

Automatyczna generacja wirtualnych elementów infrastruktury kolejowej

Paweł Lisiecki, Maciej Szłapczyński

Autocomp Management Sp. z o.o., ul. 1 Maja 36, 71-627 Szczecin

Ewelina Chołodowicz

Szkoła Doktorska w Zachodniopomorskim Uniwersytecie Technologicznym w Szczecinie, al. Piastów 19, 70-310 Szczecin

Streszczenie: Niniejszy artykuł poświęcono zagadnieniom modelowania wirtualnego świata 3D na potrzeby symulatorów kolejowych oraz problematyce tworzenia mapy. Zaproponowano algorytm wykorzystujący metody sztucznej inteligencji do wykrywania, klasyfikacji i umieszczania obiektów infrastruktury kolejowej z nagrania wideo oraz danych GPS w wirtualnym świecie 3D. Proponowane rozwiązanie, wspierające automatyczną generację wirtualnych elementów infrastruktury kolejowej, stanowi istotną nowość w obszarze badań.

Słowa kluczowe: symulator ruchu drogowego, symulator kolejowy, konwolucyjna sieć neuronowa, detekcja obiektów, generacja świata 3D, GPS, GIS, OSM

1. Wprowadzenie. Problemy częściowe w literaturze problemu

W pracy [1] przedstawiono wybrane symulatory ruchu drogowego oraz zarysowano problemy związane z ich wykorzystaniem w praktyce. W pracach [2–6] przedstawiono w aplikacjach praktyczne aspekty wykorzystania danych GIS do generowania świata wirtualnego, nie tylko na potrzeby symulatorów. W pracach [4, 7] przedstawiono modele cyfrowe bazujące na GIS wraz z przepływami pracy nad tymi modelami. W artykule [8] zaproponowano algorytm automatycznego generowania świata 3D na podstawie danych GIS. W pracach [5, 9] źródłem danych do trójwymiarowego modelu jest również OSM (ang. *OpenStreetMap*). Autorzy prac [10–12] sięgają po dane z OSM w celu generowania ścieżek ruchu autonomicznego. System przeszukiwania i korekty połączeń w wygenerowanych na podstawie OSM ścieżkach jest przedmiotem pracy [12]. Prezentacja danych zawartych w systemach GIS jest przedmiotem prac [13–16]. Z perspektywy wizualizacji są to duże lub bardzo duże zbiory danych i wymagają stosowania narzędzi znanych na co dzień z gier komputerowych [7]. Problem badawczy stanowiący przedmiot niniejszego artykułu to opracowane w ramach projektu narzędzia do wykrywania, klasyfikacji i umieszczania obiektów infrastruktury kolejowej z nagrania wideo w wirtualnym świecie 3D. Algorytm detekcji infrastruktury kolejowej został omówiony m.in. w pracy [17]. Praca przedstawia implementację hybrydowego algorytmu

detekcji oraz rozpoznawania semaforów kolejowych na trasie przejazdu pociągu. W ramach badań przeprowadzono testy symulacyjne, w których szukano optymalnego rozwiązania dla zadania detekcji i klasyfikacji obiektów infrastruktury kolejowej. Wynikiem badań jest hybrydowy algorytm, który wykorzystuje konwolucyjną sieć neuronową YOLO do wykrywania semaforów na obrazach oraz klasyczne metody przetwarzania obrazu do klasyfikacji sygnalizacji na wykrytym semaforze. Problematykę oraz rozwiązania w zakresie detekcji obiektów infrastruktury kolejowej przedstawiono również w [18–20].

W kolejnej części zarysowano problemy, z jakimi należy się zmierzyć podejmując wyzwanie opracowania przepływu pracy oraz narzędzia informatycznego, wspierającego generację obiektów infrastruktury kolejowej na potrzeby symulatora.

2. Problemy tworzenia świata 3D o wysokim poziomie szczegółowości

Tworząc cyfrowe mapy na potrzeby symulatora kolejowego należy zwrócić uwagę na kilka kluczowych zagadnień. Dostępne rozwiązania zapewniające detekcję i klasyfikację obiektów nie osadzają obiektów w świecie wirtualnym – rozwiązują tylko problem detekcji i klasyfikacji [17–19]. Błędy oraz niedokładności w ogólnie dostępnych danych (np. *OpenStreetMap*) to kolejne wyzwanie podczas tworzenia map na potrzeby symulatorów [4, 12]. Rozwiązania przeznaczone do generowania wirtualnych światów 3D pozwalają na automatyczne generowanie świata na podstawie danych zewnętrznych (np. *OpenStreetMap*), jednakże generacja ma często charakter generyczny lub uproszczony, w związku z czym tak wygenerowane dane nie mogą zostać użyte jako reprezentacja wirtualnego świata dla symulatorów [6].

