategii II z wykorzystaniem krosvalidacji. Natomiast tylko jeden filtr (dla próbek 7x7 pikseli) wskazano w oparciu o dokładność ACC wg strategii I, gdzie podział danych treningowych oraz walidacyjnych był stały. Jednak wartości miar oceny dokładności dla tego filtru były najwyższe. W związku z powyższym nie można jednoznacznie wskazać miary oceny dokładności, która powinna być wykorzystywana podczas poszukiwania odpowiednich wartości filtrujących dane źródłowe.

Miary uzyskane na danych testowych spadły względem miar uzyskanych na danych walidacyjnych. Jednak otrzymane wartości są nadal na zadowalającym poziomie.

Tab. 34 Wybrane miary oceny dokładności klasyfikacji z wykorzystaniem filtrowania danych dla próbek 7x7 pikseli (opracowanie własne)

Klasyfikator	Miara	Eksperyment obliczeniowy		Test
RFC	ACC	Strategia I	0,92	0,86
RFC	F1	Strategia II	0,86	0,73
GBC	F1	Strategia II	0,88	0,77

Tab. 35 Wybrane miary oceny dokładności klasyfikacji z wykorzystaniem filtrowania danych dla próbek 14x14 pikseli (opracowanie własne)

Klasyfikator	Miara	Eksperyment obliczeniowy		Test
RFC	F1	Strategia II	0,82	0,70
GBC	F1	Strategia II	0,82	0,71

8.3 PORÓWNANIE OTRZYMANYCH WYNIKÓW Z LITERATURĄ PRZEDMIOTU

Przygotowano zestawienie miar dokładności dla klasyfikatora RFC oraz SVM (SVC) otrzymanych podczas własnych eksperymentów obliczeniowych oraz występujących w literaturze przedmiotu (Tab. 36).

Do porównania wybrano najwyższe wyniki oceny dokładności. W większości prac używano innych nazw miar oceny klasyfikatorów (Tab. 36). Przy czym parametr OA oznacza dokładność całkowitą i w przypadku klasyfikacji binarnej jest tym samym parametrem, który był stosowany w niniejszej pracy pod nazwą ACC. Miara PA jest dokładnością producenta i wyznacza się ją w ten sam sposób co czułość. Miara UA to dokładność użytkownika i jej odpowiednikiem jest precyzja (Géron, 2020; Lillesand i in., 2015). Miary F1 (Tab. 36) zostały obliczone wg wzoru podanego w rozdziale 3.6.

W przytoczonych pracach badawczych wyróżniano zróżnicowaną liczbę klas, w związku z czym nie można dokonać bezpośredniego porównania dokładności ogólnej zestawionej w tabeli (Tab. 36). Jedynie w badaniu Kaplan i in. (2023) oraz Yu i in. (2022), podobnie jak w badaniach własnych, wykonywano klasyfikację binarną. Jednak w przypadku Yu i in. 2022 wynik dokładności całkowitej podano dopiero po klasyfikacji CNN, natomiast w Kaplan i in. 2023 posłużono się dokładnością zrównoważoną dla klasy dachów azbestowych, wyznaczoną jako średnią wartość PA oraz UA.

Klasyfikator RFC pozwolił uzyskać bardzo wysokie miary dokładności we wszystkich pracach badawczych. W pracy Abriha i in. (2018) wynik F1 był bliski wartości 100%, jednak należy zwrócić uwagę, że w tych badaniach wykorzystano dodatkowe dane: wskaźnik NDVI oraz modele DTM oraz DSM. Ponadto Abriha i in. (2018) jak również Tommasini i in. (2019) wykonali klasyfikację pikselową, więc trudno bezpośrednio porównać te wyniki z własnymi. Z kolei w publikacji Gibril i in. (2017) przedstawiono wyniki klasyfikacji obiektowej, za pomocą której wyróżniono 9 klas pokrycia terenu. Chociaż dokładność całkowita przekroczyła 82%, to miara F1 dla klasy azbestowej wynosiła 72%. W pracy Kaplan i in. (2023) otrzymano wysoką wartość F1 dla RFC, na poziomie 88%. Trzeba jednak mieć na uwadze, że do przetwarzania wykorzystano obraz o wysokiej rozdzielczości przestrzennej (52cm), na podstawie którego wyznaczano dodatkowe obrazy teksturowe. Dla przytoczonego wyniku klasyfikacja opierała się na 53 warstwach. Na tym tle można stwierdzić, że rezultaty własnych badań (F1 na poziomie 78%) również

potwierdzają skuteczność wykorzystania klasyfikatora RFC do identyfikacji dachów azbestowo – cementowych.

Tab. 36 Zestawienie miar dokładności dla klasyfikatorów RFC oraz SVM (opracowanie własne)

Klasyfikator	Badanie	Obraz źródłowy	Liczba klas	OA [%]	PA [%]	UA [%]	F1 [%]
RFC	Abriha i in. 2018	MSI	3	96	94	97	95
	Tommasini i in. 2019	MSI	5	82	75	90	82
	Gibril i in. 2017	MSI	9	82	80	65	72
	Kaplan i in. 2023	RGB + kanał termalny	2	-	92	85	88
	Opracowanie własne	MSI	2	86	69	90	78
SVM	Szabo i in. 2014	HSI	7	80	87	98	92
	Hamedianfar i Shafri, 2014b	MSI	13	76	84	54	66
	Gibril i in. 2017	MSI	9	82	100	58	73
	Norman i in. 2020	MSI + LiDAR	6 (2 x azbest) ¹⁸	70	80	80	80
	Yu i in. 2022	HSI + MSI	2	-	86	90	88
	Kaplan i in. 2023	RGB + kanał termalny	2	-	95	83	88
	Opracowanie własne (SVC)	MSI	2	63	0	0	0

W przypadku klasyfikatora SVM dwukrotnie korzystano z obrazów hiperspektralnych. W badaniu Szabo i in. (2014) wartość F1 na poziomie 92% dla dachów azbestowych uzyskano przy użyciu danych dodatkowych, takich jak NDVI, DTM i DSM oraz wstępnych przekształceń obrazu źródłowego (transformacja MNF, ang. *Minimum Noise Fraction*). W przypadku wykorzystania 126 oryginalnych kanałów spektralnych obrazu źródłowego, wartość F1 wynosiła 65%. W pracy tej porównywano kilka klasyfikatorów, jednak SVM okazał się najbardziej skutecznym. Natomiast klasyfikacja w pracy Yu i in. (2022) była wykonywana na kilku etapach. Warto zwrócić uwagę na duży nakład pracy poprzedzający właściwą klasyfikację. Z uwagi na wykorzystanie obrazów hiperspektralnych dokonano analiz laboratoryjnych próbek dachowych wraz z klasyfikacją SAM. Klasyfikator SVM został wykorzystany dwukrotnie jako wstępne prace przed właściwą klasyfikacją sieciami konwolucyjnymi. Pierwsza klasyfikacja SVM miała za

¹⁸ Wyróżniano dwie klasy dachów azbestowych: stare i nowe pokrycia dachowe.

zadanie wykryć czarne oraz szare materiały na lotniczych obrazach hiperspektralnych. Natomiast druga klasyfikacja SVM miała wykryć na wskazanych obszarach z pierwszego kroku dachy azbestowe na satelitarnym obrazie wielospektralnym. Wartość miary F1 dla azbestu po drugiej analizie była wysoka i wynosiła 88%. Wartości miar oceny dokładności w publikacji Hamedianfar i Shafri, (2014) dotyczą jednej klasy, która zawierała dachy azbestowe oraz dachy z ciemnego betonu. Stanowi to zasadniczą wadę tego podejścia, gdyż celem jest wydzielenie azbestu, a te dwa pokrycia dachowe mają podobne krzywe spektralne (Krówczyńska i in., 2015). Czułość klasyfikacji dla dachów azbestowych metodą SVM w badaniach Gibril i in. (2017) wynosiła aż 100%, jednak po uwzględnieniu precyzji, wartość miary F1 spadła do 73%. W badaniach prezentowanych w publikacji Norman i in. (2020) dokonywano obiektowej klasyfikacji całego obrazu. Wyróżniano dwie klasy dachów azbestowych: stare i nowe pokrycia dachowe. Proces klasyfikacji wspierano danymi z LiDAR-u oraz korzystano m. in. z cech teksturalnych. Podana wartości miary F1 na poziomie 80% jest średnią arytmetyczną obu klas. Kaplan i in. 2023 uzyskali wysoką wartość F1 nie tylko dla klasyfikatora RFC, ale również dla klasyfikatora SVM (88%). W tym przypadku do klasyfikacji wybrano tylko 6 wskaźników, wskazanych jako najodpowiedniejsze do klasyfikacji dachów azbestowych, podczas analiz algorytmem Boruta. Wyznaczone miary oceny dokładności podczas własnych eksperymentów dla SVM (SVC) są bardzo niskie. Wartość F1 wynosi zero, gdyż podczas testowania klasyfikatorów żaden rzeczywisty dach azbestowy nie został sklasyfikowany poprawnie. W porównaniu do badań opisanych w literaturze, gdzie miary dla klasyfikatora SVM była na dość wysokim poziomie, uzyskane własne wyniki oceny dla klasyfikacji wielospektralnego obrazu satelitarnego tym klasyfikatorem, nie potwierdzają jego skuteczności (Tab. 36).

