

## **System graficznego rozpoznawania obiektów ruchomych**

Mateusz Majewski<sup>\*</sup>, Dariusz Pałka<sup>\*\*</sup>

Warszawska Wyższa Szkoła Informatyki

---

### **Abstrakt**

System graficznego rozpoznawania ruchomych obiektów to rozwiązanie, które może być bardzo szeroko stosowane w życiu codziennym. Taki system powinien rozpoznać obiekty określone przez użytkownika przed jego uruchomieniem. Algorytmy stosowane w tym systemie powinny mieć dużą szybkość i dokładność. Istnieje wiele rozwiązań dla takich algorytmów, które wykorzystują wiele różnych technik i ustawień. W artykule przedstawiono podstawowe pojęcia i zasady systemu rozpoznawania obiektów mobilnych, a także przedstawiono koncepcję nowego systemu graficznego do rozpoznawania obiektów mobilnych, który ma większe możliwości niż systemy dotychczas używane.

**Słowa kluczowe** – grafika, rozpoznawanie obiektów, rozpoznawanie obiektów ruchomych

---

<sup>\*</sup> E-mail: m.majewski@poczta.wysi.edu.pl

<sup>\*\*</sup> E-mail: dpalka@poczta.wysi.edu.pl

## 1. Wprowadzenie

Obraz dynamiczny jest frazą, która może być definiowana poprzez film lub animację komputerową. Taki obraz różni się w zależności od sposobu zapisywania oraz od wykorzystanego sprzętu służącego do jego przechwytywania. Film dzieli się na sekwencję obrazów odtwarzanych w danej prędkości. Cechami filmu jest jego rozdzielczość oraz szybkość odtwarzania sekwencji obrazów podawanych w klatkach na sekundę.

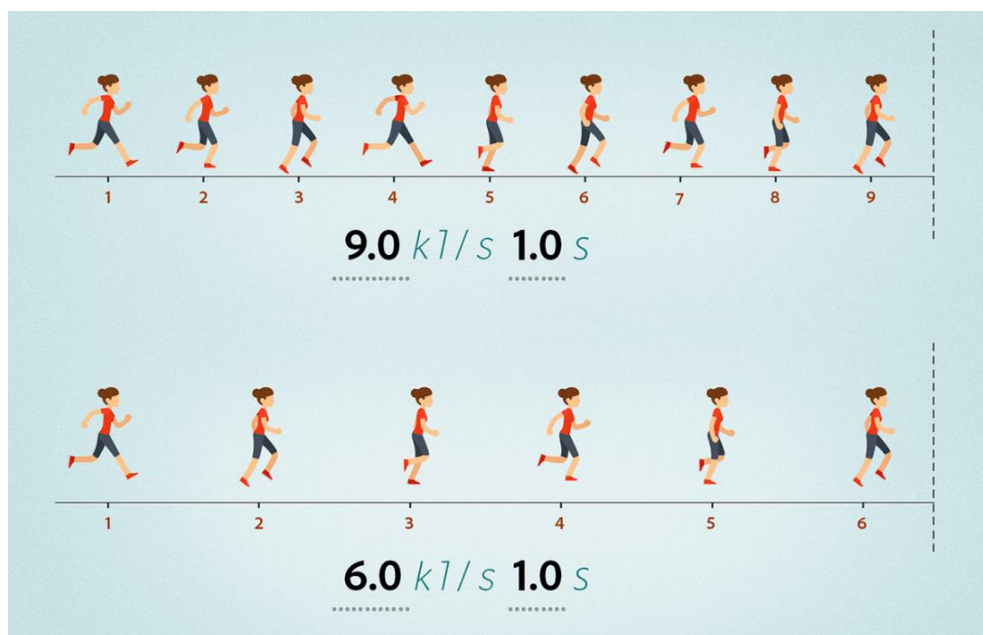
Rozdzielczość jest liczbą pikseli podawaną w odniesieniu do szerokości i wysokości pojedynczego obrazu. Rozdzielczość 1920 x 1080 oznacza obraz o szerokości 1920 pikseli i wysokości 1080 pikseli. Im wyższa rozdzielczość obrazu, tym większa jakość, która jest widoczna dla oka człowieka. W zależności od sprzętu, na jakim jest odtwarzany film, rozdzielczość obrazu może się różnić ze względu na istniejące standardy proporcji szerokości do wysokości obrazu [1]. Monitory komputerowe, telewizory, sprzęty mobilne czy przenośne posiadają różne rozdzielczości, ponieważ jest to zależne od ich różnych parametrów np. takich jak rozmiar ekranu.

Dla ludzkiego oka film odtwarzany z większą ilością klatek na sekundę będzie płynniejszy oraz będzie przypominał rzeczywistość (rysunek1). Taka płynność odtwarzania filmu jest bardzo ważna, gdy obraz wykorzystywany jest w różnych systemach i musi być to wykorzystane do sprawdzenia różnych informacji, szczególnie w przypadku, gdy wymagana jest wysoka precyzja.

Do przechwytywania obrazu dynamicznego, jakim jest film, wykorzystywane są różnego rodzaju kamery. Kamery mogą być oddzielnym sprzętem lub mogą być częścią innego urządzenia, jako urządzenie wbudowane. Kamery wideo mogą posiadać różnego rodzaju zastosowanie. Mogą być wykorzystywane w systemach bezpieczeństwa, do użytku domowego, hobbistycznie, w systemach wspomagających automatykę oraz mogą być wykorzystywane w przemyśle filmowym. Mają one możliwość zapisu filmu w pamięci zewnętrznej urządzenia lub bezpośrednio w jego pamięci wewnętrznej. Pamięć zewnętrzna może być zorganizowana w postaci karty pamięci, dysku lub pamięci USB.