Modelowanie wirtualnego świata 3D na potrzeby symulatorów kolejowych jest bardzo czasochłonnym procesem. W pracy [10] w celu zwiększenia efektywności generowania świata 3D ograniczono jego szczegóły. Z tego powodu nie jest możliwe

Autor korespondujący:

Paweł Lisiecki, plisiecki@autocomp.com.pl

Artykuł recenzowany

nadesłany 22.05.2024 r., przyjęty do druku 11.06.2024 r.



Zezwala się na korzystanie z artykułu na warunkach licencji Creative Commons Uznanie autorstwa 3.0

wykorzystanie takiego rozwiązania do zastosowań symulatorów, które z reguły wymagają wysokiego poziomu immersyjności. Oprogramowanie, które jest dedykowane dla generowania świata 3D o wysokim stopniu odwzorowania ma zwykle problemy z generacją charakterystycznych budynków, które swoim kształtem odbiegają od standardowej zabudowy. Dodatkowo, eksportowane dane są zapisywane w formie warstw tylko do odczytu. Warstwy takie mogą być przetwarzane jedynie w dedykowanym oprogramowaniu [14], co znacznie utrudnia proces korekcji charakterystycznych budynków w modelowanym świecie.

3. Tworzenie wirtualnego świata 3D na potrzeby symulacji

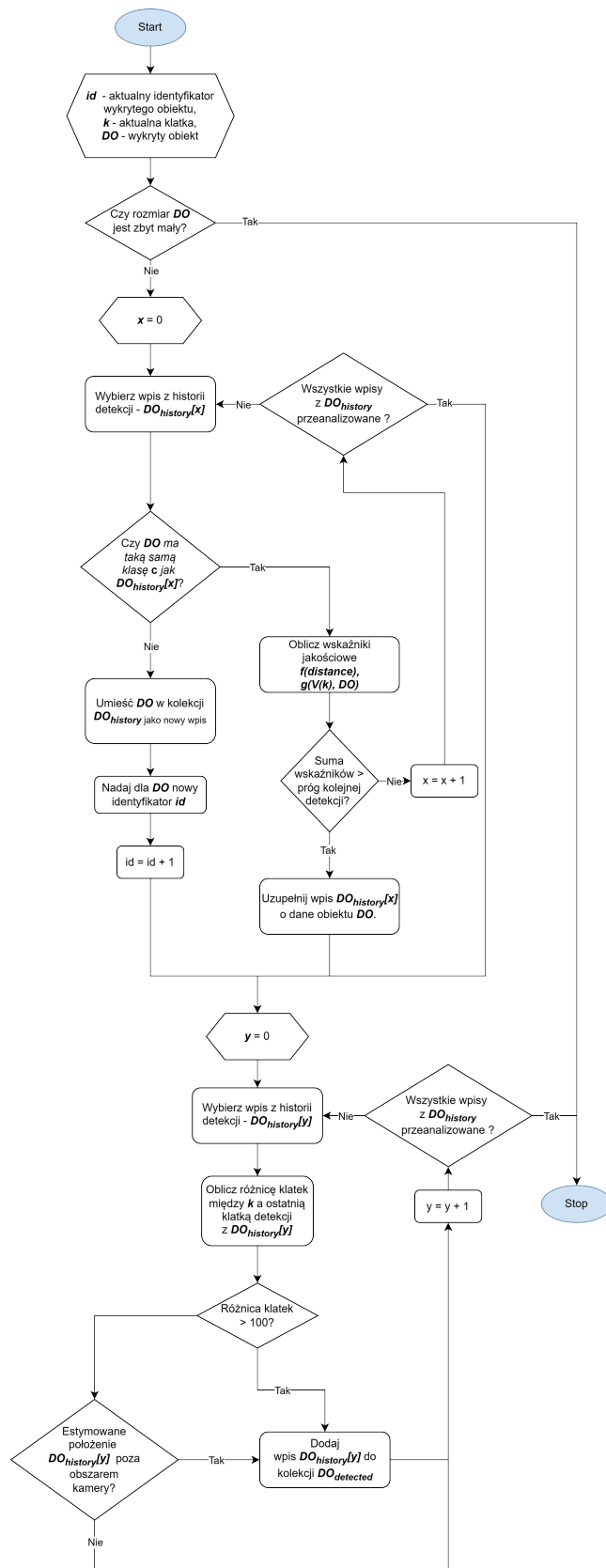
Tworzenie wirtualnego świata 3D na potrzeby symulatora to złożony i czasochłonny proces, który składa się z wielu etapów. W artykule skupiono się przede wszystkim na problematyce tworzenia wirtualnego świata 3D dla symulatorów kolejowych. Proces tworzenia map dla symulatora rozpoczyna się od gromadzenia danych na temat modelowanej linii kolejowej. W tym celu przygotowywane są nagrania z kamery umieszczonej w kabinie maszynisty, która jest skierowana na przednią szybę pociągu. Kamera rejestruje przejazd pociągu na danej linii kolejowej z perspektywy maszynisty. Następnie w analogiczny sposób realizowany jest przejazd w przeciwnym kierunku. Zebrane nagrania stanowią podstawowe źródło danych dla projektantów wirtualnego świata 3D.

