

Andrzej Głowacz*, Zbigniew Mikrut**, Piotr Pawlik**

Algorytm wideodetekcji korzystający z metody obliczania przepływu optycznego

1. Wstęp

Jednym z ważnych zadań projektu INSIGMA [19] jest analiza sytuacji na skrzyżowaniu, która powinna być dokonywana automatycznie przy wykorzystaniu sekwencji obrazów z kamer. Analizowane mają być wielopasmowe drogi lub – w uzasadnionych przypadkach – pojedyncze pasy ruchu. Celem analizy jest m.in. określenie:

- długości kolejek pojazdów (w metrach lub w liczbie oczekujących samochodów),
- liczby pojazdów opuszczających skrzyżowanie,
- prędkości pojazdów.

Proces analizy sekwencji cyfrowych obrazów zwykle dzieli się na kilka etapów. Są to:

- 1) wstępne wykrywanie obiektów: przykładowe metody to skorzystanie z obrazu różnicowego lub pola ruchu, generowanie tła i odejmowanie od niego aktualnego obrazu, poszukiwanie uprzednio zdefiniowanych cech;
- 2) segmentacja, zwykle realizowana przy pomocy operacji progowania, która ostatecznie dzieli obserwowaną scenę na obiekty i tło;
- 3) analiza wykrytych obiektów: etykietowanie, obramowanie, scalanie lub podział obiektów, określenie ich charakterystycznych cech oraz innych parametrów.

W rozdziale 2 zostaną porównane trzy algorytmy obliczania *optical flow*, stanowiące podstawę detekcji pojazdów będących w ruchu. Porównania dokonano pod kątem dokładności wykrywania obiektów oraz czasu obliczeń. W rozdziale 3 zostanie przedstawiony algorytm segmentacji, w którym spróbowano rozwiązać problemy scalania i dzielenia się obiektów odpowiadających pojazdom.

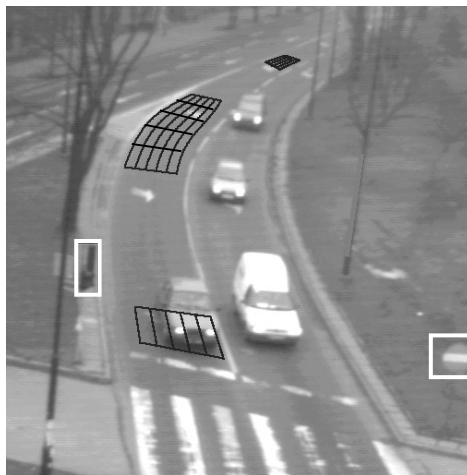
* AGH Akademia Górniczo-Hutnicza, Wydział Elektrotechniki, Automatyki, Informatyki i Elektroniki, Katedra Telekomunikacji, al. A. Mickiewicza 30, 30-059 Kraków

** AGH Akademia Górniczo-Hutnicza, Wydział Elektrotechniki, Automatyki, Informatyki i Elektroniki, Katedra Automatyki, al. A. Mickiewicza 30, 30-059 Kraków

Obliczenia przeprowadzono na sekwencjach wideo, nagranych z górnych pięter domu akademickiego na jednym z krakowskich skrzyżowań. Wyniki działania algorytmu wideodetekcji pokazano na przykładzie zliczania pojazdów w rozdziale 4.

2. Algorytmy obliczania przepływu optycznego

Jako podstawę segmentacji wybrano metodę przepływu optycznego (*optical flow*), która umożliwia wykrywanie obiektów będących w ruchu. Metoda ta była badana w zespole Laboratorium Biocybernetyki w poprzednich latach [15, 10, 1], jednak nie doczekała się praktycznego zastosowania. Testy, z uwagi na dość długi czas obliczeń przepływu optycznego na ówczesnych komputerach, zakończono na etapie segmentacji obiektów (czas analizy był około 3 razy dłuższy od czasu rzeczywistego, pomimo implementacji w języku C i prób optymalizacji). Obecnie – jak się wydaje – zaistniały warunki, aby podjąć eksperymenty ponownie. Dysponujemy o wiele szybszymi, wielordzeniowymi komputerami, równocześnie mając dodatkowe możliwości zrównoleglania obliczeń przy użyciu kart graficznych (*Graphics Processing Unit* – GPU) lub układów reprogramowalnych (*Field Programmable Gate Array* – FPGA).