Aktualne kamery zapisują obraz elektronicznie i są powszechne dzięki czemu można je znaleźć już w niemal wszystkich typach urządzeń mobilnych, takich jak

tablety, smartfony, kamery internetowe czy rejestratory samochodowe. Upowszechnienie takich urządzeń rejestrujących lub przekazujących obrazy wideo umożliwiło zwykłemu użytkownikowi na eksperymentowanie z różnymi przypadkami wykorzystującymi obraz dynamiczny.



**Rysunek 1.** Płynność obrazu podana w klatkach na sekundę [2]

Kamery bezpieczeństwa (kamery IP) wykorzystywane aktualnie służą do zapewnienia bezpieczeństwa pewnego obszaru (rysunek 2). Może to być obszar budynku mieszkalnego osoby prywatnej lub obszar całej organizacji. Takie kamery można podłączyć do sieci komputerowej tworząc zorganizowany system wideorejestracji, w ramach którego istnieje możliwość przechwytywania obrazu w wysokiej rozdzielczości i kontrolowania wydzielonego obszaru. Kamery mogą mieć zdalny dostęp dla użytkownika poprzez aplikację internetową czy aplikację mobilną na smartfonie i dzięki temu mogą być dowolnie sterowane i konfigurowane.



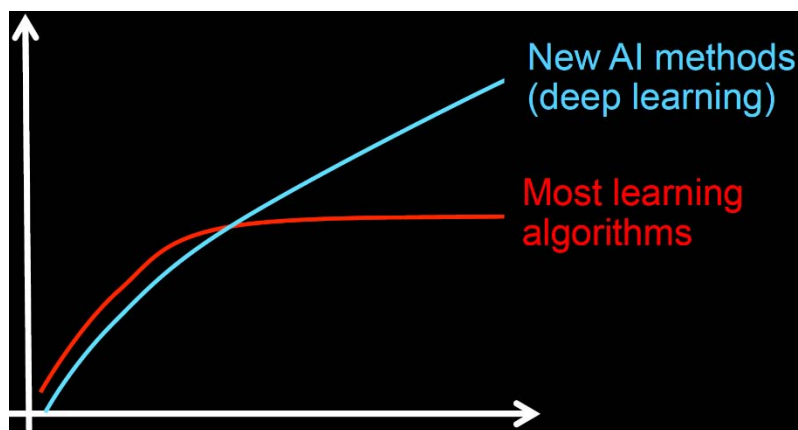
**Rysunek 2.** Prosty system rejestratora wideo wraz z kamerami IP [3]

Elementem wspomagającym przetwarzanie obrazów pochodzących z kamer są karty graficzne wykorzystywane w sprzęcie komputerowym. To one mają na celu wspomaganie odtwarzania obrazu wideo, wykonywania operacji sterowanych oprogramowaniem czy zwiększenie jakości otrzymywanych obrazów graficznych w grach komputerowych. Karty graficzne są oddzielną częścią sprzętu, który jest instalowany na płycie głównej komputera (w formie karty zintegrowanej lub niezintegrowanej – montowanej dodatkowo), stanowiąc element bardzo wydajny. Karta graficzna zintegrowana na płycie głównej posiada znacznie mniejszą wydajność i pobiera znaczną część mocy z centralnego procesora komputera. Karta graficzna jest wymaganą jednostką, która wspomaga niektóre operacje graficzne na komputerze, w tym bardzo silnie wspiera rozpoznawanie obiektów ruchomych.

## **2. Sposoby rozpoznawania obiektów ruchomych**

Rozpoznanie obiektu polega na znajdowaniu go na obrazie i sprawdzeniu czy znaleziony obiekt należy do grupy, która jest sprecyzowana przed rozpoczęciem etapu rozpoznawania. Takie techniki wykorzystywane są w systemach bezpieczeństwa, liniach produkcyjnych, na skrzyżowaniach, w samosterujących samochodach i innych. Taki

system musi być bardzo precyzyjny, aby nie popełniał błędów podczas rozpoznawania obiektów. Nie może rozpoznać danego obiektu jako obiektu innego. Będąc mało precyzyjnym może doprowadzić do sytuacji niebezpiecznych szczególnie wtedy, gdy wymagana jest dokładność bardzo wysoka, która jest zbliżona do możliwości ludzkiego oka.



**Rysunek 3.** Porównanie osiągnięć głębokiego uczenia ze starszymi algorytmami [4]

Do rozpoznawania obiektów ruchomych wykorzystywane są algorytmy bazujące na splotowej sieci neuronowej (ang. *Convolutional Neural Network*, CNN). Taka sieć składa się z warstw różnego rodzaju, które odpowiadają za rozpoznawanie kolorystyki, kształtów, rozmiarów, tekstury i innych cech obiektów, tak aby zgadzały się ze źródłem, które znajduje się w modelu danych. Głębokie uczenie odpowiada za wykorzystywanie sieci neuronowych do zadań uczenia się na podstawie wzorców i przykładów, ustawiania ich parametrów oraz do rozpoznawania obiektów na obrazach czy filmach wideo.

Uczenie sieci nie jest procesem liniowego programowania, lecz odwzorowuje myślenie człowieka, tak, aby rozpoznawanie obiektów było łatwiejsze i nie prowadziło do błędów. Głębokie uczenie jest początkiem automatyzowania procesów obsługi systemów sprawiając, że człowiek nie będzie musiał obsługiwać takich systemów, a systemy będą wykonywały operacje co najmniej na poziomie człowieka. Takie uczenie jest bardzo efektywne i może zostać szybciej zrealizowane (rysunek 3).