W związku z tym, że kamera rejestruje obraz z perspektywy maszynisty, to na podstawie nagrania, projektant jest w stanie stwierdzić, które elementy infrastruktury kolejowej oraz otoczenia są istotne oraz w jakim stopniu należy je odwzorować (osoba korzystająca z symulatora musi mieć zapewnione takie same pole widzenia, jak maszynista w rzeczywistym pojeździe). Kolejnym etapem jest przygotowanie geograficznych danych referencyjnych, na podstawie których projektant będzie modelował przebieg linii kolejowej. W tym celu wykorzystywane są dane OpenStreetMap. Kolejnym etapem jest tworzenie wirtualnego świata 3D przy pomocy graficznego edytora. Projektant wykorzystuje dane z nagrania oraz dane geograficzne do modelowania świata 3D w edytorze map. Stworzona mapa jest następnie weryfikowana w środowisku testowym pod kątem zgodności z materiałem wideo. W przypadku błędów lub nieprawidłowości, mapa jest modyfikowana i ponownie testowana. Ostatnim etapem jest eksport mapy do formatu akceptowanego przez symulator kolejowy oraz integracja stworzonej mapy z systemem symulacji.

W całym procesie, najbardziej czasochłonnym etapem jest etap modelowania świata 3D. Wynika to głównie z powodu, że jest to proces manualny, który dodatkowo wymaga ciągłej weryfikacji modelowanego obszaru z danymi wideo oraz danymi geograficznymi. Na szybkość wykonywania świata 3D wpływa również jakość nagrania, ponieważ to ono jest głównym źródłem danych dla projektanta. Ogólnodostępne dane geograficzne nie zawierają informacji o rozmieszczeniu znaków kolejowych, czy semaforów.

W ramach realizacji projektu edytora stworzono moduł sprzętowo-programowy do automatycznego wykrywania, klasyfikacji oraz umieszczania w odpowiednim systemie referencyjnym koordynatów (CRS) elementów infrastruktury kolejowej. Przy pomocy konwolucyjnej sieci neuronowej, na podstawie danych wideo dokonywana jest detekcja i klasyfikacja elementów infrastruktury kolejowej. Każdy wykryty obiekt jest klasyfikowany a następnie śledzony, do czasu, gdy dany obiekt zniknie z pola widzenia (znajdzie się poza obszarem rejestrowanym przez kamerę), lub upły-

nie pewna ilość klatek w których obiekt nie był już wykrywany przez sieć neuronową. Dane o detekcjach są zestawiane z danymi GPS we wspólnej bazie czasowej. Dzięki takiemu rozwiązaniu możliwe jest umieszczenie wykrytego obiektu



w odpowiednim miejscu w wirtualnym świecie. Takie rozwiązanie pozwala na automatyzację procesu wykrywania znaków infrastruktury kolejowej przez projektanta. W stworzonym oprogramowaniu Hermes [21] użytkownik ma możliwość podglądu nagrania wideo wraz z zaznaczonymi detekcjami obiektów oraz pozycji kamery na scenie 2D (rys. 3). Dodatkowo możliwy jest podgląd detekcji na osi czasu, dzięki czemu możliwa jest nawigacja po czasie w celu przejścia do danej detekcji obiektu. Dzięki temu, projektant może efektywnie wprowadzać elementy do wirtualnego świata, ponieważ nie musi oglądać całego nagrania, a jedynie nawigować między kolejnymi detekcjami.

4. Detekcja, klasyfikacja i lokalizacja obiektów infrastruktury kolejowej



Rys. 2. Wyniki działania rozwiązania dla detekcji, klasyfikacji i śledzenia obiektów
Fig. 2. Results of the solution for object detection, classification and tracking

Proces detekcji i klasyfikacji obiektów przebiega w sposób standardowy dla konwolucyjnej sieci neuronowej. Proces uczenia sieci został opisany krok po kroku w pracy [17]. To, co wyróżnia prezentowane w artykule rozwiązanie od klasycznego procesu detekcji i klasyfikacji obiektu, to śledzenie znalezionej detekcji oraz jego lokalizacja według danych zarejestrowanych przez GPS o wysokiej rozdzielczości pomiaru. Konwolucyjna sieć neuronowa była uczona na zbiorze danych składającym się z ponad 25 000 zdjęć (pochodzących z nagrań z przejazdów) na których oznaczono obiekty należące do ponad 40 różnych klas obiektów infrastruktury kolejowej, do których należą: semafor, tabliczka kilometrażowa, pas świetlny, wskaźniki W1, W4, W5 i inne.