Warto zwrócić uwagę, że klasyfikacja wg własnych badań opierała się wyłącznie na jasnościach pikseli, bez dodatkowych danych wspomagających oraz bez dodatkowych wskaźników. Co potwierdza skuteczność zaproponowanego sposobu filtrowania danych spektralnych dla klasyfikatora RFC. Ważną cechą zaproponowanego filtru jest również jego elastyczność. Istnieje możliwość dopasowania macierzy filtrującej do wybranej miary oceny dokładności klasyfikatora, w zależności od decyzji użytkownika.

9 PODSUMOWANIE I WNIOSKI

W prezentowanej pracy przedstawiono wyniki badań przeprowadzonych w celu identyfikacji dachów azbestowo – cementowych z wykorzystaniem obrazów satelitarnych. W obliczeniach wykorzystano informacje na temat rzeczywistych pokryć dachowych, uzyskane w wyniku wywiadu terenowego. Pojedyncze próbki pokryć dachowych przyporządkowywano do jednej z dwóch klas: azbest lub inne pokrycia dachowe. Warto zauważyć, że do badań wykorzystywano dwa zbiory danych przedstawiające te same pokrycia dachowe. Zbiór danych o rozmiarze 7x7 pikseli przedstawiał piksele zawierające wyłącznie fragmenty pokryć dachowych, natomiast zbiór danych o rozmiarze 14x14 pikseli zawierał piksele pokryć dachowych wraz z otoczeniem wokół budynków.

Otrzymane wartości metryki F1 dla azbestu na danych walidacyjnych w przypadku klasyfikacji z wykorzystaniem opracowanego w pracy filtru są wyższe niż w przypadku klasyfikacji bez jego zastosowania (klasyfikacja w programie SAGA). W związku z tym można potwierdzić przyjętą tezę, że zastosowanie odpowiedniego filtrowania danych spektralnych poprawia możliwość wykrywania dachów azbestowo-cementowych.

Wartości miar oceny klasyfikatorów otrzymane dla obu zbiorów danych (7x7 pikseli oraz 14x14 pikseli) spadają dla większych fragmentów obrazu satelitarnego (14x14 pikseli). Jednak na podstawie przeprowadzonych badań można sformułować wniosek, że w przypadku wykorzystania algorytmu genetycznego spadek wartości ACC oraz F1 nie jest wysoki, a otrzymane wyniki nadal są na zadowalającym poziomie. Generalnie można stwierdzić, że zarejestrowane na zdjęciu bliskie otoczenie budynku wpływa na wyniki klasyfikacji, jednak filtracja danych źródłowych pozwala go ograniczyć.

W prezentowanej pracy zbadano działanie losowej oraz stratyfikowanej metody walidacji krzyżowej na niezrównoważonych zbiorach danych wejściowych. Na podstawie uzyskanych wyników zauważono, że stosowanie kroswalidacji stratyfikowanej pozwalało uzyskiwać nieco wyższe metryki dokładności.

Podczas eksperymentów obliczeniowych pokazano efektywne działanie algorytmu genetycznego podczas doboru macierzy filtrującej dane spektralne. Wyniki miar oceny dokładności wzrastały względem filtrów wyznaczonych losowo.

Wyniki analizy dokładności klasyfikacji na przefiltrowanych danych testowych pozwalają wskazać dwa klasyfikatory, które wydają się być odpowiednia do wykrywania dachów azbestowo-cementowych. Są nimi klasyfikatory RFC oraz GBC należące do klasyfikatorów opartych na losowym lesie drzew decyzyjnych. W dalszych badaniach można rozważyć modyfikacje algorytmu filtrującego, np. poprzez uwzględnienie wagowania wybranych kanałów spektralnych, w celu podniesienia skuteczności klasyfikacji.

W związku z otrzymanymi wartościami miar oceny dokładności klasyfikacji można potwierdzić tezę, że podstawowe klasyfikatory uczenia maszynowego są skutecznym narzędziem w procesie automatycznej identyfikacji dachów azbestowo – cementowych na satelitarnych obrazach wielospektralnych.

10 LITERATURA

Abbasi, M.; Mostafa, S.; Vieira, A.S.; Patorniti, N.; Stewart, R.A. Mapping Roofing with Asbestos-Containing Material by Using Remote Sensing Imagery and Machine Learning-Based Image Classification: A State-of-the-Art Review. *Sustainability*. **2022**, 14, 8068.

Abdulkarim, B.; Kamberov, R.; Hay, G. Supporting urban energy efficiency with volunteered roof information and the Google Maps API. *Remote Sensing*. **2014**, 6, 9691–9711.

Abriha, D.; Kovács, Z.; Ninsawat, S.; Bertalan, L.; Balázs, B.; Szabó, S. Identification of roofing materials with Discriminant Function Analysis and Random Forest classifiers on pan-sharpened WorldView-2 imagery—A comparison. *Hungarian Geographical Bulletin*. **2018**, 67(4), 375–392.

Aggarwal, A.; Singh, H.; Kumar, P.; Singh, M. Optimizing Power Consumption for CNC Turned Parts Using Response Surface Methodology and Taguchi's Technique — A Comparative Analysis. *Journal of Materials Processing Technology*. **2008**, 200 (1), 373–384.

Baluja, S.; Caruana, R. Removing the Genetics from the Standard Genetic Algorithm. *In Machine Learning Proceedings*. **1995**, 38-46

Bassani, C.; Cavalli, R.M.; Cavalcante, F.; Cuomo, V.; Palombo, A.; Pascucci, S.; Pignatti, S. Deterioration status of asbestos-cement roofing sheets assessed by analyzing hyperspectral data. *Remote Sensing of Environment*. **2007**, 109, 361–378.

Bolan, S.; Kempton L.; McCarthy T.; Wijesekara, H; Piyathilake, U.; Jasemizad, T.; Padhye, L.P.; Zhang, T.; Rinklebe, J.; Wang, H.; Kirkham, M.B.; Siddique K.H.M.; Bolan, N. Sustainable management of hazardous asbestos-containing materials: Containment, stabilization and inertization. *Science of The Total Environment*. **2023**, 881, 163456.

Brandi, G.; Tavolari, S. Asbestos and Intrahepatic Cholangiocarcinoma. *Cells*. **2020**, 9(2), 421.

Breiman, L. Random Forests. *Machine Learning*, **2001**, 45, 5–32.

Cavalli, R.M.; Pascucci, S; Pignatti, S. Hyperspectral remote sensing data to map hazardous materials in a rural and industrial district: The Podgorica dwellings case studies. *2009 First Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing, Grenoble, France*, 2009, 1-4.

Cilia, C.; Panigada, C.; Rossini, M.; Candiani, G.; Pepe, M.; Colombo, R. Mapping of Asbestos Cement Roofs and Their Weathering Status Using Hyperspectral Aerial Images. *ISPRS International Journal of Geo-Information*. **2015**, 4, 928–941.

Clark, R.N.;, Gallagher, A.J.; Swayze, G.A. Material Absorption Band Depth Mapping of Imaging Spectrometer Data Using a Complete Band Shape Least-Squares Fit with Library Reference Spectra, [w:] *Proceedings of the Second Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) Workshop*. JPL Publication 90-54, **1990** 176-186.

Cortes, C.; Vapnik, V. Support-vector networks. *Machine Learning*. **1995**, 20, 273–297.

Davis S.M., Landgrebe D.A., Phillips T.L., Swain P.H., Hoffer R.M., Lindenlaub J.C., Silva L.F. Remote sensing: the quantitative approach. *McGraw-Hill International Book Company, New York*. **1978**.

Despini, F.; Silvestri, M.; Panigada, C.; Anelli, M.; Costanzini, S.; Musacchio, M.; Teggi, S.; Muscio, A.; Pignatti, S.; Bogliolo, M.P. Identification of potential asbestos roofs with World View 3 images with pixel and object oriented techniques: a case study in Italy. *20th EGU General Assembly, EGU2018, Proceedings from the conference held 4-13 April, 2018 in Vienna, Austria*. **2018**, 16480.

Eiben, A.E.; Smith, J.E. Introduction to Evolutionary Computing. *Natural Computing Series*. Springer, Berlin, Heidelberg. **2015**.

Fiumi, L.; Campopiano, A.; Casciardi, S.; Ramires, D. Method validation for the identification of asbestos-cement roofing. *Applied Geomatics*. **2012**, 4(1), 55–64.

Fiumi, L.; Congedo, L.; Meoni, C. Developing expeditious methodology for mapping asbestos-cement roof coverings over the territory of Lazio Region. *Applied Geomatics*. **2014**, 6, 37–48.

Foody, G. M. Status of Land Cover Classification Accuracy Assessment. *Remote Sensing of Environment*. **2002**, 80 (1), 185–201.

Frassy, F.; Candiani, G.; Maianti, P.; Marchesi, A.; Nodari, F.R.; Rusmini, M.; Albonico, C.; Gianinetto, M. Airborne remote sensing for mapping asbestos roofs in aosta valley. *2012 IEEE International Geoscience and Remote Sensing Symposium, Munich, Germany*. **2012**, 7541-7544.