Rys. 1. Redukcja informacji obrazowej w celu symulacji pętli indukcyjnych (linie czarne) oraz elementy konfiguracyjne krakowskiego wideodetektora (białe prostokąty)

Alternatywą do wzmiankowanej wyżej metody pola przepływu jest algorytm wykrywania statycznych i dynamicznych obiektów za pomocą generowania i odejmowania tła, który był stosowany w krakowskim wideodetektorze [14, 2, 13]. Dzięki odpowiedniej redukcji informacji wizyjnej (do kilku linii pikseli, zdefiniowanych wzdłuż każdego pasa ruchu – por. rys. 1) oprogramowanie wideodetektora pracowało w czasie rzeczywistym – nawet na komputerze przemysłowym klasy PC104+ (Celeron 400 MHz, 128MB SDRAM).

Wideodetektor pracował w trybie symulacji pętli indukcyjnych, ponieważ w owym czasie zakładano, że zastąpi on rzeczywiste pętle i będzie w stanie przekazywać informacje dla sterownika świateł analogicznie jak rzeczywiste pętle indukcyjne. Realizacja za jego pomocą zadań określonych we wstępie wymagałaby zbyt radykalnych zmian, głównie w sposobie generacji tła oraz uwzględnienia podczas analizy interesujących **obszarów** drogi a nie tylko wybranych **linii**. Zmiany te są jednak możliwe do przeprowadzenia i być może zostaną zrealizowane, stanowiąc uzupełnienie metody *optical flow*.

Przewagę metod opartych na obliczaniu pola przepływu autorzy widzą w uniezależnieniu wyników detekcji pojazdów od warunków atmosferycznych oraz w możliwości detekcji pojazdów o kolorze mało różniącym się od tła. Zastosowanie ww. metod wymusza wprowadzenie algorytmów śledzenia wykrytych pojazdów i zapamiętywania ich położenia – przede wszystkim w momentach zatrzymywania się.

Metody wyznaczania przepływu optycznego można podzielić na trzy podstawowe grupy:

- 1) gradientowe, opierające się na analizie pochodnych wartości intensywności obrazu;
- 2) częstotliwościowe, bazujące na filtrach operujących w dziedzinie częstotliwości;
- 3) korelacyjne, przeszukujące przestrzeń obrazu.

Niezależnie od różnic dzielących te metody większość z nich opiera się na trzech fazach obliczeń [4, 5, 3]. Są to:

- a) wstępne przetwarzanie filtrami o selektywnej przepustowości w celu wydobycia pożądanej struktury sygnału oraz poprawienia stosunku poziomu sygnału do szumów;
- b) obliczenie podstawowych miar, takich jak pochodne cząstkowe względem czasu lub lokalne powierzchnie korelacyjne;
- c) integracja miar w celu wyliczenia dwuwymiarowego przepływu optycznego.

Opierając się na pracach [5, 11, 7], których autorzy przeprowadzili porównawcze testy metod służących do wyznaczania przepływu optycznego, oraz na uzyskanych wynikach, [15] stwierdzono, że w pierwszej kolejności należy sprawdzić przydatność dwóch metod gradientowych pierwszego rzędu: metody lokalnej Lucasa–Kanade [12] oraz metody globalnej Horna–Schuncka [8, 9]. Interesujące wydaje się porównanie obu metod ze sobą, ponieważ opierają się one na podobnych danych wejściowych (rzęd pochodnych), jednak sposób obliczeń, dający końcowy rezultat, jest dla obu metod odmienny.

Sposób implementacji obu metod został dokładnie opisany w pracy [15]. Dla obu metod istnieją parametry, które mają wpływ na wyniki. Są to:

- sposób wyznaczenia pochodnych cząstkowych: zastosowana metoda różniczkowania numerycznego, wpływająca na właściwości gradientu i precyzję obliczeń;
- parametr kryterium progowego τ dla metody Lucasa–Kanade, mający wpływ na rozmiar i położenie obszaru, dla którego wyznaczana jest dwuwymiarowa prędkość optyczna oraz normalna prędkość optyczna, co wpływa na precyzję obliczeń;

- parametr α dla metody Horna–Schuncka, regulujący wpływ ograniczenia płynności zmian prędkości optycznej, a przez to wpływający na ciągłość pola przepływu optycznego;
- liczba iteracji dla metody Horna–Schuncka, wpływająca na propagację informacji o prędkości optycznej i pozwalająca na wykrycie większych prędkości dających większe przemieszczenia.