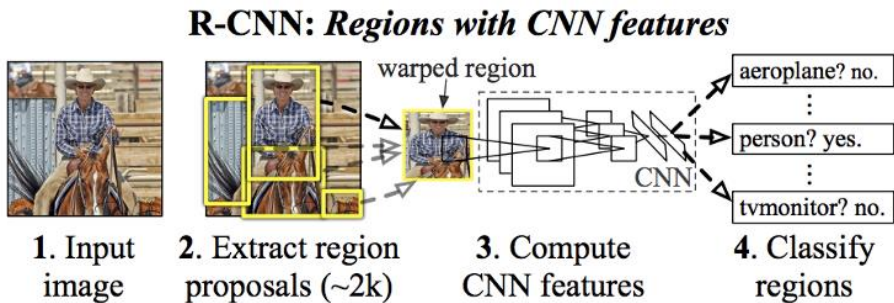
### 3. Pierwsze algorytmy w systemach rozpoznawania obiektów

Algorytmy powinny mierzyć się z rozpoznawaniem obiektów, które mogą posiadać różne proporcje, rozmiary, kolory oraz mogą być również w części ukryte na obrazie. Algorytmy, które zapoczątkowały erę bardzo wydajnego rozpoznawania obiektów pozwoliły na szybki rozwój algorytmów nowych. To one sprawiły, że użytkownicy zaczęli uczestniczyć w tworzeniu własnych algorytmów, w ten sposób uczestniczyć w wyzwaniach związanych z rozpoznawaniem obiektów.

Pierwszymi algorytmami posiadającymi wysoką doskonałość jest rodzina algorytmów R-CNN. Ta grupa składa się z algorytmu bazującego R-CNN oraz jego następców – Fast R-CNN i Faster R-CNN [5]. Skrót R-CNN można rozwinąć jako Regions with CNN (*Convolutional Neural Network*), czyli regiony ze splotową siecią neuronową. Następne algorytmy bazują na tym pierwszym i wprowadzają poprawki, które pozwalają na szybsze wykonywanie operacji, stąd dołączenie do nazwy słów „Fast” i „Faster”.

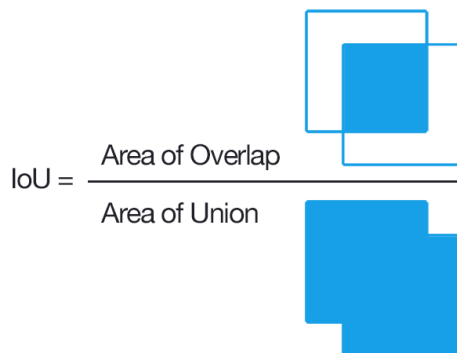
Algorytm wykorzystuje metodę nakładania siatki z podziałem na regiony, która umieszczana jest na obrazie wejściowym w celu rozpoznawania obiektów. Takich regionów jest 2000. Służą one do klasyfikacji obiektów i opisane są danymi przedstawiającymi poszczególne kategorie obiektów. Po rozpoznawaniu obiektów wyniki podawane są w procentach mAP (*mean Average Precision* – średnia precyzja). Taki algorytm w międzynarodowym konkursie PASCAL VOC (PASCAL Visual Object Classes) 2010 osiągał wynik 53,7% mAP. PASCAL VOC 2010 jest zestawem danych obrazów z roku 2010 wykorzystywanych pierwotnie podczas wspomnianego konkursu, w celu sprawdzenia ich możliwości. W późniejszym czasie zaczęto wykorzystywać go jako wzorzec w nowo budowanych algorytmach.

Zastosowanie algorytmu do rozpoznawania pojedynczego obiektu przy wykorzystaniu procesora centralnego CPU stacji komputerowej może zająć przykładowo 60 sekund, natomiast przy wykorzystaniu procesora graficznego GPU proces ten można skrócić nawet trzykrotnie. Rysunek 4 przedstawia etapy przetwarzania obrazu poprzez algorytm R-CNN. W pierwszym etapie przekazywany jest obraz, na który nakładana jest siatka proponowanych regionów (jest ich 2000), a następnie każdy wybrany obszar przechodzi przez warstwy sieci R-CNN i jest klasyfikowany jako obiekt z danej kategorii.



**Rysunek 4.** Struktura algorytmu R-CNN [5]

Części obiektów, które są sprawdzane w wyznaczonych regionach nałożonej siatki powinny wstępnie spełniać wartość IoU (*Intersection over Union* – część wspólna do całości, rysunek 5) wynoszącą 0.3. Wartość ta zwana inaczej współczynnikiem Jaccarda (*Jaccard overlap* [6]) jest stosunkiem części wspólnej pola obiektu wyznaczonego przez model i pola obiektu wyznaczonego przez zbiór danych testowych do sumy tych pól. Pole obiektu wyznaczonego przez model jest polem aktualnie wybranym do weryfikacji czy w danym miejscu znajduje się szukany obiekt.



**Rysunek 5.** Wartość IoU [7]

Następne algorytmy z tej rodziny wprowadzały różne modyfikacje, które umożliwiały osiągnięcie lepszych wyników precyzji oraz szybkości działania. Poprawa uczenia się

w tych algorytmach prowadzi do wzrostu ich szybkości, a wykorzystanie nowych warstw wspomagających algorytm pozwala na rozpoznawanie obiektów o wiele lepiej niż wersji poprzedniej. Fast R-CNN z poprawami ma możliwość osiągnięcia 68,4% mAP, a Faster R-CNN nawet 78,8% mAP ze znacznie lepszym czasem.