Frassy, F.; Candiani, G.; Rusmini, M.; Maianti, P.; Marchesi, A.; Nodari, F.R.; Via, G.D.; Albonico, C.; Gianinetto, M. Mapping asbestos-cement roofing with hyperspectral remote sensing over a large mountain region of the Italian western alps. *Sensors*. **2014**, 14, 15900–15913.

Friedman, J.H. Greedy Function Approximation: A Gradient Boosting Machine. *The Annals of Statistics*. **2001**, 29, 5, 1189–1232.

Gašparović, M.; Jogun, T. The effect of fusing Sentinel-2 bands on land-cover classification. *International Journal of Remote Sensing*. **2018**, 39, 822–841.

Géron, A. Uczenie maszynowe z użyciem Scikit-Learn i TensorFlow. Wydanie II. *Helion*. **2020**.

Geurts, P., Ernst, D., Wehenkel, L. Extremely randomized trees. *Machine Learning*. **2006**, 63, 3-42.

Ghanea, M.; Moallem, P.; Momeni, M. Building extraction from high-resolution satellite images in urban areas: Recent methods and strategies against significant challenges. *International Journal of Remote Sensing*. **2016**, 37, 5234–5248.

Gibril, M.B.A.; Shafri, H.Z.M.; Hamedianfar, A. New semi-automated mapping of asbestos cement roofs using rule-based object-based image analysis and Taguchi optimization technique from WorldView-2 images. *International Journal of Remote Sensing*. **2017**, 38, 467–491.

Goldberg, D.E. Algorytmy genetyczne i ich zastosowanie, *Wydawnictwo Naukowo-Techniczne, Warszawa*. **1998**.

Grygierek, K. Samoadaptacyjna metoda algorytmów genetycznych w optymalizacji przestrzennych kratownic. *Modelowanie Inżynierskie*. **2014**, 21(52), 80-86.

Guo, W.; Yang, W.; Zhang, H.; Hua, G. Geospatial Object Detection in High-resolution Satellite Images Based on Multi-Scale Convolutional Neural Network. *Remote Sensing*. **2018**, 10, 131.

Hamedianfar, A.; Shafri, H.Z.M.; Mansor, S.; Ahmad, N. Improving detailed rule-based feature extraction of urban areas from WorldView-2 image and lidar data. *International Journal of Remote Sensing*. **2014**, 35, 1876–1899.

- Hamedianfar, A.; Shafri, H.Z.M. Detailed intra-urban mapping through transferable OBIA rule sets using WorldView-2 very-high-resolution satellite images. *International Journal of Remote Sensing*. **2015**, 36, 3380–3396.
- Haralick, R.M. Statistical and structural approaches to texture. *Proceedings of the IEEE*. **1979**, 67, 786–804.
- Harrison, M. Uczenie maszynowe w Pythonie. Leksykon kieszonkowy. *Helion*. **2020**.
- Hikuwai, M.V.; Patorniti, N.; Vieira, A.S.; Frangioudakis Khatib, G.; Stewart, R.A. Artificial Intelligence for the Detection of Asbestos Cement Roofing: An Investigation of Multi-Spectral Satellite Imagery and High-Resolution Aerial Imagery. *Sustainability* **2023**, 15, 4276.
- Ho, T.K. Random decision forests. *Proceedings of 3rd International Conference on Document Analysis and Recognition, Montreal, QC, Canada*. **1995**, 1, 278–282.
- Hodgson, M.E.; Jensen, J.R.; Tullis, J.A.; Riordan, K.D.; Archer, C.M. Synergistic use of lidar and color aerial photography for mapping urban parcel imperviousness. *Photogrammetric Engineering & Remote Sensing*. **2003**, 69, 973–980.
- Holland, J. H. Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence. U Michigan Press. **1975**.
- Huang, J.; Zhang, X.; Xin, Q.; Sun, Y.; Zhang, P. Automatic building extraction from high-resolution aerial images and LiDAR data using gated residual refinement network. *ISPRS Journal of Photogrammetry and Remote Sensing*. **2019**, 151, 91–105.
- Huang, X.; Li, S.; Li, J.; Jia, X.; Li, J.; Zhu, X.X.; Benediktsson, J.A. A multispectral and multiangle 3-D convolutional neural network for the classification of ZY-3 satellite images over urban areas. *IEEE Transactions on Geoscience and Remote Sensing*. **2020**, 59, 0266–10285.
- IARC - International Agency for Research on Cancer. Asbestos (chrysotile, amosite, crocidolite, tremolite, actinolite, and anthophyllite). IARC Monogr Eval Carcinog Risks Hum. **2012**, 100C, 219–309.
- Ilehag, R.; Schenk, A.; Huang, Y.; Hinz, S. KLUM: An urban VNIR and SWIR spectral library consisting of building materials. *Remote Sensing*. **2019**, 11, 2149.
- Kamp, D. W. Subspecialty in Translational Medicine: Asbestos-induced lung diseases: an update. *Translational Research*. **2009**, 153(4), 143–152.
- Kaplan, G.; Gašparović, M.; Kaplan, O.; Adjiski, V.; Comert, R.; Mobariz, M.A. Machine Learning-Based Classification of Asbestos-Containing Roofs Using Airborne RGB and Thermal Imagery. *Sustainability*. **2023**, 15, 6067.
- Kotthaus, S.; Smith, T.E.L.; Wooster, M.J.; Grimmond, C.S.B. Derivation of an urban materials spectral library through emittance and reflectance spectroscopy. *ISPRS Journal of Photogrammetry and Remote Sensing*. **2014**, 94, 194–212.
- Krówczyńska, M.; Raczko, E.; Staniszevska, N.; Wilk, E. Asbestos-cement roofing identification using remote sensing and convolutional neural networks (CNNs). *Remote Sensing*. **2020**, 12, 408.

Krówczyńska M.; Wilk E. Aerial imagery and geographic information systems used in the asbestos removal process in Poland. [w:] 33th EARSeL Symposium Towards Horizon 2020: Earth Observation and Social Perspectives, edited by Lasaponara R., Masini N., Biscione M., Matera, Italy. **2013**, 823-828.

Krówczyńska, M.; Wilk, E.; Pabjanek P. Zróżnicowanie spektralne dachowych pokryć azbestowo-cementowych. *Teledetekcja Środowiska*. **2015**, 53, 75–86.

Krówczyńska, M.; Wilk, E.; Pabjanek P.; Kycko M. Hyperspectral Discrimination of Asbestos-Cement Roofing. *Geomatics and Environmental Engineering*. **2017**, 11, 47–65.

Kruse, F. A.; Lefkoff, A.B. ; Boardman, J.B; Heidebrecht, K.B.; Shapiro, A.T.; Barloon, P.J.; Goetz, A.F.H. The Spectral Image Processing System (SIPS) - Interactive Visualization and Analysis of Imaging spectrometer Data. *Remote Sensing of Environment*. **1993**, 44, 145–163.

Książek. J. Study of Selected Textural Features Properties on Asbestos Roof Images. *Geomatics and Environmental Engineering*. **2018**, 12(4), 45-54.

Kursa, M. B.; Rudnicki. W. R. Feature Selection with the Boruta Package. *Journal of Statistical Software*, **2010**, 36, 1-13.

Lillesand, T.; Kiefer, R.W.; Chipman, J. Remote Sensing and Image Interpretation, 7th Edition, Wiley, **2015**.

Luo, L.; Li, P.; Yan, X. Deep learning-based building extraction from remote sensing images: A comprehensive review. *Energies*. **2021**, 14, 7982.

Marino, C.M.; Panigada, C.; Busetto, L. Airborne hyperspectral remote sensing applications in urban areas: Asbestos concrete sheeting identification and mapping. *In Proceedings of the IEEE/ISPRS Joint Workshop on Remote Sensing and Data Fusion over Urban Areas (Cat. No. 01EX482), Rome, Italy, 8–9 November 2001*. 212–216.

Michalewicz Z., Algorytmy Genetyczne + struktury danych = programy ewolucyjne, Wydawnictwo Naukowo Techniczne, Warszawa **2003**.

Militello, G.M.; Gaggero, L.; La Maestra, S. Asbestiform amphiboles and cleavage fragments analogues: overview of critical dimensions, aspect ratios, exposure and health effects. *Minerals*. **2021**, 11(5), 525.

Myint, S.W.; Gober, P.; Brazel, A.; Grossman-clarke, S.; Weng, Q. Per-pixel vs. object-based classification of urban land cover extraction using high spatial resolution imagery. *Remote Sensing of Environment*. **2011**, 115, 1145–1161.

Nielsen M.A. Neural Networks and Deep Learning. *Determination Press*, **2015**.

Norman, M.; Shafri, H.Z.M.; Mansor, S.; Yusuf, B.; Radzali, N.A.W.M. Fusion of multispectral imagery and LiDAR data for roofing materials and roofing surface conditions assessment. *International Journal of Remote Sensing*. **2020**, 41, 7090–7111.

Osińska-Skotak, K. Zastosowanie technik teledetekcyjnych do inwentaryzacji cementowo-azbestowych pokryć dachowych. *Teledetekcja Środowiska*. **2014**, 51(2), 73-83.

