

# Automatyczne rozpoznawanie pojazdów uprzywilejowanych

**Streszczenie.** W niniejszym artykule zaprezentowano system do automatycznego rozpoznawania pojazdów uprzywilejowanych występujących na terenie Polski, działający na podstawie analizy obrazów i wykorzystujący sztuczne sieci neuronowe. Przygotowano bazę obrazów do przeprowadzenia skuteczności klasyfikacji wybranych rodzajów pojazdów uprzywilejowanych, z uwzględnieniem ich gabarytów i możliwości wykrywania sygnałów świetlnych. Dla najlepszej konfiguracji przebadanych sieci neuronowych i rozdzielczości obrazów osiągnięto ponad 99% dokładność klasyfikacji.

**Abstract.** This article presents a system for automatic recognition of emergency vehicles in Poland, using artificial neural networks and image analysis. A database of images was prepared to carry out the classification tests of selected types of emergency vehicles, taking into account their dimensions and the ability to detect light signals. For the best configuration of the tested neural networks and image resolution, over 99% classification accuracy was achieved. (**Automatic recognition of emergency vehicles**).

**Słowa kluczowe:** rozpoznawanie pojazdów uprzywilejowanych, system wizyjny, analiza obrazów, sztuczne sieci neuronowe.

**Keywords:** recognition of emergency vehicles, vision system, image analysis, artificial neural networks.

## 1. Wprowadzenie

Pojazdy służb ratunkowych, np. policji, straży pożarnej lub karetki pogotowia, używając sygnałów świetlnych i dźwiękowych, uzyskują status pojazdu uprzywilejowanego w ruchu drogowym. Ma to duże znaczenie, zwłaszcza w przypadku ratowania ludzkiego życia. Obowiązkiem kierowcy jest ułatwienie przejazdu pojazdu uprzywilejowanego [1].

Kierowca, w przypadku napotkania pojazdu uprzywilejowanego, powinien zareagować bez zbędnych opóźnień i odpowiednio, np. zjeżdżając z drogi. Istotne jest więc szybkie wykrycie pojazdu uprzywilejowanego. W zależności od okoliczności, w związku z koniecznością utrzymywania ciągłej uwagi i narażeniem na różne bodźce rozprasające, zdarza się, że kierowca nie zareaguje odpowiednio szybko.

Pojazdy uprzywilejowane mogą być wykrywane automatycznie na podstawie generowanych przez nie sygnałów dźwiękowych przy wykorzystaniu metod wykorzystujących sztuczne sieci neuronowe. W pracy [2] ograniczono się do wykrycia jednej klasy syren i nie rozróżniano rodzaju pojazdu uprzywilejowanego.

W artykule [3], na podstawie obrazu z kamery umieszczonej na pojeździe, przy użyciu sztucznych sieci neuronowych, zrealizowano wykrywanie karetek pogotowia. Po wykryciu karetki, na podstawie sygnału dźwiękowego rejestrowanego przez mikrofon i metody opartej na tzw. maszynie wektorów nośnych (Support Vector Machine, SVM), dokonano rozpoznania stanu uprzywilejowania pojazdu [3].

Na obrazach z monitoringu można rozpoznawać pojazdy uprzywilejowane z podziałem na klasy w zależności od występowania niebieskiego, czerwonego lub zmieszanego koloru sygnału świetlnego. W pracy [4] sygnał świetlny był charakteryzowany na podstawie segmentacji wykorzystującej modele barw HSV i RGB, a klasyfikacja odbywała się przy wykorzystaniu metody SVM.

Także w produkowanych współczesnych samochodach pojawiają się systemy do wykrywania pojazdów uprzywilejowanych, które dźwiękowo i wizualnie ostrzegają kierowcę. Wykrycie realizowane jest na podstawie analizy sygnałów z mikrofonów i kamer [5–7].

W niniejszej pracy zaproponowano i przetestowano możliwości automatycznego rozpoznawania wybranych pojazdów uprzywilejowanych na podstawie analizy obrazów. Problem detekcji pojazdów jest szeroko opisanym zagadnieniem, w którym wykorzystuje się całe spektrum metod [8], skupiono się więc na etapie klasyfikacji w charakterystycznych, określonych warunkach. Uwzględniono podział na wybrane rodzaje pojazdów uprzywilejowanych występu-

jące w polskich warunkach, a do klasyfikacji wykorzystano sztuczne sieci neuronowe. Badania eksperymentalne wykonano na specjalnie przygotowanej do tego bazy obrazów.