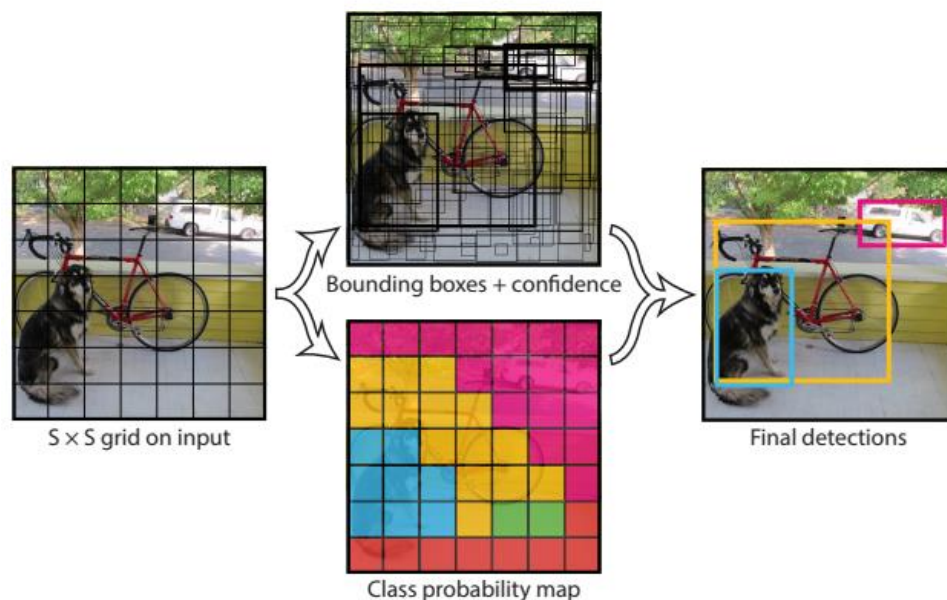
#### 4. Badanie wybranych algorytmów

W celu przeprowadzenia eksperymentu weryfikacji i oceny działania algorytmów rozpoznawania obiektów graficznych, do badania porównawczego zostały wybrane dwa przykładowe algorytmy: YOLO i SSD. YOLO (*You Only Look Once*) [8] jest algorytmem przetwarzającym obrazy dynamiczne w czasie rzeczywistym. Został on opracowany przez zespół ekspertów w roku 2015. Algorytm jest na bieżąco aktualizowany oraz udostępniane są jego nowe wersje posiadające nową, rozszerzoną funkcjonalność. Aktualnie wykorzystywana jest trzecia wersja tego algorytmu i wyróżnia się ona od poprzednich wersji silnie zmodyfikowaną architekturą budowy.

Dodatkowo należy zwrócić uwagę na algorytm SSD (Single Shot Multibox Detector, jednorzutowe rozpoznanie w różnych oknach) [6] [9]. Algorytm ten został opracowany w 2016 roku przez zespół kierowany przez Wei Liu [6]. Bazuje on na strukturze Caffe (struktura wykorzystywana do głębokiego uczenia).

Oba algorytmy częściowo bazują na R-CNN, ale wykorzystują inne techniki. Mają one możliwość osiągnięcia podobnych wyników w takich samych badaniach. YOLO wykonuje operacje nakładania obwiedni obiektów (prościej mówiąc czworokątnych obramowań obrazu, kształtu lub tekstu, które można przekształcać lub obracać) na obraz w celu klasyfikacji obiektów oraz przekazywania zgodności obiektu. Taki model posiada 24 warstwy spłotowe, które odpowiadają za różne czynności. Algorytm posiada wersję uproszczoną, która wykorzystuje 9 warstw sieci spłotowej zamiast 24 wykorzystywanych w wersji pełnej. To rozwiązanie pozwala na szybsze, lecz mniej dokładne przetwarzanie obrazów. Na obraz na wejściu nakładana jest siatka, która tworzy obwiednie, a następnie wykorzystuje je do rozpoznawania szukanych obiektów (rysunek 6). Algorytm ten bardzo szybko wykonuje swoje działania i ma możliwość przetwarzania filmów na żywo przez kamerę wideo. Podczas wykonanych testów algorytm osiągał 63.4% mAP z prędkością 45 kl/s, co jest bardzo wysokim wynikiem.