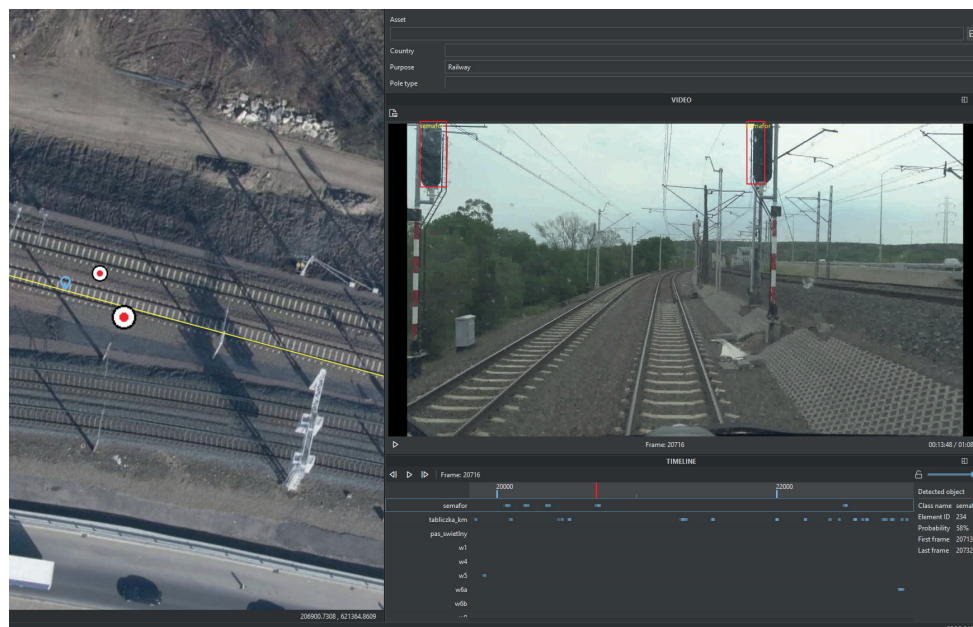
Proces śledzenia obiektu (rys. 1) przebiega w następujący sposób: dla każdej klatki nagrania (k) pobierane są dane o wszystkich obiektach, które w danej klatce zostały wykryte. Pojedynczy wpis o wykrytym obiekcie (DO) zawiera informacje o rozmiarze obszaru wykrytego obiektu na nagraniu (w pikselach), o klasie obiektu (C) oraz o prawdopodobieństwie (w procentach) przynależności wykrytego obiektu do danej klasy (C). Przykładowe wyniki detekcji zostały przedstawione na rys. 2.

Dla każdego obiektu (DO) wykrytego w danej klatce (k) nagrania wykonywane są następujące czynności:

1. Wstępna selekcja detekcji na podstawie wymiarów wykrytego obiektu;
2. Sprawdzanie, czy obiekt o danej klasie (C) nie został wcześniej wykryty;
3. Jeżeli obiekt nie został wcześniej wykryty, to DO jest umieszczany w kolejce już wykrytych obiektów $DO_{history}$ i nadawany jest mu kolejny identyfikator dla danej klasy obiektu (id), a także zapisywana jest aktualna klatka k ;
4. Jeżeli obiekt klasy C został wcześniej wykryty, to jest sprawdzane, czy którykolwiek z wcześniej wykrytych obiektów z $DO_{history}$ ma taką samą klasę jak DO ;

5. Jeżeli taki obiekt zostanie znaleziony, to obliczane są wskaźniki jakościowe, które określają, czy dana detekcja DO jest kolejną detekcją wykrytego już wcześniej obiektu;
6. W pierwszej kolejności obliczany jest wskaźnik $f(distance)$, gdzie $distance$ to odległość między DO a danym obiektem ($DO_{history}[x]$) z historii;
7. Następnie wyliczany jest wskaźnik $g(V(k), DO)$, gdzie $V(k)$ to estymowany wektor prędkości ruchu $DO_{history}[x]$ wyliczany na podstawie ostatnich dwóch detekcji. Wskaźnik f określa, czy nowa detekcja DO znajduje się w pobliżu poprzedniej detekcji, natomiast wskaźnik g określa, czy nowa detekcja DO znajduje się w pobliżu nowej (potencjalnej) pozycji $DO_{history}[x]$;
8. Następnie oba wskaźniki są sumowane. Jeżeli suma jest większa od pewnej przyjętej wartości parametru, to dana detekcja DO jest traktowana jako kolejna detekcja obiektu $DO_{history}[x]$ i do historii detekcji $DO_{history}[x]$ wpisywane jest wykryte położenie DO . Dodatkowo obliczany jest wektor prędkości ($V(k)$) $DO_{history}[x]$ na podstawie $distance$ oraz różnicy między klatką k a klatką ostatniej detekcji $DO_{history}[x]$;
9. Dla wszystkich obiektów w $DO_{history}$ dla danej klatki k wykonywany jest następujący algorytm: (1) Obliczana jest różnica między klatką k a ostatnią klatką detekcji $DO_{history}[x]$; (2) Obliczana jest estymowane położenie obiektu $DO_{history}[x]$; Jeżeli z obliczeń wyniknie, że obiekt znajdzie się poza obszarem rejestrowanym przez kamerę, to dany obiekt $DO_{history}[x]$ jest przenoszony do listy wykrytych i zakończonych detekcji $DO_{detected}$.