Osińska-Skotak, K.; Ostrowski, W. Use of satellite and ALS data for classification of roofing materials on the example of asbestos roof tile identification. *Technical Sciences*. **2015**, 18(4), 283–298.

- Pál, K.F. Genetic algorithms for the traveling salesman problem based on a heuristic crossover operation. *Biological Cybernetics*. 69, 539–546.
- Park, Y.; Guldmann, J.-M. Measuring continuous landscape patterns with Gray-Level Co-Occurrence Matrix (GLCM) indices: An alternative to patch metrics? *Ecological Indicators*. **2020**, 109, 105802.
- Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; Vanderplas, J.; Passos, A.; Cournapeau, D.; Brucher, M.; Perrot, M.; Duchesnay, E. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*. **2011**, 12, 2825-2830.
- Pinho, C.M.D.; Fonseca, L.M.G.; Korting, T.S.; de Almeida, C.M.; Kux, H.J.H. Land-cover classification of an intra-urban environment using high-resolution images and object-based image analysis. *International Journal of Remote Sensing*. **2012**, 33, 5973–5995.
- Pinho, C.M.D.; Silva, F.C.; Fonseca, L.M.C.; Monteiro, A.M.V. Intra-Urban Land Cover Classification from High-Resolution Images Using the C4.5 Algorithm. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. **2008**, 37(B7), 695–700.
- POKA: Programme for Asbestos Abatement in Poland 2009–2032, Annex to the Resolution No. 39/2010 of the Council of Ministers of 15 March **2010**.
- Potvin, J.Y. Genetic algorithms for the traveling salesman problem. *Annals of Operations Research*. **1996**, 63, 337–370.
- Raczko, E.; Krówczyńska, M.; Wilk, E. Asbestos roofing recognition by use of convolutional neural networks and high-resolution aerial imagery. Testing different scenarios. *Building and Environment*. **2022**, 217, 109092.
- Smadi, A.A.; Yang, S.; Abugabah, A.; Alzubi, A.A.; Sanzogni, L. A Pansharpening Based on the Non-Subsampled Contourlet Transform and Convolutional Autoencoder: Application to QuickBird Imagery. *IEEE Access*. **2022**, 10, 44778-44788.
- Small, C. High spatial resolution spectral mixture analysis of urban reflectance. *Remote Sensing of Environment*. **2003**, 88, 170–186.
- Smoliński, A. Samoadaptacyjna optymalizacja genetyczna. *Zeszyty Naukowe Wydziału Elektroniki i Informatyki Politechniki Koszalińskiej*. **2016**, 10, 39-50.
- Szabó, S.; Burai, P.; Kovács, Z.; Szabó, G.; Kerényi, A.; Fazekas, I.; Paládi, M.; Buday, T.; Szabó, G. Testing algorithms for the identification of asbestos roofing based on hyperspectral data. *Environmental Engineering and Management Journal*. **2014**, 143(11), 2875–2880.
- Taherzadeh, E.; Shafri, H.Z.M. Development of a generic model for the detection of roof materials based on an object-based approach using WorldView-2 satellite imagery. *Advances in Remote Sensing*. **2013**, 2, 312–321.
- Taherzadeh, E.; Shafri, H.Z.M.; Shahi, K. Roof material detection based on object-based approach using WorldView-2 satellite imagery. *Journal of Control, Automation and Electrical Systems*. **2014**, 8, 1737–1740.
- Talukdar, S.; Singha, P.; Mahato, S.; Pal, S.; Liou, Y.-A.; Rahman, A. Land-use land-cover classification by machine learning classifiers for satellite observations—A review. *Remote Sensing*. **2020**, 12, 1135.

- Tomljenovic, I.; Höfle, B.; Tiede, D.; Blaschke, T. Building extraction from airborne laser scanning data: An analysis of the state of the art. *Remote Sensing*. **2015**, *7*, 3826–3862.
- Tommasini, M.; Bacciottini, A.; Gherardelli, M. A QGIS Tool for Automatically Identifying Asbestos Roofing. *ISPRS International Journal of Geo-Information*. **2019**, *8*, 131.
- Turing, A.M. Computing machinery and intelligence. *Mind*. **1950** LIX (238), 433–460.
- Viana, C.M.; Oliveira, S.; Oliveira, S.C.; Rocha, J. Land use/land cover change detection and urban sprawl analysis. *Spatial Modeling in GIS and R for Earth and Environmental Sciences*. **2019**, 621–651.
- Virta, R.L. Asbestos: Geology, Mineralogy, Mining, and Uses. U.S. Department of the Interior, U.S. Geological Survey. *Open-File Report* **2002**-149.
- Vu, T.T.; Yamazaki, F.; Matsuoka, M. Multi-scale solution for building extraction from LiDAR and image data. *International Journal of Applied Earth Observation and Geoinformation*. **2009**, *11*, 281–289.
- Wang, P.; Zhang, L.; Zhang, G.; Bi, H.; Dalla Mura, M.; Chanussot, J. Superresolution land cover mapping based on pixel-, subpixel-, and superpixel-scale spatial dependence with pansharpening technique. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*. **2019**, *12*, 4082–4098.
- Welch, R. Spatial resolution requirements for urban studies. *International Journal of Remote Sensing*. **1982**, *3*(2), 139–146.
- WHO. Chrysolite Asbestos. **2014**, ISBN: 978 92 4 156481 6.
- Winiczenko, R. Algorytmy genetyczne I ich zastosowania. *Postępy Techniki Przetwórstwa Spożywczego*. **2008**, *1*, 107–110.
- Yu, T.-T.; Lin, Y.-C.; Lan, S.-C.; Yang, Y.-E.; Wu, P.-Y.; Lin, J.-C. Mapping Asbestos-Cement Corrugated Roofing Tiles with Imagery Cube via Machine Learning in Taiwan. *Remote Sensing*. **2022**, *14*, 3418.
- Zephaniah, H.S.; Aremu, S.O. Asbestos: A Silent Potent Killer. *European Journal of Environment and Public Health*. **2019**. *3*(2), em0030.

SPIS TABEL

Tab. 1 Rodzaj obrazu źródłowego wykorzystanego w identyfikacji dachów azbestowych na przestrzeni lat (opracowanie własne)	21
Tab. 2 Zestawienie badań na temat wykrywania dachów azbestowych z wykorzystaniem obrazów hiperspektralnych (Abbasi i in., 2022, zmodyfikowano)	22
Tab. 3 Zestawienie badań na temat wykrywania dachów azbestowych z wykorzystaniem obrazów wielospektralnych (Abbasi i in., 2022, zmodyfikowano).....	24
Tab. 4 Zestawienie metod klasyfikacji, stosowanych klasyfikatorów oraz oceny dokładności, w badaniach nad wykrywaniem azbestowo – cementowych pokryć dachowych (Abbasi i in., 2022, zmodyfikowano)	27
Tab. 5 Macierz błędów dla klasyfikatora binarnego (Géron, 2020, zmodyfikowano)	41
Tab. 6 Parametry kanałów spektralnych satelity WorldView-3 (www.digitalglobe.com, dostęp: maj 2023)	50
Tab. 7 Średnie długości fali reprezentujące poszczególne kanały spektralne (opracowanie własne)	57
Tab. 8 Macierze X_i oraz macierz y dla przykładowych danych (opracowanie własne)	58
Tab. 9 Zestawienie średniej długości fali elektromagnetycznej dla poszczególnych kanałów spektralnych (opracowanie własne)	60
Tab. 10 Macierze X_i przefiltrowane macierzą F (opracowanie własne).....	60
Tab. 11 Macierz NO : zliczone niezerowe wartości w zestawie macierzy X_i (opracowanie własne)	61
Tab. 12 Macierz X dla przykładowych danych (opracowanie własne)	61
Tab. 13 Zestawienie podstawowych wartości statystycznych dla zestawu danych 7x7 pikseli (opracowanie własne)	62
Tab. 14 Zestawienie podstawowych wartości statystycznych dla zestawu danych 14x14 pikseli (opracowanie własne)	62
Tab. 15 Wymiar macierzy wykorzystywanych w procesie klasyfikacji (opracowanie własne)	68
Tab. 16 Wyniki klasyfikacji z wykorzystaniem algorytmów poprawiających filtr danych (opracowanie własne)	76
Tab. 17 Numer próby losowej i numer generacji algorytmu genetycznego, dla których wyznaczono najwyższą wartość parametru oceny dokładności klasyfikacji (opracowanie własne)	78
Tab. 18 Wartość środkowa i odchylenie (parametry filtra) dla poszczególnych kanałów spektralnych wyznaczone przez algorytm genetyczny wykorzystujący jako funkcje oceny dokładność ACC (opracowanie własne)	80
Tab. 19 Wartość środkowa i odchylenie (parametry filtra) dla poszczególnych kanałów spektralnych wyznaczone przez algorytm genetyczny wykorzystujący jako funkcje oceny parametr $F1$ (opracowanie własne)	80
Tab. 20 Zestawienie wyników oceny klasyfikatora przy użyciu krosvalidacji, dla próbek 7x7 pikseli (opracowanie własne)	83
Tab. 21 Zestawienie wyników oceny klasyfikatora przy użyciu krosvalidacji, dla próbek 14x14 pikseli (opracowanie własne)	83
Tab. 22 Średnie wartości miar oceny dokładności podczas klasyfikacji z wykorzystaniem algorytmów poprawiających filtr danych przy zastosowaniu krosvalidacji (opracowanie własne)	86
Tab. 23 Numer próby losowej i numer generacji algorytmu genetycznego, dla których wyznaczono najwyższą średnią wartość parametru oceny dokładności klasyfikacji (opracowanie własne)	87