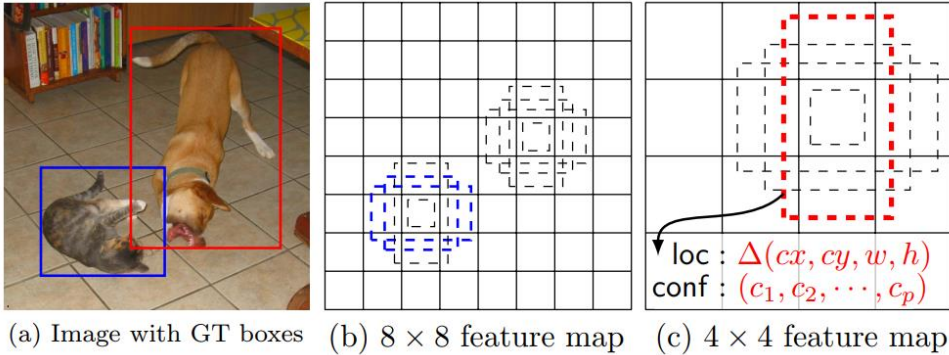


**Rysunek 6.** Podział obrazu na mapę prawdopodobieństwa klas oraz mapę obwiedni wraz z wynikiem końcowym [8]

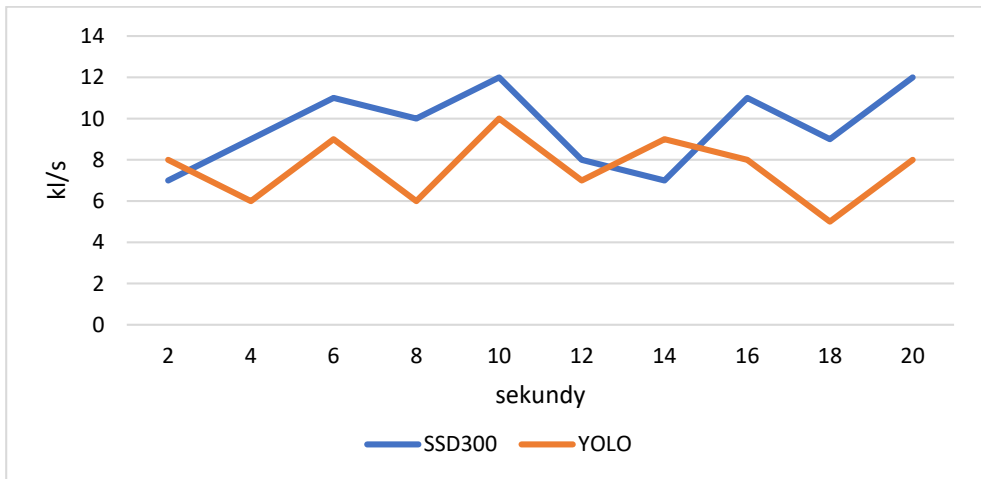
Następnym przeanalizowanym algorytmem jest Single Shot Multibox Detector (SSD). Algorytm ten także ma możliwość osiągnięcia wysokiej precyzji wraz z utrzymaniem szybkości. Wykorzystuje on obwiednie o różnych rozmiarach celem rozpoznawania obiektów różnego rodzaju na dowolnych obrazach. Dzięki wykorzystaniu takiej techniki możliwe jest rozpoznawanie obiektów dużych oraz małych na tym samym obrazie. Algorytm również wykorzystuje różnorodne rozmiary oraz kształty obwiedni, które zostały ustawione jako domyślne podczas korzystania z algorytmu. Ostatni etap rozpoznawania obiektów składa się z technik udoskonalających dane wyjściowe wraz z wykorzystaniem obwiedni, które zostały oznaczone wcześniej jako błędne.

SSD posiada dwie wersje algorytmu: SSD300 i SSD512. Różnią się one rozdzielczością obrazu wejściowego. SSD300 osiąga w testach 79,6% mAP zaś SSD512 osiąga 81,6% mAP. Rysunek 7 (a) przedstawia GT (*ground-truth* – prawdziwe, realne) obszary, na których znalezione zostały dwa obiekty będące zwierzętami (pies

i kot), (b) przedstawia podział na mapę  $8 \times 8$  oraz znalezienie obiektu mniejszego, (c) przedstawia współrzędne lokalizacji (loc) oraz wartości przedstawiające zaufanie istnienia obiektu w tym miejscu na podstawie porównywanych cech (conf).



Rysunek 7. Różne rozmiary obwiedni [9]

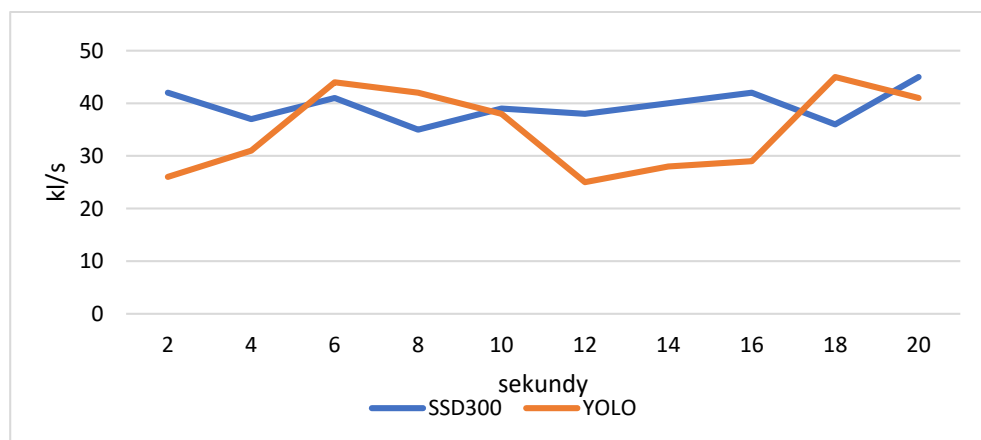


Rysunek 8. Wykres badania 1 – szybkość przetwarzania obrazu przy wykorzystaniu CPU

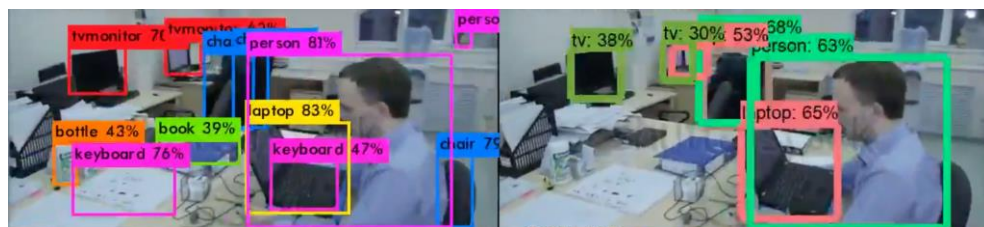
Algorytmy zostały wykorzystane do testów w dwóch badaniach, które polegały na porównaniu ich dokładności oraz szybkości działania. Testy przeprowadzone zostały

na sprzęcie posiadającym kartę graficzną Gigabyte RTX 2060 oraz procesor Intel i5 6500.