## 2. Baza obrazów pojazdów uprzywilejowanych

W celu wytrenowania i przetestowania sztucznych sieci neuronowych przygotowano odpowiednią bazę obrazów.

Według wiedzy autorów, nie ma ogólnodostępnych zbiorów zdjęć pojazdów uprzywilejowanych w polskich warunkach drogowych. Fotografie zostały więc zebrane poprzez pobranie ich przy wykorzystaniu serwisu Grafika Google [9]. Wykorzystano kryterium wyszukiwania polegające na wyborze tylko darmowych obrazów. Kierowano się oficjalnym opisem malowania pojazdów uprzywilejowanych zawierającym charakterystyczne kolory karoserii, występujące oznaczenia, ich barwy i wymagane napisy, zawartym w odpowiednich aktach prawnych [10, 11].

Jako odniesienie zawierające pojazdy nie będące pojazdami uprzywilejowanymi, wykorzystano, poza nielicznymi wyjątkami, zbiór [12]. Ostatecznie, w bazie obrazów do przeprowadzenia analiz znalazły się pojazdy pogotowia, policji, straży pożarnej, straży miejskiej, wojska oraz pojazdy nie będące pojazdami uprzywilejowanymi. Dobór pięciu wymienionych rodzajów pojazdów uprzywilejowanych wynikał z największej dostępności zdjęć i powszechności występowania tych pojazdów na drogach.

Zdjęcia sprawdzono pod kątem duplikatów oraz podobieństwa, gdyż w procesie ich pozyskiwania zdarzało się, że automatycznie pobrano dwie takie same fotografie lub zdjęcie i jego fragment. Do tego zadania użyto programu o nazwie „Czkawka” [13]. Umożliwia on wykrycie duplikatów na podstawie rozmiaru, nazwy i podobieństwa na bazie różnych algorytmów i filtrów. Część zdjęć zweryfikowano manualnie.

Następnie, z każdego zdjęcia manualnie wycinano prostokątne obszary zainteresowania (ang. region of interest, ROI) obejmujące pojazdy, pod kątem przeprowadzenia testów skuteczności klasyfikacji. Manualne oznaczanie obszarów zainteresowania mogło by również mieć zastosowanie w testach detekcji. Z kolei, w docelowej aplikacji użytkownika używającej wytrenowane i przetestowane modele sztucznych sieci neuronowych, proces rozpoznawania może przebiegać całkowicie automatycznie.

W rezultacie w bazie zebrano 6731 obrazów. Rozdzielczość obrazów była różnorodna, w przedziale od kilkadziesiąt do kilkadziesiąt do kilka tysięcy na kilka tysięcy pikseli.

W końcowym zbiorze danych zawarto zdjęcia pojazdów sfotografowane pod różnorodnymi kątami i z różnym poziomem oświetlenia, również nocne. W części zdjęć widoczne były jedynie fragmenty pojazdów lub były one przyśłonięte innymi pojazdami, bądź też obiektami z otoczenia. W przypadkach, gdy pojazd na zdjęciu źródłowym znajdował się dalszej perspektywie, powstawały obrazy o niskiej rozdzielczości. Mimo tego, uwzględniono je w zbiorze danych.

Zdjęcia zostały podzielone na pięć zestawów klas. Podział ustalono na podstawie cech charakterystycznych pojazdów, takich jak rodzaj pojazdów, gabaryty oraz występowanie w danym momencie włączonego sygnału świetlnego. W każdym zestawie klas obrazy zostały podzielone na rozłączne zbiory: treningowy, walidacyjny i testowy, w proporcjach, odpowiednio, około 70%, 15% i 15% ze wszystkich zdjęć danej klasy. W tabeli 1 przedstawiono podział zdjęć na zestawy klas.

