

# Moduł klasyfikacji chmur punktów dla systemu CDE

Automatyczne rozpoznawanie elementów infrastruktury w danych LiDAR

**Autor:** Maciej Kapała , Filip Skoczylas , Jan Niewiadomski , Dawid Winogrodzki , Mikołaj Banaszak

**Data:** 2025





# Cel modułu

## Główne cele systemu

- Automatyczna klasyfikacja elementów otoczenia z chmur punktów (teren, trawa, roślinność, infrastruktura)
- Minimalizacja ręcznej obróbki danych geodezyjnych
- Przygotowanie danych do dalszych analiz w systemach GIS
- Integracja z istniejącym środowiskiem CDE

## Kluczowe korzyści

- Znaczące przyspieszenie analiz terenowych
- Spójność i standaryzacja danych LiDAR
- Eliminacja błędów ludzkich
- Redukcja kosztów operacyjnych



# Architektura modułu



## Wejście (LAS/LAZ)

Obsługa plików wejściowych i walidacja metadanych



## Preprocessing

Filtracja szumu i normalizacja wysokości punktów



## Segmentacja

Podział chmury na logiczne regiony przestrzenne



## Klasyfikacja

Heurystyki i modele ML do identyfikacji obiektów



## Eksport wyników

Zapis sklasyfikowanych danych i statystyk

# Warstwy systemu

## I/O Layer

Obsługa formatów LAS/LAZ,  
walidacja metadanych i parametrów  
wejściowych

## Preprocessing

Filtracja szumu, normalizacja  
wysokości, wykrywanie outliers

## Core Logic

Algorytmy analizy przestrzennej,  
segmentacja i klasyfikacja punktów

## Export Layer

Zapis nowych klas punktów, generowanie raportów i  
statystyk

## GUI (CustomTkinter)

Interfejs użytkownika z podglądem i konfiguracją  
parametrów

# Zastosowane metody klasyfikacji

1

## Filtry wysokościowe

Analiza wartości min-max oraz percentyli do wyznaczania przedziałów wysokości

2

## Segmentacja przestrzenna

Podział na podstawie gęstości punktów i różnic wysokości w sąsiedztwie

3

## Heurystyki terenowe

- **Trawa:** niskie Z + rozproszona gęstość
- **Roślinność:** duże odchylenia Z
- **Obiekty:** gęsta struktura

4

## Klasteryzacja

DBSCAN i KMeans do wykrywania skupisk i grupowania podobnych obiektów

5

## Feature engineering

Przygotowanie cech pod przyszłe modele uczenia maszynowego

# Pipeline przetwarzania referencyjnej chmury punktów

01

## Wczytanie

Odczyt LAS/LAZ z zastosowaniem skali (scale factors) i offsetów współrzędnych

02

## Segmentacja

Podział punktów na regiony przestrzenne i wykrywanie skupisk (cluster analysis)

03

## Klasyfikacja

Reguły heurystyczne + logika geometryczna z możliwością integracji modeli ML

04

## Eksport

Zapis do nowego LAS z nadanymi wartościami pola classification

# Przykład algorytmu klasyfikacji

## Pseudokod implementacji

```
minZ = min(Z)
maxZ = max(Z)
rangeZ = maxZ - minZ
```

```
threshold_low = minZ + 0.10 * rangeZ
threshold_mid = minZ + 0.40 * rangeZ
```

```
for each point P:
    if P.z < threshold_low:
        P.class = GRASS
    elif P.z < threshold_mid:
        P.class = BUSHES
    else:
        P.class = OBJECT
```

## Przedziały wysokości

**0-10%**

Trawa i powierzchnie niskie

**10-40%**

Krzewy i roślinność niska

**40%+**

Obiekty i roślinność wysoka



# Interfejs użytkownika (GUI)



## Wybór plików

Wybór pliku wejściowego i ustawienie ścieżki wyjściowej z walidacją formatów



## Status i postęp

Podgląd statusu przetwarzania w czasie rzeczywistym z paskiem postępu



## Okno ustawień

Konfiguracja formatu wyjściowego (.las, .laz, .txt), procenta punktów do renderu, parametrów segmentacji



## Obsługa błędów

Automatyczne logowanie zdarzeń i komunikaty o błędach z sugestiami rozwiązań