Tab. 24 Wartość środkowa i odchylenie (parametry filtra) dla poszczególnych kanałów spektralnych wyznaczone przez algorytm genetyczny wykorzystujący jako funkcje oceny dokładność ACC (opracowanie własne)	89
Tab. 25 Wartość środkowa i odchylenie (parametry filtra) dla poszczególnych kanałów spektralnych wyznaczone przez algorytm genetyczny wykorzystujący jako funkcje oceny parametr F1 (opracowanie własne)	89
Tab. 26 Macierze błędów dla klasyfikacji wykonanej w oprogramowaniu SAGA (opracowanie własne)	94
Tab. 27 Wyniki oceny klasyfikacji w oprogramowaniu SAGA (opracowanie własne)	94
Tab. 28 Ocena dokładności klasyfikacji (7x7 pikseli) na danych testowych z wykorzystaniem uzyskanych filtrów (opracowanie własne)	95
Tab. 29 Zestawienie wybranych macierzy błędów oraz miar oceny dokładności dla próbek 7x7 pikseli (opracowanie własne)	96
Tab. 30 Ocena dokładności klasyfikacji (14x14 pikseli) na danych testowych z wykorzystaniem uzyskanych filtrów (opracowanie własne)	97
Tab. 31 Zestawienie wybranych macierzy błędów oraz miar oceny dokładności dla próbek 14x14 pikseli (opracowanie własne)	98
Tab. 32 Zestawienie oceny dokładności klasyfikacji wszystkich strategii dla próbek 7x7 pikseli (opracowanie własne)	99
Tab. 33 Zestawienie oceny dokładności klasyfikacji wszystkich strategii dla próbek 14x14 pikseli (opracowanie własne)	99
Tab. 34 Wybrane miary oceny dokładności klasyfikacji z wykorzystaniem filtrowania danych dla próbek 7x7 pikseli (opracowanie własne)	103
Tab. 35 Wybrane miary oceny dokładności klasyfikacji z wykorzystaniem filtrowania danych dla próbek 14x14 pikseli (opracowanie własne)	103
Tab. 36 Zestawienie miar dokładności dla klasyfikatorów RFC oraz SVM (opracowanie własne)	105

SPIS RYSUNKÓW

Rys. 1 Typy azbestu wyróżnione ze względu na budowę włókien (Brandi i Tavalari, 2020, zmodyfikowano)	13
Rys. 2 Ilość wyrobów zawierających azbest: zinwentaryzowanych oraz usuniętych z podziałem na województwa w Polsce. (Mg = tona).	16
Rys. 3 Ogólna procedura badań na temat wykrywania dachów azbestowych (Abbasi i in., 2022, zmodyfikowano)	18
Rys. 4 Lokalizacja obszaru badań: (A) w granicach Rzeczypospolitej Polski; (B) w granicach województwa małopolskiego (opracowanie własne)	46
Rys. 5 Obszar wsi Krzczonów (https://www.polskawliczbach.pl/wies_Krzczonek_malopolskie , dostęp: maj 2023)	46
Rys. 6 Azbestowe pokrycia dachowe we wsi Krzczonów (<i>Google Street View</i>)	48
Rys. 7 Inne pokrycia dachowe we wsi Krzczonów: (A) dachówka cementowa, (B) blachodachówka, (C) dachówka cementowa	49
Rys. 8 Obraz satelitarny WorldView-3	51
Rys. 9 Schemat wstępnego opracowanie danych (opracowanie własne)	53
Rys. 10 Wartsa punktowa przedstawiająca rozmieszczenie dachów (opracowanie własne)	53
Rys. 11 Wektory przygotowane do wycięcia fragmentów dachów na tle wyostzonego obrazu satelitarnego WorldView-3 a – azbest; n – inne pokrycie dachowe (opracowanie własne)	53
Rys. 12 Fragment kodu Python rozdzielającego obrazu wielospektralne na pojedyncze kanały (opracowanie własne)	54
Rys. 13 Schemat wstępnego przygotowania danych do procesu klasyfikacji (opracowanie własne)	55
Rys. 14 Fragment kodu Python przekształcającego format danych tif na macierz npy oraz spłaszczającego macierzy npy (opracowanie własne)	56
Rys. 15 Fragment kodu Python tworzącego macierze X_i oraz y (opracowanie własne)	56
Rys. 16 Schemat danych wejściowych do przykładu obliczeniowego algorytmu filtrowania (opracowanie własne)	58
Rys. 17 Macierz F wykorzystywana w przykładzie obliczeniowym (opracowanie własne)	59
Rys. 18 Zakresy przedziałów filtrowania danych przedstawione na osi wzrostu wartości jasności pikseli (opracowanie własne)	59
Rys. 19 Rozpiętość jasności pikseli dla próbek dachów azbestowych oraz innych pokryć dachowych w poszczególnych kanałach spektralnych (opracowanie własne)	64
Rys. 20 Rozkład jasności pikseli dla wszystkich kanałów spektralnych z rozróżnieniem zestawów próbek (opracowanie własne)	64
Rys. 21 Rozkłady jasności pikseli w poszczególnych kanałach spektralnych dla próbek 7x7 pikseli (opracowanie własne)	66
Rys. 22 Rozkłady jasności pikseli w poszczególnych kanałach spektralnych dla próbek 14x14 pikseli (opracowanie własne)	67
Rys. 23 Schemat prowadzonych eksperymentów obliczeniowych (opracowanie własne)	70
Rys. 24 Schemat blokowy eksperymentów obliczeniowych, pozwalający na wyznaczenie parametrów filtrowania danych (opracowanie własne)	72
Rys. 25 Wyniki klasyfikacji dla próbek 7x7 pikseli z podziałem na klasyfikatory (A) metryka: dokładność ACC; (B) metryka: parametr F1 (opracowanie własne)	77
Rys. 26 Wyniki klasyfikacji dla próbek 14x14 pikseli z podziałem na klasyfikatory (A) metryka: dokładność ACC; (B) metryka: parametr F1 (opracowanie własne)	77

Rys. 27 Numer próby losowej dla której klasyfikator uzyskał najwyższy wynik metryk dokładności (A) próbki 7x7 pikseli; (B) próbki 14x14 pikseli (opracowanie własne).....	79
Rys. 28 Numer generacji dla której klasyfikator uzyskał najwyższy wynik metryk dokładności (A) próbki 7x7 pikseli; (B) próbki 14x14 pikseli (opracowanie własne).....	79
Rys. 29 Zakresy przedziałów filtrowania dla poszczególnych kanałów spektralnych wyznaczone przez algorytm genetyczny wykorzystujący jako funkcje oceny dokładność ACC (opracowanie własne).....	81
Rys. 30 Zakresy przedziałów filtrowania dla poszczególnych kanałów spektralnych wyznaczone przez algorytm genetyczny wykorzystujący jako funkcje oceny parametr F1 (opracowanie własne).....	82
Rys. 31 Wyniki średniej oceny klasyfikatora, dla próbek 7x7 pikseli, otrzymane podczas: wyznaczania filtrów (Eks_obl), stratyfikowanej walidacji krzyżowej (3CV_strat), walidacji krzyżowej (3CV) (opracowanie własne).....	84
Rys. 32 Wyniki średniej oceny klasyfikatora, dla próbek 14x14 pikseli, otrzymane podczas: wyznaczania filtrów (Eks_obl), stratyfikowanej walidacji krzyżowej (3CV_strat), walidacji krzyżowej (3CV) (opracowanie własne).....	85
Rys. 33 Wyniki klasyfikacji dla próbek 7x7 pikseli z podziałem na klasyfikatory przy zastosowaniu kroswalidacji (A) metryka: dokładność ACC; (B) metryka: parametr F1 (opracowanie własne)....	86
Rys. 34 Wyniki klasyfikacji dla próbek 14x14 pikseli z podziałem na klasyfikatory przy zastosowaniu kroswalidacji (A) metryka: dokładność ACC; (B) metryka: parametr F1 (opracowanie własne).....	87
Rys. 35 Numer próby losowej dla której klasyfikator uzyskał najwyższy średni wynik metryk dokładności (A) próbki 7x7 pikseli; (B) próbki 14x14 pikseli (opracowanie własne).....	88
Rys. 36 Numer generacji dla której klasyfikator uzyskał najwyższy średni wynik metryk dokładności (A) próbki 7x7 pikseli; (B) próbki 14x14 pikseli (opracowanie własne).....	88
Rys. 37 Zakresy przedziałów filtrowania dla poszczególnych kanałów spektralnych wyznaczone przez algorytm genetyczny wykorzystujący jako funkcje oceny dokładność ACC (opracowanie własne).....	90
Rys. 38 Zakresy przedziałów filtrowania dla poszczególnych kanałów spektralnych wyznaczone przez algorytm genetyczny wykorzystujący jako funkcje oceny parametr F1 (opracowanie własne).....	91
Rys. 39 Fragment wyników klasyfikacji RFC z zaznaczonymi próbkami dachowymi 7x7 pikseli sklasyfikowane piksele: kolor czarny – azbest, kolor biały – inne pokrycia dachowe; rzeczywista próbka dachu: ramka czerwona – azbest, ramka niebieska – inne pokrycie dachowe (opracowanie własne).....	92
Rys. 40 Fragment wyników klasyfikacji SVC z zaznaczonymi próbkami dachowymi 7x7 pikseli sklasyfikowane piksele: kolor czarny – azbest, kolor biały – inne pokrycia dachowe; rzeczywista próbka dachu: ramka czerwona – azbest, ramka niebieska – inne pokrycie dachowe (opracowanie własne).....	93
Rys. 41 Fragment wyników klasyfikacji RFC z zaznaczonymi próbkami dachowymi 14x14 pikseli sklasyfikowane piksele: kolor czarny – azbest, kolor biały – inne pokrycia dachowe; rzeczywista próbka dachowa: ramka czerwona – azbest, ramka niebieska – inne pokrycie dachowe (opracowanie własne).....	93
Rys. 42 Fragment wyników klasyfikacji SVC z zaznaczonymi próbkami dachowymi 14x14 pikseli.....	94
Rys. 43 Wartości miar oceny dokładności dla próbek 7x7 pikseli z wykorzystaniem filtrów wg strategii I (A) funkcja oceny AG: ACC; (B) funkcja oceny AG: F1 (opracowanie własne).....	96
Rys. 44 Wartości miar oceny dokładności dla próbek 7x7 pikseli z wykorzystaniem filtrów wg strategii II (A) funkcja oceny AG: ACC; (B) funkcja oceny AG: F1 (opracowanie własne).....	96