Pierwsze badanie dotyczyło rozpoznania samochodu na żywo. Zbliżając się powoli do zaparkowanego pojazdu użyto obydwu algorytmów, aby sprawdzić, który z nich wcześniej rozpozna, że obiekt jest pojazdem. YOLO rozpoznał pojazd 20 metrów wcześniej niż SSD. Oba algorytmy osiągały podobne wyniki wykorzystując CPU czy GPU. SSD300 osiągał na CPU 7-12 kl/s, a YOLO 5-10 kl/s (rysunek 8). Na GPU SSD osiągał 30-45 kl/s, a YOLO 25-45 kl/s (rysunek 9).



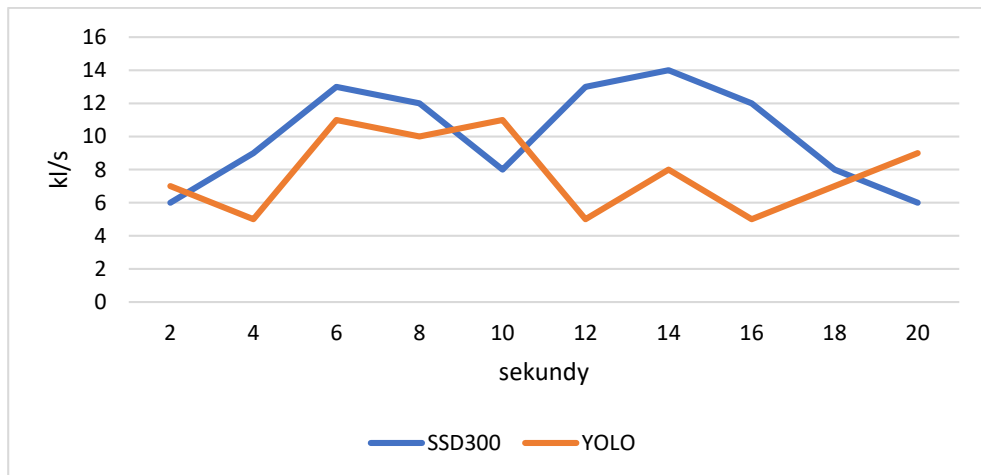
**Rysunek 9.** Wykres badania 1 – szybkość przetwarzania obrazu przy wykorzystaniu GPU



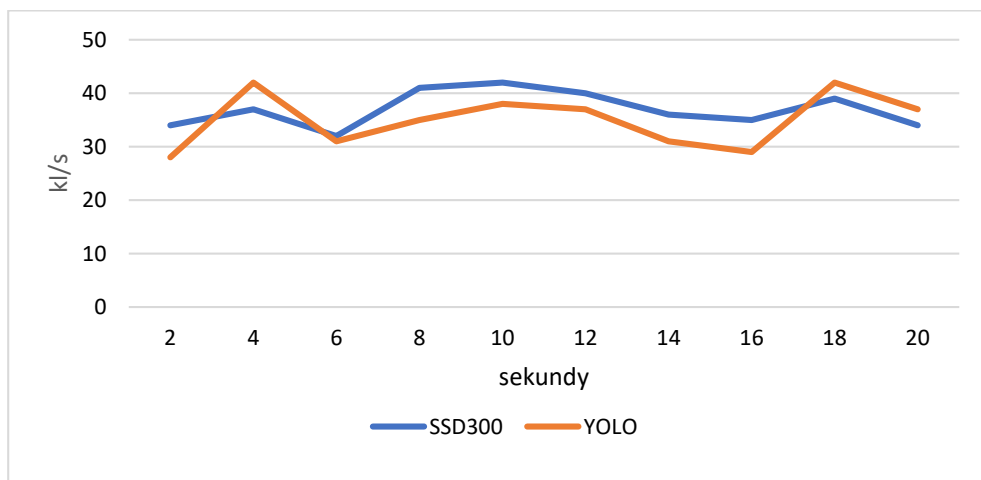
**Rysunek 10.** Test porównania YOLO i SSD w rozpoznaniu przedmiotów biurowych

W ramach drugiego badania przeprowadzono test rozpoznawania obiektów na próbce przykładowego klipu wideo pobranego z sieci internetowej. Klip wideo

przedstawił osobę pracującą przy biurku, na którym znajdował się sprzęt biurowy różnego rodzaju (rysunek 10).



**Rysunek 11.** Wykres badania 2 – szybkość przetwarzania obrazu przy wykorzystaniu CPU



**Rysunek 12.** Wykres badania 2 – szybkość przetwarzania obrazu przy wykorzystaniu GPU

Przeprowadzony eksperyment rozpoznawania obiektów umożliwił rozpoznanie laptopa przez model algorytmu YOLO w 83%, a przez SSD w 65%. Tak małe obiekty jednak nie zawsze są dokładnie rozpoznawane przez algorytmy, ponieważ ze względu na ich kształty oraz ułożenie na obrazie obiekty te mogą przypominać inne obiekty. Algorytm SSD300 przetwarzał na CPU 6-14 kl/s, a YOLO 5-11 kl/s (rysunek 11). Na GPU algorytm SSD przetwarzał 34-42 kl/s, a YOLO 28-42 kl/s (rysunek 12).

