

Podstawy Sieci Neuronowych

Sprawozdanie końcowe z projektu

Kamil Klepusewicz 280118 Szymon Brodziak 280133

Klasyfikacja obrazów znaków drogowych z wykorzystaniem Głębokich Sieci Neuronowych (CNN)

20.01.2026r.

Spis treści

1.	Wstęp i analiza problemu	2
2.	Przygotowanie danych (Preprocessing).....	2
3.	Architektura sieci.....	2
4.	Proces uczenia i automatyzacja	3
5.	Eksperymenty porównawcze.....	3
6.	Analiza wyników końcowych	4
7.	Podsumowanie (Sukcesy i Porażki)	6

1. Wstęp i analiza problemu

Celem projektu było zaprojektowanie i wytrenowanie modelu sieci neuronowej zdolnego do klasyfikacji zdjęć znaków drogowych. Wykorzystano zbiór danych **GTSRB (German Traffic Sign Recognition Benchmark)**, który jest standardem w dziedzinie rozpoznawania obrazów (Computer Vision).

Wybrano architekturę **Konwolucyjnej Sieci Neuronowej (CNN)**, ponieważ sieci te najlepiej radzą sobie z ekstrakcją cech wizualnych (kształty, krawędzie, kolory) przy zachowaniu odporności na przesunięcia czy zmiany perspektywy obiektu na zdjęciu.

2. Przygotowanie danych (Preprocessing)

Oryginalny zbiór danych charakteryzował się dużym niebalansowaniem klas, co mogło prowadzić do faworyzowania przez sieć znaków występujących częściej. W ramach przygotowania danych wykonano następujące kroki:

1. **Filtracja danych (Rozwiążanie problemu):** Zidentyfikowano klasy rzadkie (posiadające niewielką liczbę próbek). W celu ustabilizowania procesu uczenia odrzucono klasy, które posiadały mniej niż 250 zdjęć. Pozwoliło to na wytrenowanie modelu na bardziej wiarygodnym zestawie danych.
2. **Standaryzacja wymiarów:** Wszystkie obrazy wejściowe zostały przeskalowane do jednolitego formatu 30x30 pikseli w przestrzeni barw RGB (3 kanały).
3. **Normalizacja:** Wartości pikseli (0-255) zostały znormalizowane do zakresu [0, 1], co przyspiesza zbieżność algorytmu gradientowego.
4. **Podział zbioru:** Dane podzielono na zbiór treningowy (80%) oraz walidacyjny (20%), służący do weryfikacji postępów modelu w trakcie uczenia.

3. Architektura sieci

Zaprojektowano model sekwencyjny (Sequential) o strukturze głębokiej, składający się z bloków ekstrakcji cech oraz bloku klasyfikacji. Szczegóły architektury:

- **Warstwa wejściowa:** Input(30, 30, 3).
- **Bloki Konwolucyjne (Ekstrakcja cech):**
 - Zastosowano dwie sekwencje warstw splotowych Conv2D (z filtrami 32 oraz 64) wykorzystujących funkcję aktywacji **ReLU**.
 - Po każdym bloku splotowym zastosowano warstwę redukującą wymiarowość MaxPooling2D (2x2) w celu zmniejszenia nakładu obliczeniowego i wydobycia najważniejszych cech.
- **Blok Klasyfikacji:**
 - Warstwa Flatten (spłaszczenie map cech do wektora).
 - Warstwa gęsta Dense (256 neuronów) z aktywacją **ReLU**.

- Warstwa wyjściowa Dense z funkcją aktywacji **Softmax**, zwracająca rozkład prawdopodobieństwa dla poszczególnych klas.