Tabela 1. Liczba obrazów w poszczególnych zestawach klas

| zestaw | klasy                      |                          |                           |                          |        |                    | suma |
|--------|----------------------------|--------------------------|---------------------------|--------------------------|--------|--------------------|------|
|        | pogotowie                  | policja                  | straż pożarna             | straż miejska            | wojsko | nieuprzywilejowane |      |
| 1      |                            |                          |                           |                          | –      |                    | 5750 |
| 2      |                            | 1097                     | 1730                      |                          | 981    |                    | 6731 |
| 3      | 780                        | 316 / 781 (duże / małe)  | 1205 / 525 (duże / małe)  | 775                      |        | 1368               | 5750 |
| 4      | 438 / 342 (bez/na sygn.)   | 688 / 409 (bez/na sygn.) | 1120 / 610 (bez/na sygn.) | 635 / 140 (bez/na sygn.) | –      |                    |      |
| 5      | 2881 / 1501 (bez/na sygn.) |                          |                           |                          |        |                    |      |

Do pierwszego zestawu klas (5 klas), wybrano podstawowe rodzaje pojazdów uprzywilejowanych, tj. pojazdy pogotowia, policji, straży pożarnej i straży miejskiej, a także pojazdy nie będące pojazdami uprzywilejowanymi.

W drugim zestawie klas (6 klas), uwzględniono dodatkowo klasę zawierającą pojazdy wojskowe.

W trzecim zestawie klas (7 klas), oprócz rodzaju pojazdów, w przypadku pojazdów policyjnych i straży pożarnej, uwzględniono ich rozmiar, rozróżniając pojazdy małe i duże. Celem rozpoznawania pojazdów pod względem ich rozmiarów mogłoby być np. ostrzeganie kierowców przed zbliżaniem się dużego pojazdu, na który trzeba zostawić więcej miejsca, aby ten mógł swobodnie przejechać.

W czwartym zestawie klas (9 klas), oprócz rodzaju pojazdów, uwzględniono emisję sygnału świetlnego, tj. czy pojazd ma włączony sygnał świetlny („na sygnale”), czy też sygnał świetlny jest wyłączony („bez sygnału”). Rozpoznawanie pojazdów bez włączonego sygnału świetlnego może także być przydatne. Przykładowo, pojazd pogotowia ratunkowego stojący, bądź jadący w pobliżu, może nagle włączyć sygnały świetlne i tym samym stać się pojazdem uprzywilejowanym. Dzięki rozpoznaniu takiej sytuacji, kierowca mógłby być przygotowany na konieczność ustąpienia miejsca temu pojazdowi. Ponadto mogą zdarzyć się sytuacje, że system nie zadziała prawidłowo i sklasyfikuje pojazd z włączonymi sygnałami świetlnymi jako pojazd bez sygnałów świetlnych. W takim przypadku, lepsze jest wykrycie pojazdu jako pojazd bez sygnałów świetlnych niż całkowity brak wykrycia i ostrzeżenia.

Piąty zestaw klas (3 klasy) zawierał obrazy pojazdów uprzywilejowanych podzielone ze względu na załączenie lub wyłączenie sygnału świetlnego, bez uwzględniania rodzaju pojazdu. Takie rozpoznawanie miało na celu ogólne wykrycie pojazdów uprzywilejowanych emitujących sygnał świetlny, co samo w sobie mogłoby stanowić istotną informację dla kierowcy.

W bazie danych obrazów nie uwzględniono przypadków, w których pojazd cywilny, tj. bez typowego malowania, staje się pojazdem uprzywilejowanym tylko na podstawie sygnałów świetlnych i dźwiękowych.

Suma obrazów w większości zestawów klas (z wyjątkiem drugiego), wyniosła 5750. Drugi zestaw klas zawierał dodatkowo pojazdy wojskowe, a więc całą bazę obrazów (patrz tabela 1).

Tak przygotowana baza obrazów, w postaci wydzielonych zestawów klas i zbiorów, posłużyła do treningów, walidacji i testów sztucznych sieci neuronowych.

### 3. Architektura sztucznych sieci neuronowych

Automatyczne rozpoznawanie pojazdów uprzywilejowanych zostało zrealizowane na podstawie klasyfikacji obiektów na obrazach z przygotowanej bazy. Do tego zadania wykorzystano różne architektury spłotowych sztucznych sieci neuronowych. Na poszczególnych etapach realizacji procesu treningu i testowania wykorzystano różne narzędzia programistyczne.