## **5. Koncepcja nowego modelu rozpoznawania obiektów ruchomych**

W efekcie porównania opisanych powyżej modeli podjęto próbę zaprojektowania nowego uniwersalnego modelu przetwarzania ruchomych obrazów graficznych. Koncepcja tego modelu zakłada dążenie do osiągnięcia lepszych wyników działania w stosunku do aktualnie dostępnych modeli. Proponowany model powinien cechować się łatwością korzystania z rozwiązań oraz pozwalać na ich szybszy rozwój, ciągłe udoskonalanie oraz zwiększanie wydajności.

Proponowany model bazować powinien przynajmniej na obu przeanalizowanych rozwiązaniach YOLO i SSD. Ponadto powinien wykorzystywać część funkcjonalności przeanalizowanych algorytmów oraz wprowadzać nowe, które mogą sprawić, że algorytm może być dokładniejszy, szybszy, łatwiejszy w użyciu, bardziej powszechny, częściej używany oraz łatwiejszy do rozbudowy. Algorytm powinien posiadać następujące modyfikacje:

- hybrydowe rozwiązanie wykorzystujące pracę modułów GPU i CPU,
- wykorzystanie rozdzielczości 600 x 600 obrazu wejściowego,
- wykorzystanie różnych rozmiarów siatek od 12 x 12 do 60 x 60,
- wykorzystanie nowego modelu danych dzielonego na kategorie obiektowe,
- podział dostępu do modelu danych na użytkowników zwykłych, zaufanych oraz administratorów,
- podział sieci na warstwy zależne od rozmiaru siatki,
- wykorzystanie aplikacji z interfejsem graficznym,
- wykorzystanie aplikacji mobilnej umożliwiającej używanie algorytmu na urządzeniach mobilnych.

Hybrydowe rozwiązanie GPU i CPU umożliwić może podział zadań na oba moduły co sprawi, że ulegnie zwiększeniu szybkość wykonywanych operacji. Proporcjonalnie, gdy GPU będzie wykonywało operacje na  $\frac{3}{4}$  części obrazu, CPU będzie wykonywało operacje na  $\frac{1}{4}$  części obrazu. W zależności od wydajności wykorzystywanego sprzętu wynik może się poprawić od 15% (jeśli GPU jest bardzo wydajna) do 30% (jeśli CPU jest bardzo wydajny). Takie rozwiązanie jest wykorzystywane już w innych systemach (np. programy do modelowania 3D), lecz nigdy w systemach rozpoznawania obiektów.

Rozdzielczość 600 x 600 na wejściu powinna umożliwić podział na siatki wielokrotności 6. Dodatkowo wykorzystywane mogłyby być rozmiary siatek od 12 x 12 do 60 x 60. Na siatki nakładane powinny być obwiednie, które przeprowadzą proces rozpoznawania obiektów. Obwiednie wówczas posiadać będą różne rozmiary oraz proporcje, które umożliwią rozpoznawanie większości obiektów. Siatka 12 x 12 umożliwi rozpoznawanie dużych obiektów a 60 x 60 małych obiektów. Taka siatka mogłaby zostać wybrana przez użytkownika w ustawieniach. Każda komórka siatki wykorzystuje 3 obwiednie w różnych rozmiarach. Przykładowo, gdy użytkownik zdecyduje się na siatkę w rozmiarze 12 x 12 otrzyma wtedy 432 (144 x 3) obwiednie, które mają na celu rozpoznawanie dużych obiektów.

Algorytm może również wykorzystać nowy interfejs graficzny dla wersji desktopowej oraz mobilnej, który powinien umożliwić wykonywanie operacji z dużą szybkością. Taki interfejs byłby ułatwieniem dla nowych użytkowników chociażby z tego powodu, że nie wymagałby on od użytkowników konieczności poznawania komend wykorzystywanych w algorytmach.

Architektura modelu bazuje na SSD, ponieważ aktualnie jest on najbardziej wydajny, lecz dodatkowo wprowadza warstwy, które są wykorzystywane w rozwiązaniu wyboru rozmiaru siatek przez użytkownika. Takie rozwiązanie ma możliwość zmiany modelu na bardzo elastyczny, dzięki czemu może być w pełni wykorzystany lub jedynie w małej części.

Proponowany model powinien umożliwiać wykorzystywanie danych w zależności od sytuacji, ponieważ użytkownik sam może wybrać jakie dane go interesują, czy to będzie pojazd czy inny obiekt ruchomy. Udostępniony użytkownikom model danych może zostać pobrany w pełni lub jedynie w części, co sprawia, że sam model może być bardzo lekki i wykonywać operacje szybko lub może być pełny i dokładny,

lecz wykonywać operacje wolniej ze względu na ilość zamieszczonych danych obrazów z każdej wybranej kategorii obiektów. Analizę porównawczą koncepcji zaproponowanego modelu z modelami tradycyjnymi przedstawiono w tabeli 1.