4. Proces uczenia i automatyzacja

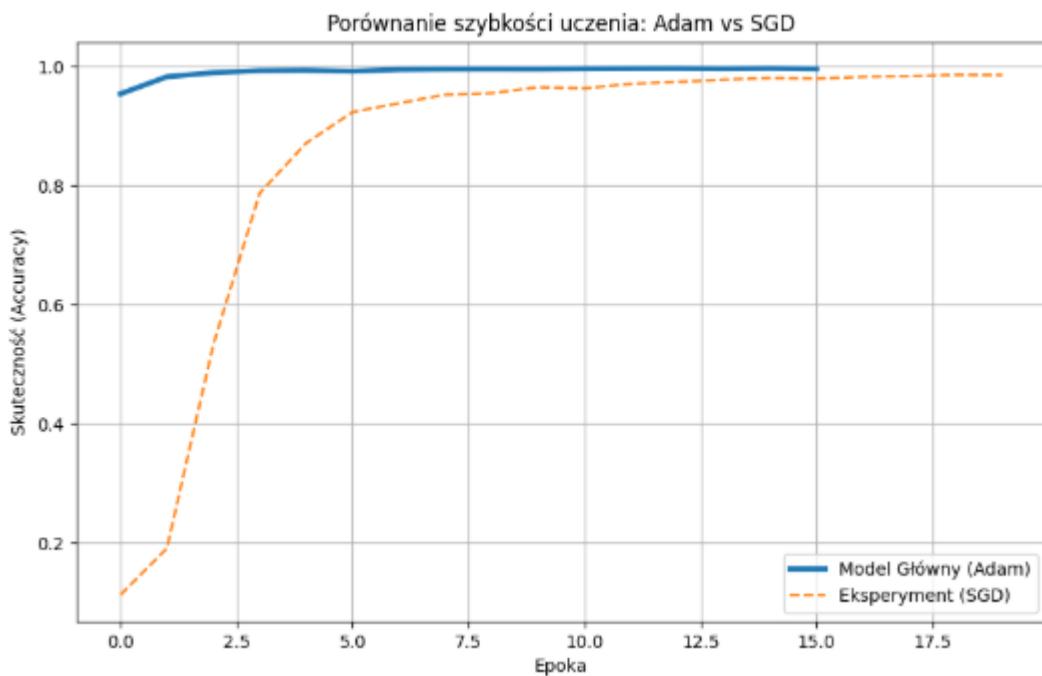
W procesie treningu skonfigurowano następujące parametry:

- **Funkcja straty:** Categorical Crossentropy (adekwatna dla klasyfikacji wieloklasowej).
- **Optymalizator:** Wybrano algorytm **Adam** ze współczynnikiem uczenia learning_rate=0.001.
- **Regularizacja (Unikanie przeuczenia):** Zastosowano warstwy **Dropout** (odrzucanie losowych neuronów z prawdopodobieństwem 0.25 i 0.5). Wymusza to na sieci naukę ogólnych wzorców zamiast zapamiętywania konkretnych pikseli.
- **Automatyzacja (Early Stopping):** Zaimplementowano mechanizm automatycznego zatrzymania treningu. Monitoruje on funkcję straty na zbiorze walidacyjnym (val_loss). Jeśli model nie poprawia wyników przez 5 kolejnych epok, trening jest przerwany. Pozwoliło to na zakończenie uczenia w optymalnym momencie (ok. 14-20 epoki) i zaoszczędzenie zasobów.

5. Eksperymenty porównawcze

Zgodnie z wymaganiami projektowymi przeprowadzono eksperyment mający na celu dobór optymalnego algorytmu uczącego. Porównano działanie nowoczesnego optymalizatora Adam z klasycznym algorytmem spadku gradientu SGD (Stochastic Gradient Descent).