Podstawowym narzędziem, które posłużyło do realizacji wszystkich etapów tworzenia oprogramowania, był język programowania Python [14]. Całość kodu została napisana przy pomocy edytora kodu Visual Studio Code [15], z wykorzystaniem otwartego środowiska do interaktywnych obliczeń Jupyter [16].

Trening spłotowych sieci neuronowych i przeprowadzone testy zostały wykonane z wykorzystaniem techniki GPGPU (ang. general purpose computing on graphics processing units). Użyto karty graficznej Nvida GeForce GTX 1050 Ti w wersji z 4 GB pamięci [17]. Głównymi bibliotekami, które zostały wykorzystane w procesie treningu i testu, były Keras [18] i TensorFlow [19].

Przed przeprowadzeniem treningu ustalono rozdzielczość, do jakiej będą skalowane obrazy przed przekazaniem ich dalej do sztucznych sieci neuronowych oraz uruchomiono funkcję wstępnie przetwarzającą dane. Mając tak przygotowany zbiór danych, możliwym stało się przejście do kolejnego kroku, jakim było utworzenie modeli poszczególnych sieci.

Jako źródło modeli wykorzystano bibliotekę Keras, która oferuje szeroki wybór gotowych architektur sieci [18]. Przetestowano sześć architektur spłotowych sieci neuronowych: DenseNet121 [20], EfficientNetV2B0 [21], InceptionV3 [22], MobileNetV3Large, MobileNetV3SmallMinimalistic [23] oraz ResNet50V2 [24].

Jako początkowe wartości wag badanych sieci, wykorzystano wartości wstępnie wytrenowane na zbiorze danych z konkursu ImageNet [25], dostępnych poprzez bibliotekę Keras. Następnie, w użytych modelach zamieniono na nową dotychczasową finalną warstwę w pełni połączoną [18]. Wykorzystaną w tej warstwie funkcją aktywacji była znormalizowana funkcja wykładnicza, a liczbą wyjść była równa liczbie zadanych klas.

Tak przygotowany model został następnie poddany treningowi. Jako funkcję błędu wykorzystano funkcję wieloklasowej entropii krzyżowej. W roli optymalizatora użyto algorytmu Adam z szybkością uczenia się na poziomie 0,0001. Ponadto, w trakcie treningu sieci, wykorzystano algorytmy redukcji szybkości uczenia się, wczesnego zatrzymania oraz zapisu najlepszych wag modelu [18].

Kryterium zatrzymania dla wszystkich architektur był brak dalszej poprawy monitorowanej metryki (zmiana wartości mniejsza niż 0,01) w trakcie zadanej liczby epok równej 6. Monitorowaną metryką była funkcja błędu.

W celu realizacji procesu testowania, zastosowano moduły NumPy [26] oraz „pandas” [27], które posłużyły odpowiedniej strukturyzacji otrzymanych danych wynikowych. Do wykonania właściwych testów posłużył pakiet „scikit-

learn” [28]. Wykorzystano w nim funkcje pozwalające uzyskać macierze pomyłek i miary jakości klasyfikacji. Wizualizacji zebranych danych dokonano za pośrednictwem biblioteki Matplotlib [29].

Proces uczenia przeprowadzono kolejno dla wszystkich rozważanych architektur dla jednej rozdzielczości obrazów ze zbioru obrazów z przygotowanej bazy. Następnie, po przeanalizowaniu wyników, wybrano potencjalnie najlepszą architekturę, dla której przeprowadzono kolejne treningi i testy przy różnych rozdzielczościach wejściowych. Na podstawie tego etapu, wyłoniono najlepszą konfigurację architektury i rozdzielczości wejściowej. Kolejnym etapem było wyszkolenie wybranej architektury na pozostałych zbiorach treningowych.

Ostatecznie, przygotowane oprogramowanie pozwoliło na przeprowadzenie testów skuteczności klasyfikacji.

#### 4. Eksperyment

W ramach eksperymentu przeprowadzono testy skuteczności przygotowanego oprogramowania na utworzonej bazie danych. We wszystkich testach, jako odniesienie, występowała klasa pojazdów nieuprzywilejowanych. Uzyskane wyniki przedstawiono w tabeli 2.