Może się jednak zdarzyć, że w historii detekcji danego obiektu będzie tylko jeden wpis, w związku z tym nie będzie obliczony wektor $V(k)$ i nie będzie możliwa estymacja położenia, co spowoduje, że taki obiekt nigdy nie zostanie przeniesiony do listy już wykrytych obiektów. Aby uniknąć takiej



Rys. 3. Podgląd wyników danych w oprogramowaniu Hermes

Fig. 3. Preview the resulting data in Hermes software

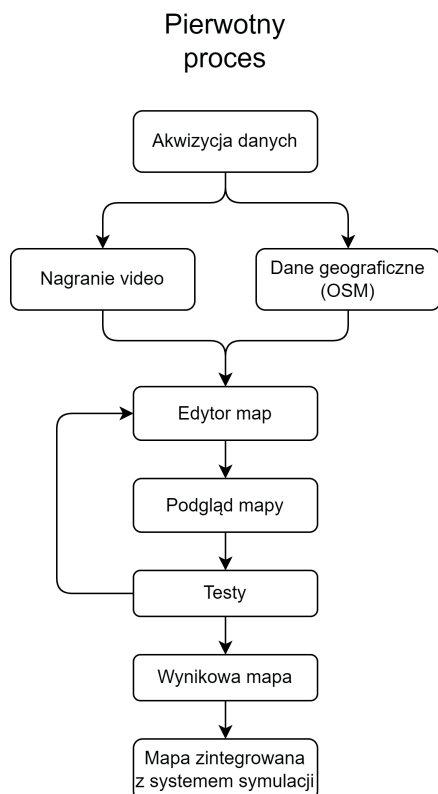
sytuacji, przyjęto, że obiekt zawsze jest przenoszony do listy już wykrytych obiektów, kiedy od ostatniej jego detekcji minęło 100 klatek. Dzięki temu rozwiązaniu, wszystkie wykryte obiekty zostaną zapisane w docelowej liście wykrytych obiektów $DO_{detected}$. Każdy wpis z listy $DO_{detected}$ zawiera informacje o całej historii detekcji, w tym o numerze klatki, w której dana detekcja wystąpiła, prawdopodobieństwie wykrycia oraz o obszarze detekcji. Zawiera również informacje o identyfikatorze obiektu, o jego klasie oraz o ostatniej ramce, w której był wykryty. Powyższe informacje są wykorzystywane do osadzenia danego elementu w wirtualnym świecie 3D. Dane o detekcjach są synchronizowane z danymi GPS, wynikiem czego są detekcje obiektów zlokalizowane według danych GPS.

W celu akwizycji danych GPS powstało urządzenie na bazie mikrokomputera Raspberry Pi oraz modułu GPS – SparkFun GPS-RTK2 Board – ZED-F9P, które może zostać zamontowane razem z kamerą w kabinie pociągu i zbierać dane o aktualnym położeniu pociągu. W związku z niską dokładnością pomiaru położenia przy użyciu standardowego modułu GPS (nie dokładności mogą sięgać nawet kilku metrów) zastosowano korektę danych GPS w postaci danych RTK. Dzięki temu udało się osiągnąć dokładność pomiarową rzędu kilku centymetrów.

Tak przygotowane dane mogą zostać zaimportowane do oprogramowania Hermes w celu dalszej obróbki, czy do wykorzystania ich jako referencja dla modelowania docelowego świata 3D.