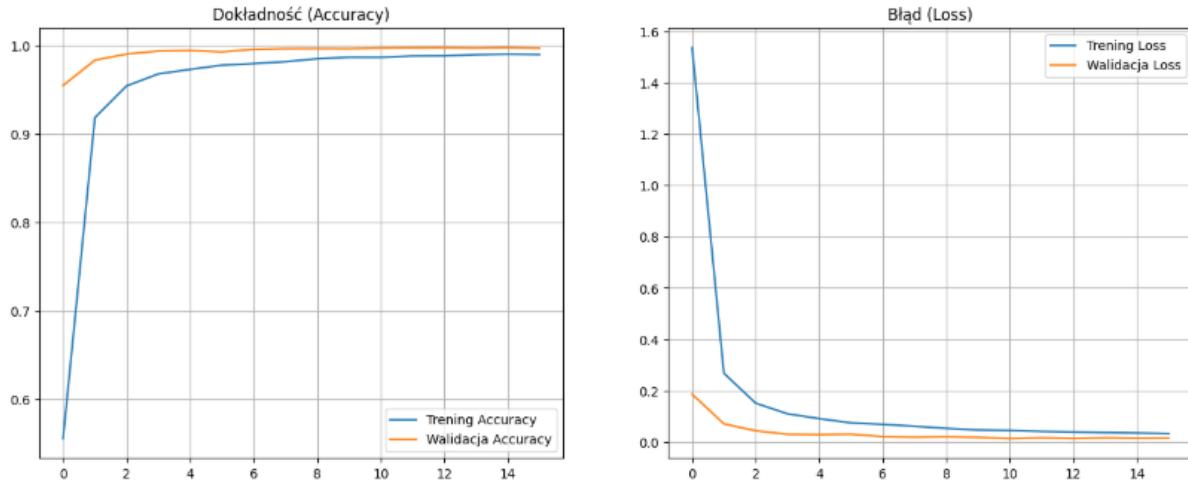
Wyniki eksperymentu: Jak widać na poniższym wykresie, model wykorzystujący algorytm Adam (linia niebieska) osiąga wysoką dokładność (powyżej 90%) już w pierwszych epokach. Model trenowany algorymem SGD (linia pomarańczowa) uczy się znacznie wolniej i wymagałby wielokrotnie dłuższego czasu treningu, aby osiągnąć zbliżone wyniki.



6. Analiza wyników końcowych

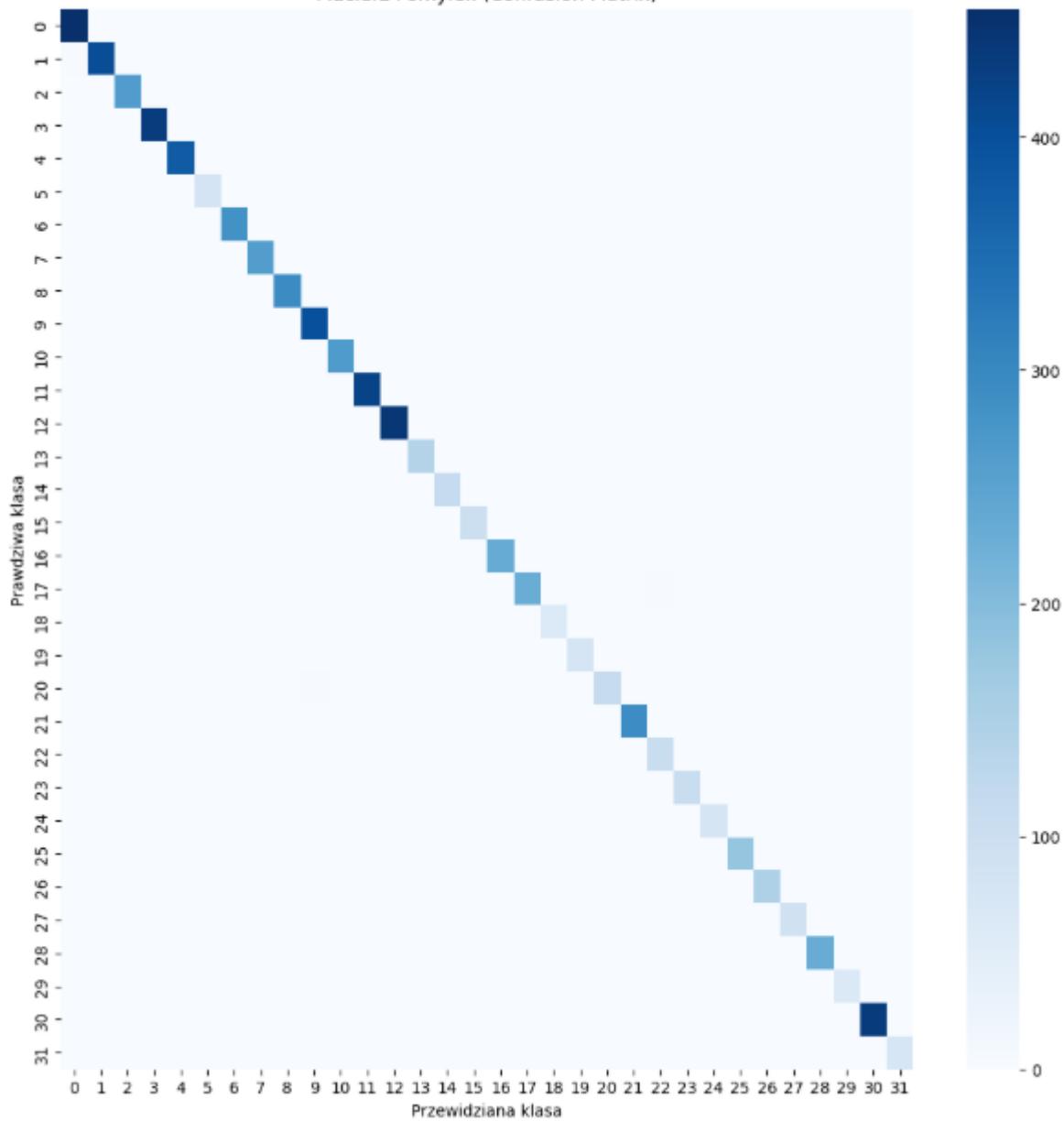
Ostateczny model osiągnął bardzo wysokie wskaźniki skuteczności:

- **Dokładność (Validation Accuracy): 99.77%**
- **Stabilność:** Krzywe uczenia (poniżej) pokazują, że strata na zbiorze walidacyjnym maleje równolegle ze stratą treningową, co świadczy o braku zjawiska przeuczenia (overfittingu).



Analiza błędów (Macierz Pomyłek): Wygenerowano macierz pomyłek (Confusion Matrix) dla zbioru walidacyjnego. Silna dominanta na przekątnej macierzy potwierdza, że model poprawnie klasyfikuje niemal wszystkie próbki. Błędy (wartości poza przekątną) są sporadyczne i dotyczą głównie znaków o bardzo zbliżonym wyglądzie wizualnym.

Macierz Pomylek (Confusion Matrix)



Wizualizacja predykcji: Poniżej przedstawiono przykładowe działanie modelu na losowych zdjęciach ze zbioru testowego. Model wykazuje bardzo wysoką pewność predykcji (często bliską 100%), nawet w przypadku zdjęć rozmytych lub o niskim kontraste.



7. Podsumowanie (Sukcesy i Porażki)

Sukcesy:

- Uzyskanie skuteczności na poziomie **99.7%**, co jest wynikiem bardzo satysfakcjonującym dla tego typu problemu.
- Skuteczne wdrożenie mechanizmów automatyzacji (Early Stopping) oraz regularyzacji (Dropout), co zapobiegło przeuczeniu sieci.
- Poprawna weryfikacja hipotezy o wyższości optymalizatora Adam nad SGD w tym zastosowaniu.

Napotkane problemy i rozwiązania:

- Głównym wyzwaniem było początkowe niezbalansowanie zbioru danych (duża dysproporcja liczby zdjęć między klasami). Problem rozwiązyano poprzez implementację skryptu filtrującego, który usuwał klasy niereprezentatywne przed procesem treningu.

Wnioski: Zaprojektowana sieć konwolucyjna poprawnie rozwiązuje postawiony problem klasyfikacji. Model jest lekki, szybki w treningu i charakteryzuje się wysoką precyzją, co czyni go zdatnym do potencjalnych zastosowań w systemach wspomagania kierowcy.