Tabela 2. Wyniki testów klasyfikacji dla pięciu zestawów klas dla wybranych architektur sieci i rozdzielczości obrazów

| zestaw | klasy  | architektura      | rozdzielczość obrazów [px] | dokładność [%] |
|--------|--|-------------------|----------------------------|----------------|
| 1      | pogotowie, policja, straż pożarna, straż miejska, nieuprzywilejowane   | DenseNet121       | 128x128                    | 96,52          |
|        |  | EfficientNetV2B0  |                            | 97,80          |
|        |  | InceptionV3       |                            | 95,02          |
|        |  | MobileNetV3 Large |                            | 95,83          |
|        |  | MobileNetV3Small  |                            | 94,09          |
|        |  | ResNet50V2        |                            | 93,97          |
|        |  | EfficientNetV2B0  | 48x48                      | 81,58          |
|        |  |                   | 96x96                      | 94,09          |
|        |  |                   | 224x224                    | 99,07          |
|        |  |                   | 336x336                    | 98,15          |
|        |  | 448x448           | 98,38                      |                |
|        |  | 560x560           | 98,26                      |                |
|        |  | 672x672           | 83,89                      |                |
| 2      | iw., wojsko  |                   |                            | 98,32          |
| 3      | pogotowie, policja (duże/male), straż pożarna (duże/male), straż miejska, nieuprzywilejowane                                     |                   |                            | 96,06          |
| 4      | pogotowie (bez/na sygn.), policja (bez/na sygn.), straż pożarna (bez/na sygn.), straż miejska (bez/na sygn.), nieuprzywilejowane | EfficientNetV2B0  | 224x224                    | 90,85          |
| 5      | uprzywilejowane (bez/na sygn.), nieuprzywilejowane   |                   |                            | 94,55          |

W pierwszych testach wykonano porównanie sześciu wytrenowanych modeli splotowych sieci neuronowych. Porównanie obejmowało klasyfikację dla podstawowych klas pojazdów uprzywilejowanych (zestaw 1), o rozdzielczości obrazów równej 128x128 pikseli. Największą wartość dokładności, wynoszącą około 98%, uzyskano przy użyciu modelu sieci EfficientNetV2B0.

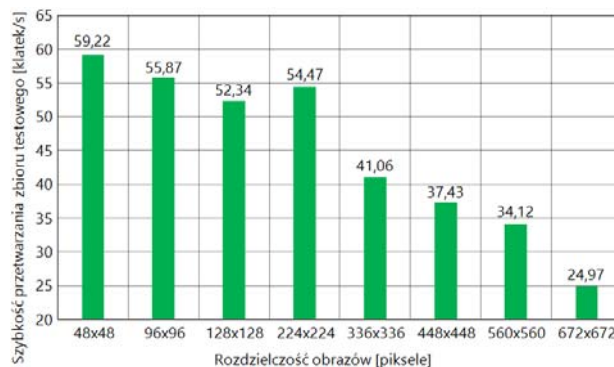
Dla tego modelu przebadano inne rozdzielczości obrazów: dwie niższe oraz pięć wyższych. Oprócz wpływu rozdzielczości na dokładność klasyfikacji zbadano także jej wpływ na szybkość przetwarzania. Wyniki przedstawiono na rysunku 1. Generalnie, przy wzroście rozdzielczości szybkość przetwarzania obrazów malała. Wyjątkiem była

wysoka wartość szybkości przetwarzania dla obrazów o rozdzielczości 224x224 piksele. Wynik ten może być związany ze strukturą modelu sieci zrealizowaną domyślnie dla takiej właśnie rozdzielczości. Najwyższą wartość dokładności, wynoszącą ponad 99%, osiągnięto również dla tej rozdzielczości (tabela 2). Wynikowe miary jakości klasyfikacji przedstawiono na rysunku 2. Dla pojazdów pogotowia uzyskano stuprocentową czułość, co oznacza, że wszystkie takie pojazdy zostały prawidłowo rozpoznane.