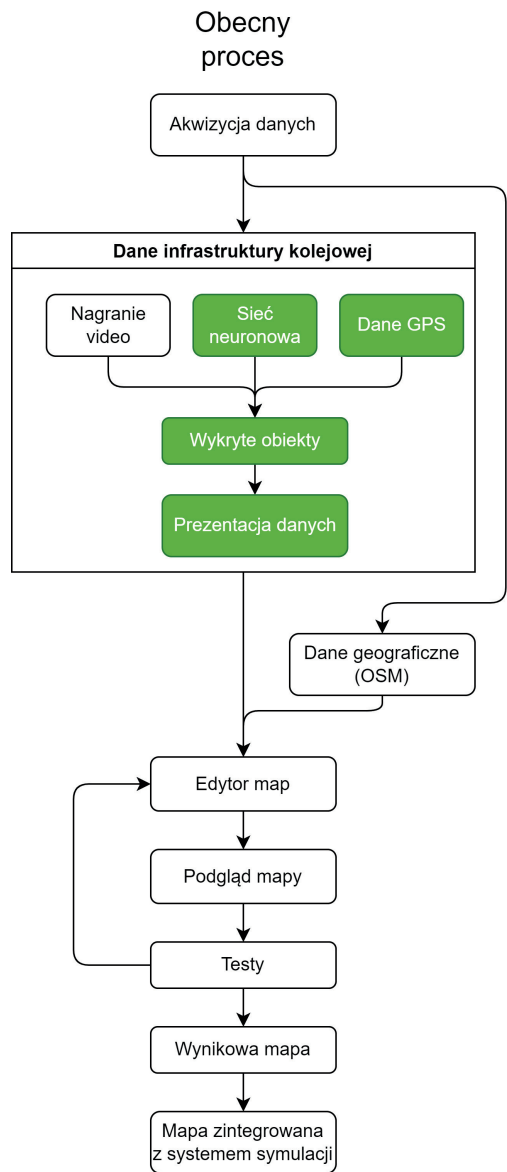
Implementacja zaprezentowanego rozwiązania pozwoliła usprawnić proces modelowania wirtualnego świata 3D na etapie przygotowania danych do modelowania oraz na samym etapie modelowania. W pierwotnym procesie (rys. 4), to dane z kamery były jedynym źródłem informacji o obiektach na trasie kolejowej. W aktualnym procesie modelowania świata 3D (rys. 5) dane zostały rozszerzone o dane detekcji oraz rozmieszczenia obiektów.

Zaproponowane rozwiązanie ma ograniczenia, które mogą zmniejszyć jego skuteczność w procesie wytwarzania wirtualnego świata 3D. Podstawowym problemem jest jakość nagrania wideo. Jeżeli nagranie nie ma odpowiedniej jakości lub jest nagrywane podczas złych warunków atmosferycznych to znaczna część obiektów może nie zostać wykryta lub zostanie nieprawidłowo wykryta. Ten problem występuje również w przypadku manualnego przetwarzania nagrania przez projektanta. Zła jakość obrazu negatywnie wpłynie na wykrywanie przez niego obiektów. Drugi problem jest związany z dokładnością detekcji przez sztuczną sieć neuronową. Podczas testów sieci udało się osiągnąć dokładność detekcji na poziomie 85 %. Wszystkie obiekty, które nie zostały wykryte lub zostały nieprawidłowo wykryte, muszą zostać znalezione ręcznie przez projektanta. Jeżeli dokładność będzie zbyt niska może się okazać, że korzyść jaka płynie z wykorzystywania rozwiązania automatycznej detekcji i klasyfikacji obiektów jest znikoma. Kolejny problem związany jest z bazą danych uczących. Podczas przygotowywania danych uczących, znaczna większość obiektów była oznaczana bardzo blisko czoła pociągu. W związku z powyższym, sieć neuronowa użyta do detekcji i klasyfikacji obiektów nie radzi sobie dobrze z wykrywaniem obiektów w oddali.



Rys. 4. Proces wytwarzania świata 3D bez modułu detekcji

Fig. 4. 3D world manufacturing process without detection module



Rys. 5. Proces wytwarzania świata 3D z wykorzystaniem modułu detekcji
Fig. 5. The process of manufacturing a 3D world using a detection module

5. Podsumowanie

W artykule skupiono uwagę na problemy związane z generacją świata 3D dla symulatorów kolejowych. Przedstawiono i opisano proces tworzenia wirtualnego świata 3D. W ramach realizacji projektu „Innowacyjna technologia tworzenia środowiska dla różnego typu symulatorów pojazdów lądowych i wodnych” powstało rozwiązanie wspierające proces generowania wirtualnego świata 3D dla symulatorów kolejowych. Zaprezentowane rozwiązanie łączy metody detekcji i klasyfikacji (sieć neuronowa) oraz dokładne dane geolokalizacyjne (na podstawie danych GPS) w celu dokładnej identyfikacji i rozmieszczenia elementów w wirtualnym świecie 3D. Rozwiązanie to usprawnia proces tworzenia świata 3D zarówno na etapie przygotowania danych jak i na etapie samego modelowania.