Trzecia testowana metoda (algorytm Broxa) uważana jest za najdokładniejszą, niestety także za najbardziej skomplikowaną obliczeniowo. Została ona zaimplementowana w programie MATLAB na podstawie prac [6, 16] przez Visesh Chariego [18]. Metoda dokonuje globalnej minimalizacji funkcjonau bardzo podobnego do zaproponowanego przez Horna i Schuncka [8]. Zamiast jednej dokonywane są dwie linearyzacje składników funkcjonau. Dodatkowo obliczenia wykonywane są na zadanej liczbie poziomów piramidy obrazu: pole przepływu wyznaczone na obrazach o niższej rozdzielczości jest używane do inicjalizacji obliczeń na obrazach o wyższej rozdzielczości przy użyciu techniki *warping*.

2.1. Przebieg testowania algorytmów

Do wstępnych testów wybrano krótki film, na którym zarejestrowano ruch drogowy na jednym z krakowskich skrzyżowań. Rejestracji dokonano z dość dużej wysokości (z balkonu domu akademickiego), wczesną wiosną (co jest istotne, ponieważ drzewa przesłaniające częściowo widok nie miały liści), przy dżdżystej pogodzie. Ze względu na niezbyt duże prędkości obserwowanych pojazdów przy obliczeniach analizowano co trzecią klatkę filmu.

Pole przepływu obliczano trzema algorytmami:

- Lucasa–Kanade (dla parametru $\tau = 32$);
- Horna–Schuncka w dwóch wariantach: dla parametrów $\alpha = 31$, $iter = 8$ – jako metody różniczkowania użyto centralnej różnicy 4-punktowej, dla parametrów $\alpha = 33$, $iter = 9$ do obliczania pochodnej użyto różnicę pierwszego rzędu;
- algorytmu Broxa [6, 16], w którym zastosowano *warping* i piramidę obrazów.

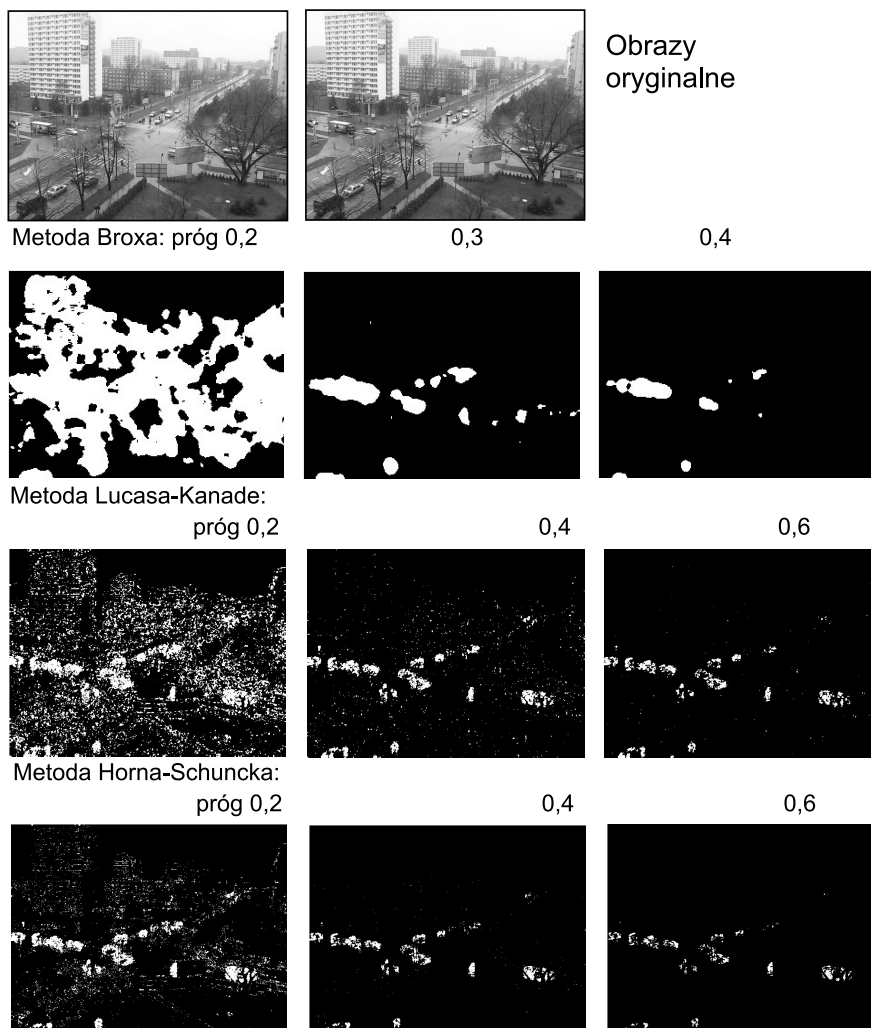
Dla każdego kroku obliczeniowego zapisywano do plików wyznaczone wektory pola przepływu w postaci dwóch składowych: poziomej i pionowej. Dalsze obliczenia przeprowadzono przy pomocy skryptu programu MATLAB. Na tym etapie nie używano informacji o kierunku pola, obliczano jedynie moduły (długości) wektorów, które następnie binaryzowano na kilku poziomach.