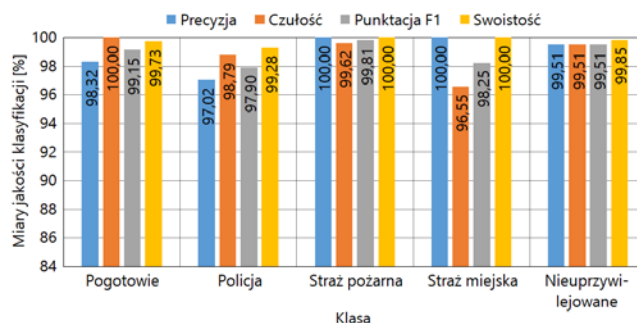
Kolejne testy przeprowadzono dla modelu EfficientNetV2B0 i rozdzielczości 224x224 piksele. Przetestowano wpływ dodania nowej klasy pojazdów, tj. pojazdów wojskowych (zestaw 2), na jakość klasyfikacji modelu. Dokładność zmniejszyła się, jednak wciąż wynosiła ponad 98%.

W wyniku klasyfikacji podstawowych rodzajów pojazdów uprzywilejowanych z uwzględnieniem rozmiaru pojazdów policyjnych i straży pożarnej (zestaw 3), uzyskano dokładność ponad 96%.

W następnym teście sprawdzono jak dokładnie dokonywana jest klasyfikacja podstawowych rodzajów pojazdów uprzywilejowanych, dodatkowo rozróżniając fakt załączenia sygnału świetlnego (zestaw 4). Wynikowa dokładność wyniosła około 91%.



Rys. 1. Szybkość przetwarzania obrazów ze zbioru testowego dla pierwszego zestawu klas dla architektury EfficientNetV2B0 dla różnych rozdzielczości obrazów



Rys. 2. Wartości miar jakości klasyfikacji uzyskane dla pierwszego zestawu klas dla architektury EfficientNetV2B0 dla rozdzielczości 224x224 piksele

Przeprowadzono również test, mający na uwadze zbadanie skuteczności klasyfikacji pojazdów rozróżnialnych na podstawie występowania, bądź też nie występowania sygnałów świetlnych, bez rozróżnienia na rodzaj pojazdu uprzywilejowanego (zestaw 5). Uzyskano, większą niż poprzednio, dokładność wynoszącą około 95%.

#### 5. Podsumowanie i wnioski

Na podstawie przeprowadzonych testów skuteczności przygotowanego oprogramowania można stwierdzić, że przy użyciu metod sztucznej inteligencji zastosowanych do analizy obrazów możliwe jest automatyczne rozpoznawanie wybranych rodzajów pojazdów uprzywilejowanych występujących w Polsce, z dokładnością przekraczającą 99%. Moż-

live jest również ich automatyczne klasyfikowanie pod względem gabarytów oraz wykrycie załączenia sygnałów świetlnych. Można też rozróżnić pojazd w malowaniu pojazdu uprzywilejowanego, nie emitujący sygnału świetlnego, potencjalnie mogący stać się pojazdem uczestniczącym w akcji.

W dalszych badaniach, oprócz zwiększenia liczby obrazów w bazie, planowane jest zwiększenie liczby klas uwzględniających innego typu pojazdy uprzywilejowane oraz pojazdy bez charakterystycznego malowania, ale z włączonym sygnałem świetlnym. Do analizy można dodatkowo wykorzystać obrazy pochodzące bezpośrednio z wideorejestratorów samochodowych lub smartfonów, korzystając z rozdzielczości obrazów wejściowych w formacie np. HD lub 4K.

Docelowa aplikacja może realizować kompletny proces rozpoznawania pojazdów uprzywilejowanych, tj. detekcji i klasyfikacji.

*Badania sfinansowano z subwencji badawczej 0211/SBAD/0223.*

**Autorzy:** dr inż. Julian Balcerek, Politechnika Poznańska, Instytut Automatyki i Robotyki, ul. Piotrowo 3a, 60-965 Poznań, E-mail: [julian.balcerek@put.poznan.pl](mailto:julian.balcerek@put.poznan.pl); dr inż. Paweł Pawłowski, Politechnika Poznańska, Instytut Automatyki i Robotyki, ul. Piotrowo 3a, 60-965 Poznań, E-mail: [pawel.pawlowski@put.poznan.pl](mailto:pawel.pawlowski@put.poznan.pl); inż. Grzegorz Blajer, Politechnika Poznańska, Instytut Automatyki i Robotyki, ul. Piotrowo 3a, 60-965 Poznań, E-mail: [grzegorz.blajer@student.put.poznan.pl](mailto:grzegorz.blajer@student.put.poznan.pl); inż. Jakub Filipkiewicz, Politechnika Poznańska, Instytut Automatyki i Robotyki, ul. Piotrowo 3a, 60-965 Poznań, E-mail: [jakub.filipkiewicz@student.put.poznan.pl](mailto:jakub.filipkiewicz@student.put.poznan.pl); inż. Kamil Koćwin, Politechnika Poznańska, Instytut Automatyki i Robotyki, ul. Piotrowo 3a, 60-965 Poznań, E-mail: [kamil.kocwin@student.put.poznan.pl](mailto:kamil.kocwin@student.put.poznan.pl).