Rys. 45 Wartości miar oceny dokładności dla próbek 7x7 pikseli z wykorzystaniem filtrów wg strategii I (A) funkcja oceny AG: ACC; (B) funkcja oceny AG: F1 (opracowanie własne).....	98
Rys. 46 Wartości miar oceny dokładności dla próbek 7x7 pikseli z wykorzystaniem filtrów wg strategii II (A) funkcja oceny AG: ACC; (B) funkcja oceny AG: F1 (opracowanie własne).....	98
Rys. 47 Wartości miar oceny dokładności klasyfikatorów dla próbek 7x7 pikseli (opracowanie własne)	100
Rys. 48 Wartości miar oceny dokładności klasyfikatorów dla próbek 14x14 pikseli (opracowanie własne)	101

ZAŁĄCZNIK 1

Kod *Python* wykorzystywany podczas eksperymentów obliczeniowych w celu wyznaczenia filtrów - strategia I

```
import os
import csv
import numpy as np
import pandas as pd
import math
import random
from datetime import datetime
from sklearn.model_selection import StratifiedShuffleSplit
from sklearn.linear_model import SGDClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report
from sklearn.svm import LinearSVC
from sklearn.svm import SVC
from sklearn.ensemble import RandomForestClassifier
from sklearn.ensemble import ExtraTreesClassifier
from sklearn.ensemble import GradientBoostingClassifier, AdaBoostClassifier, VotingClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.manifold import LocallyLinearEmbedding

import random
from deap import base
from deap import creator
from deap import tools

#klasa zawierająca globalne parametry pracy algorytmów
class SETUP:
    clf_names = ['SGD', 'RFC', 'SVC', 'ETC', 'GBC']
    clf_name = 'RFC'

    dl = [425,480,545,605,660,725,832,950,]
    x_dat = None
    y = None

    #save parametr
    path_to_csv = r'C:\result\7pix\ACC\
    name_of_file = 'sol'

    #tmp
    nb_of_inv = -1

#funkcja odczytująca dane
def readDataFromFiles(path_dat= r'C:\7pix\7pix_X_y'):
    y = np.load(path_dat + r'\y.npy')
    x_dat = [np.load(path_dat + rf'\x{i}.npy') for i in range(1,9)]
    return x_dat, y

#funkcja normalizująca dane
def normalizeData(x_dat_org):
    x_dat = []
    for t in x_dat_org:
        x_dat.append(t/(np.max(t)))
    return x_dat

#funkcja zwracająca wybrany klasyfikator
def getCLF():
    if SETUP.clf_name == 'SGD':
        return SGDClassifier(random_state=1, max_iter=10000)
    elif SETUP.clf_name == 'SVC':
        return SVC()
    elif SETUP.clf_name == 'RFC':
        return RandomForestClassifier(n_estimators=300, max_leaf_nodes=64, n_jobs=-1)
    elif SETUP.clf_name == 'ETC':
        return ExtraTreesClassifier(n_estimators=300, max_leaf_nodes=32, n_jobs=-1)
    elif SETUP.clf_name == 'GBC':
        return GradientBoostingClassifier(max_depth=2, n_estimators=300, learning_rate=1.0)
    return None
```



```

#funkcja oceny rozwizania
def evalOneMax(individual):
    #-----
    dl = SETUP.dl
    x_dat = SETUP.x_dat
    y = SETUP.y
    #-----
    q = individual[0:2]
    d = individual[1:2]
    X = np.zeros(x_dat[0].shape)
    i_x, i_y = X.shape
    for i in range(i_x):
        for j in range(i_y):
            licz = 0.001
            for ind, e in enumerate(q):
                xtmp = x_dat[ind]
                X[i,j] += dl[ind] if xtmp[i,j] <= e+d[ind] and xtmp[i,j] > e-d[ind] else 0
                licz += 1 if xtmp[i,j] <= e+d[ind] and xtmp[i,j] > e-d[ind] else 0
            X[i,j] = X[i,j]/licz
    splitter = StratifiedShuffleSplit(n_splits=42, test_size=0.3, random_state=1)
    for train_id, test_id in splitter.split(X, y):
        X_train, y_train = X[train_id], y[train_id]
        X_test, y_test = X[test_id], y[test_id]
    clf = getCLF()
    clf.fit(X_train, y_train)
    y_pred = clf.predict(X_test)
    report = classification_report(y_test, y_pred, output_dict=True, zero_division=1)
    return report['accuracy'],

def setup_genetic_algorithm():
    creator.create("FitnessMax", base.Fitness, weights=(1.0,))
    creator.create("Individual", list, fitness=creator.FitnessMax)

def genetic_algorithm(nbPop = 10, nbGenerations = 100, CXPB = 0.6, MUTPB = 0.05, startPopulations = None):

    #-----
    hof = tools.HallOfFame(maxsize=1)
    hof.b_Gen = 0

    def update_hof(population, nb_Gen):
        fc = hof[0].fitness.values[0] if len(hof)>0 else 0;
        hof.update(population)
        if fc < hof[0].fitness.values[0]:
            hof.b_Gen = nb_Gen

    #-----
    #-----
    toolbox = base.Toolbox()
    toolbox.register("attr_float", random.uniform, 0, 1)
    toolbox.register("individual", tools.initRepeat, creator.Individual, toolbox.attr_float, 16)
    toolbox.register("population", tools.initRepeat, list, toolbox.individual)
    toolbox.register("evaluate", evalOneMax)
    toolbox.register("mate", tools.cxTwoPoint)
    toolbox.register("mutate", tools.mutUniformInt, low=0, up=1, indpb=0.3)
    toolbox.register("select", tools.selTournament, tournsize=3)
    #-----

    pop = toolbox.population(n=nbPop)
    SETUP.nb_of_inv = -1
    for inv in pop:
        SETUP.nb_of_inv += 1
        inv.fitness.values = start_population[SETUP.clf_name][SETUP.nb_of_inv][0]
        for j in range(len(inv)):
            val = start_population[SETUP.clf_name][SETUP.nb_of_inv][1][j]
            inv[j] = val

    #-----

```

```

fits = [ind.fitness.values[0] for ind in pop]
g = 0
tab_stat = []
g < nbGenerations:
    g = g + 1
    print(f'Generation {g}/{nbGenerations}')
    offspring = toolbox.select(pop, len(pop))
    offspring = list(map(toolbox.clone, offspring))

    for child1, child2 in zip(offspring[::2], offspring[1::2]):
        if random.random() < CXPB:
            toolbox.mate(child1, child2)
            del child1.fitness.values
            del child2.fitness.values

    for mutant in offspring:
        if random.random() < MUTPB:
            toolbox.mutate(mutant)
            del mutant.fitness.values

    invalid_ind = [ind for ind in offspring if not ind.fitness.valid]
    fitnesses = map(toolbox.evaluate, invalid_ind)
    for ind, fit in zip(invalid_ind, fitnesses):
        ind.fitness.values = fit

    pop[:] = offspring

    fits = [ind.fitness.values[0] for ind in pop]

    length = len(pop)
    mean = sum(fits) / length
    sum2 = sum(x*x for x in fits)
    std = abs(sum2 / length - mean**2)**0.5
    print(f'\t Max:{max(fits):.3f}, Min:{min(fits):.3f}, Avg:{mean:.3f}, Std:{std:.3f}')

    tab_stat.append([min(fits), max(fits), mean, std])
    update_hof(pop, g)

return hof[0], hof[0].fitness.values[0], hof.b_Gen

def random_algorithm(nb_iterations = 10):
    fc = type('FC', (object,), {})
    fc.inv = []
    fc.fit = 0
    fc.ite = 0
    fc.all = []

    for i in range(nb_iterations):
        inv = [random.random() for i in range(16)]
        fit = evalOneMax(inv)
        fc.all.append((fit, inv))
        print(f'{i/nb_iterations*100:05.1f}% -> Now: {fit[0]:.4f}, Best: {fc.fit:.4f}')
        if fc.fit < fit[0]:
            fc.inv = inv
            fc.fit = fit[0]
            fc.ite = i
        bests = sorted(fc.all, key=lambda x: x[0], reverse=True)

    return fc.inv, fc.fit, fc.ite, bests[:100]

def addSolution(df, row):
    #zapisuje plik na dysk
    df = df.append(row, ignore_index=True)
    current_datetime = datetime.now().strftime("%Y-%m-%d_%H-%M-%S")
    file_name = f'{SETUP.name_of_file}_{current_datetime}.csv'
    path = os.path.join(SETUP.path_to_csv, file_name)
    df.to_csv(path, index=False, sep=';')
    return df

if __name__ == "__main__":