Na rysunku 2 przedstawiono wyniki otrzymane dla jednej z ramek filmu. Pokazano zbinaryzowane pole przepływu, przy czym dla każdej z trzech metod dobrano progi, zapewniające najlepszą detekcję ruchomych obiektów. Przedstawione wyniki są reprezentatywne zarówno dla testowanych metod, jak i całego filmu.

2.2. Wyniki porównania algorytmów

Metoda Broxa daje w wyniku dobrze wypełnione obszary obiektów. Jest bardzo wrażliwa na minimalne ruchy kamery (por. rys. 2 dla progu 0,2). Metoda ma tendencję do scalania blisko siebie położonych obiektów (scalone zostały na przykład dwa pojazdy

w pobliżu lewej krawędzi ramki, które nawet dla progu 0,4 nie zostały rozdzielone). Aby wykryć wolniej poruszające się obiekty potrzebny jest niższy próg (0,2) ale wtedy pojawiają się zakłócenia spowodowane drganiem kamery.



Rys. 2. Efekty binaryzacji modułu *optical flow*, obliczonego różnymi metodami

Metoda Lucasa-Kanade – dla progu 0,2 wygenerowała także bardzo dużo zakłóceń, które jednak są mniejsze niż dla metody Broxa. Detekcja jest najbardziej efektywna dla progu 0,4: wykrywanych jest więcej obiektów niż w metodzie Broxa, jednak nie zawsze tworzą one całość (są podzielone).

Metoda Horna–Schuncka – dla progu 0,2 istnieją zakłócenia, jednak są to zakłócenia punktowe i jest ich niewiele w porównaniu z poprzednio omówionymi wynikami. Dla tego progu obiekty są dobrze wypełnione.

Analizę komplikuje mokra nawierzchnia jezdni – obrazy pojazdów odbijają się w niej, zwiększając swoją powierzchnię. Z jednej strony pojazdy są pozornie większe, z drugiej prowadzi to do scalania różnych obiektów.

Niewielkie drgania kamery powodują znaczne zakłócenia pola przepływu. Ten problem jest widoczny na rysunku 2 dla najniższych progów binaryzacji oraz – jeszcze wyraźniej – na innych fragmentach filmów.

W tabeli 1 zestawiono podstawowe informacje o przeprowadzonych testach. W powiązaniu z wynikami, przedstawionymi na rysunku 2, wskazują one metodę Horna–Schuncka jako najbardziej efektywną do realizacji postawionych we wstępie zadań.

Tabela 1
Wyniki wstępnych eksperymentów*

Metoda	Rozmiar obrazu [pix]	Średni czas obliczeń dla jednej ramki** [s]	Zakres (dla ramek bez zakłóceń)	Język implementacji
Horn–Schunck $\alpha = 31$, $iter = 8$	640×480	0,09	[0; 2,67]	C
Horn–Schunck $\alpha = 33$, $iter = 9$	640×480	0,10	[0; 3,36]	C
Lucas–Kanade $\tau = 32$	640×480	0,09	[0; 6,64]	C
Brox	640×480	36,00	– [0; 0,88]	MATLAB MATLAB
	320×240	7,70		

* analizowana była co trzecia ramka filmu

** dla procesora Intel Core i7 920, 2.66GHz (W7 x64)

3. Algorytm detekcji pojazdów

Wejściem do opisywanego algorytmu jest obraz binarny, uzyskany na podstawie pola przepływu (*optical flow*) obliczonego metodą Horna–Schuncka (por. rozdz. 2). Zadaniem algorytmu jest wyodrębnienie poruszających się obiektów (np. samochodów) i ich śledzenie na kolejnych klatkach filmu. Algorytm uwzględnia (w różnym stopniu zaawansowania) problemy: dzielenia jednego obiektu na kilka części, łączenia kilku obiektów w jeden oraz czasowego zatrzymywania się obiektów.