## LITERATURA

- [1] Ratajczak P., Pojazdy uprzywilejowane – obowiązki kierowców wobec pojazdów pomagających ratować ludzkie życie, [https://d.olsonaska.policja.gov.pl/wr1/aktualnosci/biezace-inf/72097\\_Pojazdy-uprzywilejowane-obowiazki-kierowcow-wobec-pojazdow-pomagajacych-ratowac.html](https://d.olsonaska.policja.gov.pl/wr1/aktualnosci/biezace-inf/72097_Pojazdy-uprzywilejowane-obowiazki-kierowcow-wobec-pojazdow-pomagajacych-ratowac.html), 07.07.2022 (dostęp: 01.04.2023).
- [2] Islam Z., Abdel-Aty M.A., Real-time Emergency Vehicle Event Detection Using Audio Data, ArXiv, abs/2202.01367, 2022.
- [3] Garg A., Gupta A.K., Shrivastava D., Didwania Y., Bora P.J., 2019, Emergency Vehicle Detection by Autonomous Vehicle, International Journal of Engineering Research & Technology (IJERT), Volume 08, Issue 05, May 2019.
- [4] Razalli H., Ramli R., Alkawaz M.H., Emergency Vehicle Recognition and Classification Method Using HSV Color Segmentation, 2020 16th IEEE International Colloquium on Signal Processing & Its Applications (CSPA), Langkawi, Malaysia, pp. 284-289, 2020, doi: 10.1109/CSPA48992.2020.9068695.
- [5] Cerence, Cerence Emergency Vehicle Detection (EVD), <https://www.cerence.com>, 2023 (dostęp: 01.04.2023).
- [6] Mercedes-Benz Group Media, Mercedes-Benz otrzymuje pierwszą na świecie międzynarodową homologację dla systemu jazdy warunkowo zautomatyzowanej, <https://media.mercedes-benz.com/mercedes-benz-otrzymuje-pierwsza-na-swiecie-miedzynarodowa-homologacje-dla-systemu-jazdy-warunkowo-zautomatyzowanej/>, 10.12.2021 (dostęp: 19.03.2023).
- [7] Tesla, Model 3, Instrukcja użytkownika, Wersja oprogramowania: 2023.6, Europe, Autopilot, Automatyczne kierowanie, Korzystanie z funkcji automatycznego kierowania, str. 113-114, [https://www.tesla.com/ownersmanual/model3/pl\\_pl/Owners\\_Manual.pdf](https://www.tesla.com/ownersmanual/model3/pl_pl/Owners_Manual.pdf), 28.02.2023 (dostęp: 01.04.2023).
- [8] Wang H., Yu Y., Cai Y., Chen X., Chen L., Liu Q., A Comparative Study of State-of-the-Art Deep Learning Algorithms for Vehicle Detection, in IEEE Intelligent Transportation Systems Magazine, vol. 11, no. 2, pp. 82-95, Summer 2019, doi: 10.1109/MITS.2019.2903518.
- [9] Google, Grafika Google, <https://www.google.com/imghp?hl=PL> (dostęp: 17.01.2023).
- [10] Obwieszczenie Ministra Infrastruktury i Budownictwa z dnia 27 października 2016 r. w sprawie ogłoszenia jednolitego tekstu rozporządzenia Ministra Infrastruktury w sprawie warunków technicznych pojazdów oraz zakresu ich niezbędnego wyposażenia, Dziennik Ustaw Rzeczypospolitej Polskiej, Poz. 2022, Warszawa, dnia 15 grudnia 2016 r.
- [11] Rozporządzenie Ministra Infrastruktury z dnia 3 stycznia 2022 r. zmieniające rozporządzenie w sprawie warunków technicznych pojazdów oraz zakresu ich niezbędnego wyposażenia, Dziennik Ustaw Rzeczypospolitej Polskiej, Poz. 122, Warszawa, dnia 18 stycznia 2022 r.
- [12] Yang L., Luo P., Loy C.C., Tang X., A large-scale car dataset for fine-grained categorization and verification, 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, pp. 3973-3981, 2015, doi: 10.1109/CVPR.2015.7299023.