Podziękowania

Zaprezentowane w artykule tezy oraz wyniki stanowią rezultat projektu „Innowacyjna technologia tworzenia środowiska dla różnego typu symulatorów pojazdów lądowych i wodnych”, realizowanego przez Autocomp Management w ramach Regionalnego Programu Operacyjnego Województwa Zachod-

niopomorskiego 2014–2020, Oś Priorytetowa 1 Gospodarka, Innowacje, Nowoczesne Technologie, Działanie 1.1 Projekty badawczo-rozwojowe przedsiębiorstw Typ projektu 2 Projekty badawczo-rozwojowe przedsiębiorstw ukierunkowane na wdrożenie wyników prac B+R w działalności gospodarczej.

Bibliografia

1. Nguyen J., Powers S.T., Urquhart N., Farrenkopf T., Guckert M., *An overview of agent-based traffic simulators*, “Transportation Research Interdisciplinary Perspectives”, Vol. 12, 2021, DOI: 10.1016/J.TRIP.2021.100486.
2. Amara Y., Amamra A., Khemis S., *Raw GIS to 3D road modeling for real-time traffic simulation*, “Visual Computer”, Vol. 38, 2022, 239–256, DOI: 10.1007/S00371-020-02013-1/TABLES/4.
3. Khayyal H.K., Zeidan Z.M., Beshr A.A.A., *Creation and Spatial Analysis of 3D City Modeling based on GIS Data*, “Civil Engineering Journal”, Vol. 8, No. 1, 2022, 105–123, DOI: 10.28991/CEJ-2022-08-01-08.
4. Salleh S., Ujang U., Azri S., *Virtual 3D Campus for Universiti Teknologi Malaysia (UTM)*, “ISPRS International Journal of Geo-Information”, Vol. 10, No. 6, 2021, DOI: 10.3390/ijgi10060356.
5. Keil J., Edler D., Schmitt T., Dickmann F., *Creating Immersive Virtual Environments Based on Open Geospatial Data and Game Engines*, “KN – Journal of Cartography and Geographic Information”, Vol. 71, 2021, 53–65, DOI: 10.1007/s42489-020-00069-6.
6. Girindran R., Boyd D.S., Rosser J., Vijayan D., Long G., Robinson D., *On the Reliable Generation of 3D City Models from Open Data*, “Urban Science”, Vol. 4, No. 4, 2020, DOI: 10.3390/urbansci4040047.
7. Yang X., Delparte D., *A Procedural Modeling Approach for Ecosystem Services and Geodesign Visualization in Old Town Pocatello, Idaho*, “Land”, Vol. 11, No. 8, 2022, DOI: 10.3390/LAND11081228.
8. Wang H., Wu Y., Han X., Xu M., Chen W., *Automatic generation of large-scale 3D road networks based on GIS data*, “Computers & Graphics”, Vol. 96, 2021, 71–81, DOI: 10.1016/J.CAG.2021.02.004.
9. Župan R., Vinković A., Nikçi R., Pinjatela B., *Automatic 3D Building Model Generation from Airborne LiDAR Data and OpenStreetMap Using Procedural Modeling*, “Information”, Vol. 14, No. 7, 2023, DOI: 10.3390/info14070394.
10. Meng Z., Du X., Sottovia P., Foroni D., Axenie C., Wieder A., Eckhoff D., Bortoli S., Knoll A., Sommer C., *Topology-Preserving Simplification of OpenStreetMap Network Data for Large-scale Simulation in SUMO*, “SUMO Conference Proceedings”, Vol. 3, 2022, 181–197, DOI: 10.52825/scp.v3i.111.
11. Jafari A., Both A., Singh D., Gunn L., Giles-Corti B., *Building the road network for city-scale active transport simulation models*, “Simulation Modelling Practice and Theory”, Vol. 114, 2022, DOI: 10.1016/j.simpat.2021.102398.
12. Tabet F., Pentyala S., Patel B.H., Hendawi A., Cao P., Song A., Govind H., Ali M., *OSMRunner : A System for Exploring and Fixing OSM Connectivity*, 22nd IEEE International Conference on Mobile Data Management (MDM), IEEE, 2021, 193–200, DOI: 10.1109/MDM52706.2021.00039.
13. Kilsedar C.E., Brovelli M.A., *Multidimensional Visualization and Processing of Big Open Urban Geospatial Data on the Web*, “ISPRS International Journal of Geo-Information”, Vol. 9, No. 7, 2020, DOI: 10.3390/IJGI9070434.
14. Badwi I.M., Ellaithy H.M., Youssef H.E., *3D-GIS Parametric Modelling for Virtual Urban Simulation Using CityEngine*, “Annals of GIS”, Vol. 28, No. 3, 2022, 325–341, DOI: 10.1080/19475683.2022.2037019.