```

```

#step 1: wczytanie danych
x_dat, y = readDataFromFiles(r'C:\7pix\7pix_X_y')

#step 2: normalizacja danych
x_dat = normalizeData(x_dat)

#step 3: ustawienie danych w obiekcie SETUP z parametrami globalnymi
SETUP.x_dat = x_dat
SETUP.y     = y

#step 4: przygotowanie obiektu DataFrame do zbierania wyników
df = pd.DataFrame(columns=['ALG', 'FIT', 'ITE', 'IND'])

#-----EKSPERYMENTY OBLICZENIOWE-----
start_population = {}
setup_genetic_algorithm()

#step 5: algorytm losowy
for clf in SETUP.clf_names:
    SETUP.clf_name = clf
    print(f'RANDOM for {clf}')
    I, F, G, SP = random_algorithm(1000)
    df = addSolution(df, {'ALG':f'RND_{clf}', 'FIT':F, 'ITE':G, 'IND':I})
    start_population[clf] = SP

#step 6: algorytm genetyczny dla klasyfikatorów
SETUP.clf_name = clf
print(f'GENETIC for {clf}')
I, F, G = genetic_algorithm(100,30,0.65,0.02)
df = addSolution(df, {'ALG':f'GEN_{clf}', 'FIT':F, 'ITE':G, 'IND':I})

#-----

#step 7: wyświetlenie wyników
print(df)

```

ZALĄCZNIK 2

Kod Python wykorzystywany podczas krosvalidacji przy strategii I

```
import os
import csv
import numpy as np
import pandas as pd
import math
import random
import statistics as stat
from sklearn.model_selection import StratifiedKFold
from sklearn.base import clone

from datetime import datetime
from sklearn.model_selection import StratifiedShuffleSplit
from sklearn.linear_model import SGDClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report
from sklearn.svm import LinearSVC
from sklearn.svm import SVC
from sklearn.ensemble import RandomForestClassifier
from sklearn.ensemble import ExtraTreesClassifier
from sklearn.ensemble import GradientBoostingClassifier, AdaBoostClassifier, VotingClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.manifold import LocallyLinearEmbedding
from sklearn.model_selection import cross_val_score

#funkcja zwracająca wybrany klasyfikator
def getCLF(clf_name):
    if clf_name == 'SGD':
        return SGDClassifier(random_state=1, max_iter=10000)
    elif clf_name == 'SVC':
        return SVC()
    elif clf_name == 'RFC':
        return RandomForestClassifier(n_estimators=300, max_leaf_nodes=64, n_jobs=-1)
    elif clf_name == 'ETC':
        return ExtraTreesClassifier(n_estimators=300, max_leaf_nodes=32, n_jobs=-1)
    elif clf_name == 'GBC':
        return GradientBoostingClassifier(max_depth=2, n_estimators=300, learning_rate=1.0)
    return None

def readDataFromFiles(path_dat= r'D:\!jud\7pix_X_y'):
    y = np.load(path_dat + r'\y.npy')
    x_dat = [np.load(path_dat + rf'\x{i}.npy') for i in range(1,9)]
    return x_dat, y

#funkcja normalizująca dane
def normalizeData(x_dat_org):
    x_dat = []
    for t in x_dat_org:
        x_dat.append(t/(np.max(t)))
    return x_dat

#funkcja przekształcenia danych przez filtr
def getX(individual, x_dat):
    dl= [425,480,545,605,660,725,832,950,]
    q = individual[0::2]
    d = individual[1::2]
    X = np.zeros(x_dat[0].shape)
    i_x, i_y = X.shape
    for i in range(i_x):
        for j in range(i_y):
            licz = 0.001
            for ind, e in enumerate(q):
                xtmp = x_dat[ind]
                X[i,j] += dl[ind] if xtmp[i,j] <= e+d[ind] and xtmp[i,j] > e-d[ind] else 0
                licz += 1 if xtmp[i,j] <= e+d[ind] and xtmp[i,j] > e-d[ind] else 0
            X[i,j] = X[i,j]/licz
    return X
```

```

# krosvalidacja startyfikowana (uwzględniająca w podzbiorach różnorodność klas)
def crosvalid_strat(clf, X_train, y_train, cv = 3, typ='ACC'):
    skfolds = StratifiedKFold(n_splits=cv, shuffle=True,)
    val = []
    for train_index, test_index in skfolds.split(X_train, y_train):
        clone_clf = clone(clf)
        X_train_folds = X_train[train_index]
        y_train_folds = y_train[train_index]
        X_test_fold = X_train[test_index]
        y_test_fold = y_train[test_index]

        clone_clf.fit(X_train_folds, y_train_folds)
        y_pred = clone_clf.predict(X_test_fold)

        report = classification_report(y_test_fold, y_pred, output_dict=True, zero_division=1)
        v = 0
        if(typ=='ACC'): v = report['accuracy']
        if(typ=='F1'): v = report['1.0']['f1-score']
        val.append(v)

    return (val)

# krosvalidacja zwykła
def crosvalid_val(clf, X_train, y_train, cv = 3, typ='ACC'):
    param_fc = ''
    if(typ=='ACC'): param_fc = 'accuracy'
    if(typ=='F1'): param_fc = 'f1_micro'
    scores = cross_val_score(clf, X, y, cv=cv, scoring = param_fc)
    #print(scores)
    return scores

if __name__ == "__main__":
#step 1: wczytanie danych
    x_dat, y = readDataFromFiles(r'D:\!jud\7pix_X_y')

#step 2: normalizacja danych
    x_dat = normalizeData(x_dat)

#step 3: wczytanie filtrów

    solution = []
    path_sol = r'D:\!jud\sol\cross_val\cross_val'
    file_list = ['7pix_ACC.csv', '14pix_ACC.csv', '7pix_F1.csv', '14pix_F1.csv',]

    for file in file_list:
        df = pd.read_csv(os.path.join(path_sol, file), sep=";")
        print(df)
        for index, row in df.iterrows():
            if 'GEN' in row['ALG']:
                #print(row)
                F = row['IND'].replace('[', '').replace(']', '')
                F = [float(i) for i in F.split(',')]
                #print(F, type(F))
                X = getX(F, x_dat)
                ##print(X)
                clf = getCLF(row['ALG'].replace('GEN_', ''))

                sol0 = crosvalid_strat(clf, X, y)
                sol1 = crosvalid_val(clf, X, y)

                dic = {}
                dic['file'] = file;
                dic['ALG'] = row['ALG']
                dic['OLD_FIT'] = row['FIT']

                for i in range(len(sol0)):
                    dic[f'cv_strat{i}'] = sol0[i];
                dic['mean_strat'] = stat.mean(sol0)

```

```
    for i in range(len(sol1)):
        dic[f'cv_norm{i}'] = sol1[i];
    dic['mean_norm'] = stat.mean(sol1)
    solution.append(dic)

df_sol = pd.DataFrame(solution)
df_sol.to_csv(os.path.join(path_sol, f'_sol_.csv'), index=False, sep=';')
print(df_sol)
```