**Tabela 1.** Porównanie funkcji koncepcji nowego modelu z modelami tradycyjnymi

<b>Funkcje</b>	<b>Porównanie</b>
Architektura modelu sieci	Bazuje na SSD. Wykorzystuje architekturę VGG-16 (sieć neuronowa wykorzystywana w algorytmach rozpoznawania obiektów) oraz dodatkowo warstwy zależne od używanych siatek, które wybiera użytkownik.
Model danych	Nowe rozwiązanie. Zakres modelu danych jest wybierany przez użytkownika.
Rozdzielczość obrazu wejściowego	Większa od YOLO i SSD
Obwiednie na siatce	Bazuje częściowo na SSD i YOLO
Rozmiary i proporcje obwiedni	Nowe rozwiązanie. W pełni modyfikowane przez użytkownika na podstawie własnych wymagań.
Podział sieci na warstwy rozmiarów siatek	Nowe rozwiązanie. YOLO i SSD wykorzystuje w swoich sieciach wszystkie warstwy.
Aplikacja z interfejsem graficznym	Nowe rozwiązanie. YOLO i SSD nie posiadają takiej funkcji.
Aplikacja mobilna	Nowe rozwiązanie. YOLO i SSD nie posiadają takiej funkcji.
W pełni modyfikowalny algorytm	YOLO i SSD nie umożliwiają tak bardzo rozwiniętej konfiguracji modelu, która pozwala na dobór ustawień w taki sposób, aby algorytm pasował do każdego użytkownika.
Hybrydowe rozwiązanie GPU i CPU	Nowe rozwiązanie. YOLO i SSD wykorzystują albo GPU albo CPU.

## 6. Podsumowanie

W artykule przedstawiono podstawowe pojęcia i zasady systemu rozpoznawania obiektów ruchomych, a także przedstawiono koncepcję nowej wersji takiego systemu. Ta nowa wersja miałaby większe możliwości niż systemy dotychczas używane. Zaproponowane rozwiązania mogłyby pozwolić na efektywniejsze rozpoznawanie obiektów ruchomych. Skuteczność rozpoznawania stałaby się porównywalna

ze skutecznością oka ludzkiego, a nawet byłaby większa. W zależności od sytuacji oraz typu rozpoznawanego obiektu pozwalałaby ona na szeroki wybór możliwości zmiany ustawień systemu, dzięki czemu mogłaby zyskać popularność oraz rozszerzyć grono użytkowników. Z kolei obserwowany wzrost wydajności kart graficznych oraz mocy obliczeniowej procesorów – w szczególności tych wielordzeniowych – nie jest również bez znaczenia. Obserwowany tu rozwój może także znacznie przyczynić się do procesu rozpowszechnienia tego typu koncepcji.

Nie ma technicznych przeszkód, aby algorytmy bazujące na tego typu rozwiązaniach były rozwijane i stawały się coraz popularniejsze. Dlatego także przedstawiona w pracy koncepcja powinna zostać zaimplementowana i przetestowana. Niestety, jak do tej pory to się nie udało: to nadal jest to tylko koncepcja. Ze względu na złożoność istniejących algorytmów, ograniczenia sprzętowe oraz znaczne wymagania wobec członków zespołu, który mógłby dalej zajmować się rozwijaniem przedstawionej koncepcji, a także ze względu na nierozpoznaną społeczność potencjalnie zainteresowaną rozwijaniem systemu, nie udało się autorom przeprowadzenie weryfikacji przedstawionej tu koncepcji. Mimo tego autorzy mają nadzieje, że z biegiem czasu wszystkie trudności będą mogły zostać pokonane i zaprezentowana koncepcja będzie mogła zostać zrealizowana. Do czego – być może – przyczyni się też niniejszy artykuł.

## **Bibliografia**

- [1] *Rozdzielczość*, kuzniewski.pl, <https://www.kuzniewski.pl/i37,rozdzielczosc.html> [06.01.2019].
- [2] *Czas*, Adobe Animate, <https://helpx.adobe.com/pl/animate/using/time.html> [09.01.2019].
- [3] *Network video recorder*, Wikipedia, [https://en.wikipedia.org/wiki/Network\\_video\\_recorder](https://en.wikipedia.org/wiki/Network_video_recorder) [14.01.2019].
- [4] A. Ng, *Deep Learning*, <http://cs229.stanford.edu/materials/CS229-DeepLearning.pdf> [28.11.2019].
- [5] R Gandhi, *R-CNN, Fast R-CNN, Faster R-CNN, YOLO – Object Detection Algorithms*, Towards Data Science,



- <https://towardsdatascience.com/r-cnn-fast-r-cnn-faster-r-cnn-yolo-object-detection-algorithms-36d53571365e> [18.03.2019].
- [6] W. Liu, D. Anguelov, D. Erhan, Ch. Szegedy, S. Reed4, Cheng-Yang Fu, A.C. Berg, *SSD: Single Shot MultiBox Detector*, <https://arxiv.org/pdf/1512.02325.pdf> [20.05.2019].
- [7] A. Rosebrock, *Intersection over Union (IoU) for object detection*, <https://www.pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/> [19.03.2019].
- [8] J. Redmon, S. Divvalay, R. Girshick, A. Farhadiy, *You Only Look Once*, <https://arxiv.org/pdf/1506.02640v5.pdf> [06.05.2019].
- [9] E. Forson, *Understanding SSD MultiBox – Real-Time Object Detection in Deep Learning*, Towards Data Science, <https://towardsdatascience.com/understanding-ssd-multibox-real-time-object-detection-in-deep-learning-495ef744fab> [05.06.2019].
- 

## **Graphic recognition information system of moving objects**

### **Abstract**

The system of graphic recognition of moving objects is a solution that can be very widely used in everyday life. Such a system should recognize objects that are specified by the user before launching it. The algorithms that are used in this system are expected to have high speed and accuracy. There are many solutions for such algorithms that use many different techniques and settings. This article presents the basic concepts and principles of the moving object recognition system, and also presents the concept of a new graphic system for recognizing moving objects that has greater capabilities than the systems used so far.

**Keywords:** *graphics, object recognition, recognition of moving objects*