15. Flenniken J.M., Stuglik S., Iannone B.V., *Quantum GIS (QGIS): An introduction to a free alternative to more costly GIS platforms*. "EDIS", Vol. 2020, No. 2, DOI: 10.32473/edis-fr428-2020.
16. Smaczyński M., Horbiński T., *Creating a 3D Model of the Existing Historical Topographic Object Based on Low-Level Aerial Imagery*, "KN – Journal of Cartography and Geographic Information", Vol. 71, 2021, 33–43, DOI: 10.1007/S42489-020-00061-0/FIGURES/8.
17. Chołodowicz E., Lisiecki P., Lech P., *Hybrid Algorithm for the Detection and Recognition of Railway Signs*, 2020, 337–347, DOI: 10.1007/978-3-030-19738-4_34.
18. Liu W., Wang Z., Zhou B., Yang S., Gong Z., *Real-time Signal Light Detection based on Yolov5 for Railway*, IOP Conference Series: Earth and Environmental Science", Vol. 769, 2021, DOI: 10.1088/1755-1315/769/4/042069.
19. T. Ye, X. Zhang, Y. Zhang, i J. Liu, *Railway Traffic Object Detection Using Differential Feature Fusion Convolution Neural Network*, "IEEE Transactions on Intelligent Transportation Systems", Vol. 22, No. 3, 2021, 1375–1387, DOI: 10.1109/TITS.2020.2969993.
20. Zhang J., Xie Z., Sun J., Zou X., Wang J., *A Cascaded R-CNN with Multiscale Attention and Imbalanced Samples for Traffic Sign Detection*, "IEEE Access", Vol. 8, 2020, 29742–29754, DOI: 10.1109/ACCESS.2020.2972338.
21. Teclaw M. Szłapczyński M., *Generacja terenu przestrzennego za pomocą edytora Hermes*, „Pomiary Automatyka Robotyka”, R. 28, Nr 2, 2024, 113–118, DOI: 10.14313/PAR_252/113.

Automatic Generation of Virtual Railway Infrastructure Elements

Abstract: This article is devoted to the issues of modeling a 3D virtual world for railroad simulators and the problems of creating a map for such a simulator. An algorithm using artificial intelligence methods for detection, classification, and place railway infrastructure objects from video recordings in a 3D virtual world, as well as GPS data has been proposed. The proposed solution, supporting automatic generation of virtual elements of railway infrastructure, is a significant innovation in the field of research.

Keywords: traffic simulator, railway simulator, Convolutional Neural Networks, object detection, 3D world generation, GPS, GIS, OSM, OpenStreetMap

mgr inż. Paweł Lisiecki

plisiecki@autocomp.com.pl
ORCID: 0009-0002-0713-4434

Absolwent Wydziału Elektrycznego Zachodniopomorskiego Uniwersytetu Technologicznego w Szczecinie. W 2017 r. uzyskał tytuł inżyniera, a w 2019 r. tytuł magistra inżyniera. Jest doradcą ds. oprogramowania w Centrum Badawczo-Rozwojowym Autocomp Management Sp. z o.o. Jego zainteresowania naukowe obejmują metody sztucznej inteligencji w zastosowaniu do detekcji obiektów oraz metody analizy i przetwarzania danych biomedycznych.



mgr inż. Maciej Szłapczyński

mszlapczyński@autocomp.com.pl
ORCID: 0009-0004-8037-3460

Absolwent Wydziału Elektrycznego Politechniki Szczecińskiej. W 2007 r. uzyskał tytuł magistra inżyniera. W 2009 r. rozpoczął prace nad projektami badawczo-rozwojowymi w Centrum Badawczo-Rozwojowym Autocomp Management Sp. z o.o. Od 2020 r. jest z-cą dyrektora ds. Badań i Rozwoju. Główne kierunki zainteresowań rozwojowych dotyczą zastosowania symulatorów do szkoleń w obszarach transportu i wojska.



mgr inż. Ewelina Chołodowicz

cholodowicz.ewelina@gmail.com
ORCID: 0000-0001-8221-4309

Doktorantka na Wydziale Elektrycznym w Szkole Doktorskiej na Zachodniopomorskim Uniwersytecie Technologicznym. Uzyskała tytuł magistra inżyniera na kierunku Automatyka i Robotyka na Zachodniopomorskim Uniwersytecie Technologicznym w 2019 r. Zdobyła pierwszą nagrodę za pracę inżynierską w ogólnopolskim konkursie „Młodzi Innowacyjni” oraz wyróżnienie w kategorii „Młody Inżynier” przyznane w 26. edycji Plebiscytu Czytelników „Przeglądu Technicznego”. Jest autorką około 20 publikacji naukowych. Jej zainteresowania naukowe dotyczą modelowania i optymalizacji układów sterowania, w szczególności układów nieliniowych i niepewnych.