ZAŁĄCZNIK 3

Kod *Python* wykorzystywany podczas eksperymentów obliczeniowych w celu wyznaczenia filtrów z zastosowaniem krosvalidacji - strategia II

```
# import os
import csv
import numpy as np
import pandas as pd
import math
import random
from datetime import datetime
from sklearn.model_selection import StratifiedShuffleSplit
from sklearn.linear_model import SGDClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report
from sklearn.svm import LinearSVC
from sklearn.svm import SVC
from sklearn.ensemble import RandomForestClassifier
from sklearn.ensemble import ExtraTreesClassifier
from sklearn.ensemble import GradientBoostingClassifier, AdaBoostClassifier, VotingClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.manifold import LocallyLinearEmbedding
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import StratifiedKFold
from sklearn.base import clone

import random
from statistics import mean
from deap import base
from deap import creator
from deap import tools

#klasa zawierająca globalne parametry pracy algorytmów
class SETUP:
    clf_names = ['SGD', 'RFC', 'SVC', 'ETC', 'GBC']
    clf_name = 'RFC'

    dl = [425, 480, 545, 605, 660, 725, 832, 950,]
    x_dat = None
    y = None

    #save parametr
    path_to_csv = r'C:\praca\result\corss\7px\ACC'
    name_of_file = 'sol_cv'

    #tmp
    nb_of_inv = -1
    TYP = 'ACC' # Lub 'F1'

#funkcja odczytująca dane
def readDataFromFiles(path_dat= r'C:\praca\7pix\7pix_X_y'):
    y = np.load(path_dat + r'\y.npy')
    x_dat = [np.load(path_dat + rf'\x{i}.npy') for i in range(1,9)]
    return x_dat, y

#funkcja normalizująca dane
def normalizeData(x_dat_org):
    x_dat = []
    for t in x_dat_org:
        x_dat.append(t/(np.max(t)))
    return x_dat
```

```

#funkcja zwracająca wybrany klasyfikator
def getCLF():
    if SETUP.clf_name == 'SGD':
        return SGDClassifier(loss='log')#random_state=1, max_iter=10000)
    elif SETUP.clf_name == 'SVC':
        return SVC()
    elif SETUP.clf_name == 'RFC':
        return RandomForestClassifier(n_estimators=300, max_leaf_nodes=64, n_jobs=-1)
    elif SETUP.clf_name == 'ETC':
        return ExtraTreesClassifier(n_estimators=300, max_leaf_nodes=64, n_jobs=-1)
    elif SETUP.clf_name == 'GBC':
        return GradientBoostingClassifier(max_leaf_nodes=64, n_estimators=300, learning_rate=1.0)
    return None

#funkcja oceny rozwiązania
def evalOneMax(individual):

    def crosvalid_strat(clf, X_train, y_train, cv = 3, typ='ACC'):
        skfolds = StratifiedKFold(n_splits=cv, shuffle=True,)
        val = []
        for train_index, test_index in skfolds.split(X_train, y_train):
            clone_clf = clone(clf)
            X_train_folds = X_train[train_index]
            y_train_folds = y_train[train_index]
            X_test_fold = X_train[test_index]
            y_test_fold = y_train[test_index]

            clone_clf.fit(X_train_folds, y_train_folds)
            y_pred = clone_clf.predict(X_test_fold)

            report = classification_report(y_test_fold, y_pred, output_dict=True, zero_division=1)
            v = 0
            if(typ=='ACC'): v = report['accuracy']
            if(typ=='F1'): v = report['1.0']['f1-score']
            val.append(v)

        return mean(val)

    def crosvalid_val(clf, X_train, y_train, cv = 3, typ='ACC'):
        param_fc = ''
        if(typ=='ACC'): param_fc = 'accuracy'
        if(typ=='F1'): param_fc = 'f1_micro'
        scores = cross_val_score(clf, X, y, cv=cv, scoring = param_fc)
        return mean(scores)

#-----
dl = SETUP.dl
x_dat = SETUP.x_dat
y = SETUP.y
#-----
q = individual[0::2]
d = individual[1::2]
X = np.zeros(x_dat[0].shape)
i_x, i_y = X.shape
for i in range(i_x):
    for j in range(i_y):
        licz = 0.001
        for ind, e in enumerate(q):
            xtmp = x_dat[ind]
            X[i,j] += dl[ind] if xtmp[i,j] <= e+d[ind] and xtmp[i,j]> e-d[ind] else 0
            licz += 1 if xtmp[i,j] <= e+d[ind] and xtmp[i,j]> e-d[ind] else 0
        X[i,j] = X[i,j]/licz

clf = getCLF()

return (crosvalid_val(clf, X, y, cv = 3, typ = SETUP.TYP),)

def setup_genetic_algorithm():
    creator.create("FitnessMax", base.Fitness, weights=(1.0,))
    creator.create("Individual", list, fitness=creator.FitnessMax)

```



```

def genetic_algorithm(nbPop = 10, nbGenerations = 100, CXPB = 0.6, MUTPB= 0.05, startPopulations = None):

#-----
hof = tools.HallOfFame(maxsize=1)
hof.b_Gen = 0

def update_hof(population, nb_Gen):
    fc = hof[0].fitness.values[0] if len(hof)>0 else 0;
    hof.update(population)
    if fc < hof[0].fitness.values[0]:
        hof.b_Gen = nb_Gen
#-----

toolbox = base.Toolbox()
toolbox.register("attr_float", random.uniform, 0, 1)
toolbox.register("individual", tools.initRepeat, creator.Individual, toolbox.attr_float, 16)
toolbox.register("population", tools.initRepeat, list, toolbox.individual)
toolbox.register("evaluate", evalOneMax)
toolbox.register("mate", tools.cxTwoPoint)
toolbox.register("mutate", tools.mutUniformInt, low=0, up=1, indpb=0.3)
toolbox.register("select", tools.selTournament, tournsize=3)
#-----

pop = toolbox.population(n=nbPop)
SETUP.nb_of_inv = -1
for inv in pop:
    SETUP.nb_of_inv += 1
    inv.fitness.values = start_population[SETUP.clf_name][SETUP.nb_of_inv][0]
    for j in range(len(inv)):
        val = start_population[SETUP.clf_name][SETUP.nb_of_inv][1][j]
        inv[j] = val

#-----
fits = [ind.fitness.values[0] for ind in pop]
g = 0
tab_stat = []
while g < nbGenerations:
    g = g + 1
    print(f'Generation {g}/{nbGenerations}')
    offspring = toolbox.select(pop, len(pop))
    offspring = list(map(toolbox.clone, offspring))

    for child1, child2 in zip(offspring[::2], offspring[1::2]):
        if random.random() < CXPB:
            toolbox.mate(child1, child2)
            del child1.fitness.values
            del child2.fitness.values

    for mutant in offspring:
        if random.random() < MUTPB:
            toolbox.mutate(mutant)
            del mutant.fitness.values

    invalid_ind = [ind for ind in offspring if not ind.fitness.valid]
    fitnesses = map(toolbox.evaluate, invalid_ind)
    for ind, fit in zip(invalid_ind, fitnesses):
        ind.fitness.values = fit

    pop[:] = offspring

    fits = [ind.fitness.values[0] for ind in pop]

    length = len(pop)
    mean = sum(fits) / length
    sum2 = sum(x*x for x in fits)
    std = abs(sum2 / length - mean**2)**0.5
    print(f'\t Max:{max(fits):.3f}, Min:{min(fits):.3f}, Avg:{mean:.3f}, Std:{std:.3f}')

    tab_stat.append([min(fits), max(fits), mean, std])
    update_hof(pop, g)

return hof[0], hof[0].fitness.values[0], hof.b_Gen

```

```

def random_algorithm(nb_iterations = 10):
    fc = type('FC',(object,)),{}
    fc.inv = []
    fc.fit = 0
    fc.ite = 0
    fc.all = []

    for i in range(nb_iterations):
        inv = [random.random() for i in range(16)]
        fit = evalOneMax(inv)
        fc.all.append((fit,inv))
        print(f'{i/nb_iterations*100:05.1f}% -> Now: {fit[0]:.4f}, Best: {fc.fit:.4f}')
        if fc.fit < fit[0]:
            fc.inv = inv
            fc.fit = fit[0]
            fc.ite = i
        bests = sorted(fc.all, key=lambda x: x[0], reverse=True)

    return fc.inv, fc.fit, fc.ite, bests[:100]

def addSolution(df, row):
    #zapisuje plik na dysk
    df = df.append(row, ignore_index=True)
    current_datetime = datetime.now().strftime("%Y-%m-%d_%H-%M-%S")
    file_name = f'{SETUP.name_of_file}_{current_datetime}.csv'
    path = os.path.join(SETUP.path_to_csv, file_name)
    df.to_csv(path, index=False, sep=',')
    return df

if __name__ == "__main__":

#step 0:
    SETUP.TYP = 'ACC' #= 'ACC' # lub 'F1'

#step 1: wczytanie danych
    x_dat, y = readDataFromFiles(r'C:\praca\7pix\7pix_X_y')

#step 2: normalizacja danych
    x_dat = normalizeData(x_dat)

#step 3: ustawienie danych w obiekcie SETUP z parametrami globalnymi
    SETUP.x_dat = x_dat
    SETUP.y = y

#step 4: przygotowanie obiektu DataFrame do zbierania wyników
    df = pd.DataFrame(columns=['ALG', 'FIT', 'ITE', 'IND'])

#-----EKSPERYMENTY OBLICZENIOWE-----
    start_population = {}
    setup_genetic_algorithm()

#step 5: algorytm losowy
    for clf in SETUP.clf_names:
        SETUP.clf_name = clf
        print(f'RANDOM for {clf}')
        I, F, G, SP = random_algorithm(1000)
        df = addSolution(df, {'ALG':f'RND_{clf}', 'FIT':F, 'ITE':G, 'IND':I})
        start_population[clf] = SP

#step 6: algorytm genetyczny dla klasyfikatorów
    SETUP.clf_name = clf
    print(f'GENETIC for {clf}')
    I, F, G = genetic_algorithm(100,50,0.65,0.02)
    df = addSolution(df, {'ALG':f'GEN_{clf}', 'FIT':F, 'ITE':G, 'IND':I})

#-----
#step 7: wyświetlenie wyników
    print(df)

```