- [13] Mikrut R., Qarmin/czkawka, Multi functional app to find duplicates, empty folders, similar images etc., GitHub, <https://github.com/qarmin/czkawka> (dostęp: 17.01.2023).
- [14] Python Software Foundation, Python, <https://www.python.org> (dostęp: 17.04.2023).
- [15] Microsoft, Visual Studio Code, Code Editing. Redefined., <https://code.visualstudio.com> (dostęp: 17.04.2023).
- [16] Project Jupyter, Jupyter, <https://jupyter.org> (dostęp: 17.04.2023).
- [17] NVIDIA Corporation, GeForce GTX 1050 Ti, Specifications, 2021, <https://www.nvidia.com/en-gb/force/graphics-cards/geforce-gtx-1050-ti/specifications/> (dostęp: 17.04.2023).
- [18] Keras, Simple. Flexible. Powerful., <https://keras.io> (dostęp: 17.04.2023).
- [19] Abadi M., Agarwal A., Barham P., Brevdo E., Chen Z., Citro C., Corrado G. S., Davis A., Dean J., Devin M., Ghemawat S., Goodfellow I., Harp A., Irving G., Isard M., Jozefowicz R., Jia Y., Kaiser Ł., Kudlur M., Levenberg J., Mané D., Schuster M., Monga R., Moore S., Murray D., Olah C., Shlens J., Steiner B., Sutskever I., Talwar K., Tucker P., Vanhoucke V., Vasudevan V., Viégas F., Vinyals O., Warden P., Wattenberg M., Wicke M., Yu Y., Zheng X., TensorFlow: Large-scale machine learning on heterogeneous systems, 2015, software available from tensorflow.org.
- [20] Huang G., Liu Z., Van Der Maaten L., Weinberger K. Q., Densely Connected Convolutional Networks, 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, pp. 2261-2269, 2017, doi: 10.1109/CVPR.2017.243.
- [21] Tan M., Le Q. V., EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks, Proceedings of the 36th International Conference on Machine Learning, Proceedings of Machine Learning Research, editors: Kamalika C., Salakhutdinov R., vol. 97, pp. 6105-6114, 2019.
- [22] Szegedy C., Liu W., Jia Y., Sermanet P., Reed S., Anguelov D., Erhan D., Vanhoucke V., Rabinovich A., Going deeper with convolutions, 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, pp. 1-9, 2015, doi: 10.1109/CVPR.2015.7298594.
- [23] Howard A. G., Zhu M., Chen B., Kalenichenko D., Wang W., Weyand T., Andreetto M., Adam H., MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications, arXiv:1704.04861, 2017, doi: 10.48550/arXiv.1704.04861.
- [24] He K., Zhang X., Ren S., Sun J., Deep Residual Learning for Image Recognition, 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, pp. 770-778, 2016, doi: 10.1109/CVPR.2016.90.
- [25] Russakovsky O., Deng J., Su H., Krause J., Satheesh S., Ma S., Huang Z., Karpathy A., Khosla A., Bernstein M., Berg A. C., Fei-Fei L., ImageNet Large Scale Visual Recognition Challenge, International Journal of Computer Vision, 115, pp. 211-252, 2015, doi: 10.1007/s11263-015-0816-y.
- [26] NumPy, The fundamental package for scientific computing with Python, <https://numpy.org> (dostęp: 18.04.2023).
- [27] Pandas via NumFOCUS, Inc. Hosted by OVHcloud, <https://pandas.pydata.org> (dostęp: 18.04.2023).
- [28] Scikit-learn, Machine Learning in Python, <https://scikit-learn.org/stable/> (dostęp: 18.04.2023).
- [29] The Matplotlib development team, Matplotlib: Visualization with Python, <https://matplotlib.org> (dostęp: 18.04.2023).