Stwierdzono, że po binaryzacji pola przepływu nie ma gwarancji jednoznacznego odwzorowania samochodu w jeden obiekt wizyjny. Na skutek tego, że w samochodach występują jednolite obszary (które nie generują zmian w polu przepływu optycznego) lub samochód przesłaniany jest przez stałe elementy sceny, może dochodzić do podziału reprezentacji samochodu na dwa lub więcej obiektów. Z kolei na skutek wzajemnego przesłaniania się

przez samochody poruszające się po sąsiednich pasach może dochodzić do łączenia reprezentacji dwóch lub nawet więcej samochodów w jeden obiekt wizyjny. Dlatego po segmentacji został zastosowany algorytm łączący obiekty nieprawidłowo podzielone oraz dzielące te, które zostały błędnie połączone.

Algorytm operuje na liście obiektów (reprezentacji samochodów) zawierającej takie dane jak: pole powierzchni, prostokąt opisany, kilka ostatnich położeń środka ciężkości. Identyfikacja samochodu polega na poprawnym przyporządkowaniu samochodu na aktualnej klatce filmu do obiektu w liście reprezentacji. W przyporządkowaniu wykorzystywana jest także poprzedzająca klatka filmu.

Początkowo (przy pierwszej klatce filmu) zakłada się, że każdy obiekt sceny odpowiada dokładnie jednemu samochodowi. Wyznaczane są podstawowe parametry reprezentacji samochodu – powierzchnia i prostokąt opisany na obiekcie. Dokonywane jest także dołączenie małych obiektów do znacząco od nich większych, z którymi mają odpowiednio duże części wspólne prostokątów opisanych. Potem następuje nadanie unikalnych identyfikatorów reprezentacjom opisującym samochody. Po nadaniu identyfikatorów następuje wyznaczenie kolejnych parametrów opisujących samochody (m.in. środka ciężkości).

Druga (i następne) klatka jest analizowana na podstawie informacji uzyskanych w poprzednim kroku algorytmu. Początek analizy jest identyczny z początkiem pierwszego kroku – tworzona jest pomocnicza lista reprezentacji przy założeniu, że jedna pozycja listy odpowiada dokładnie jednemu samochodowi. Następnie dochodzi do uzgodnienia listy pomocniczej z listą z poprzedniego kroku (nazywaną dalej listą główną). Uzgodnienie polega na przyporządkowaniu rekordom listy pomocniczej rekordów z listy głównej tak, aby oba opisywały ten sam samochód (na dwóch kolejnych klatkach filmu).

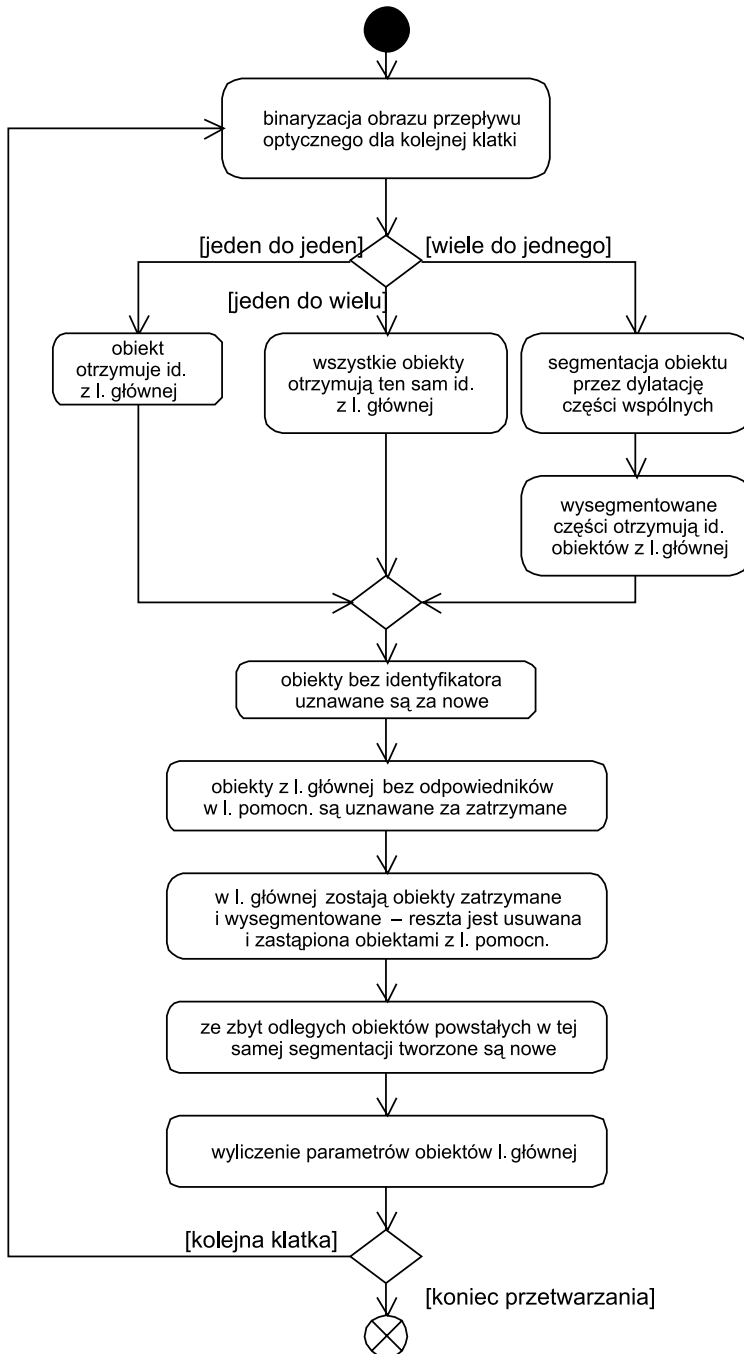
Przy uzgadnianiu trzeba uwzględnić możliwość wystąpienia następujących przypadków:

- a) jednemu rekordowi listy pomocniczej odpowiada jeden rekord listy głównej,
- b) dwóm lub więcej rekordom listy pomocniczej odpowiada jeden rekord listy głównej,
- c) jednemu rekordowi listy pomocniczej odpowiadają co najmniej dwa lub więcej rekordy listy głównej,
- d) wielu rekordom listy pomocniczej odpowiada wiele rekordów listy głównej.

przy czym termin „odpowiadanie” jest rozumiany jako posiadanie (odpowiednio dużej) wspólnej części na dwóch kolejnych klatkach. Trzy pierwsze przypadki ujęto na rysunku 3 w diagramie aktywności ilustrującym omawiany algorytm.

Pierwszy przypadek jest najprostszy – rekord listy pomocniczej otrzymuje unikalny identyfikator odpowiadającego mu rekordu listy głównej – samochód jest reprezentowany przez jeden się obiekt wizyjny sceny.

W drugim przypadku dochodzi do podziału jednego obiektu na kilka. Wszystkie rekordy listy pomocniczej otrzymują identyfikator odpowiadającego im rekordu z listy głównej – odpowiada to sytuacji, kiedy po segmentacji jeden samochód jest reprezentowany przez kilka obiektów. W celu zminimalizowania ryzyka błędnego połączenia kilku obiektów w jeden dokonywana jest analiza odległości środków ciężkości tych obiektów i w wypadku przekroczenia zadanego limitu dochodzi do nadania zbyt „odległym” obiektom nowych identyfikatorów (obiekty te reprezentują nowe pojazdy).



Rys. 3. Diagram aktywności proponowanego algorytmu

Trzeci przypadek jest odwrotnością drugiego – dotyczy połączenia się reprezentacji kilku pojazdów w jeden obiekt wizyjny. Aby nie stracić kontroli nad ruchem każdego z obiektów (samochodów), dokonuje się dodatkowej segmentacji „wspólnego” obiektu wizyjnego przez wykonanie operacji dylatacji morfologicznej po części wspólnej (powstałej po odjęciu obiektów/pojazdów z poprzedniego kadru). Dylatacja jest wykonywana kierunkowo – każdy fragment obiektu jest poszerzany w kierunku swojego dotychczasowego ruchu, zdefiniowanego przez wektor zmiany położenia środka ciężkości. Jeżeli samochód nie poruszał się (ma zerowy wektor zmiany położenia środka ciężkości), to w tym przypadku dokonywana jest dylatacja we wszystkich kierunkach.

Czwarty przypadek (wiele obiektów wizyjnych do wielu samochodów) występuje bardzo rzadko. W chwili obecnej nie jest on rozpatrywany, gdyż nakłady potrzebne do jego rozpatrzenia wielokrotnie przewyższają częstość jego występowania.

Po uzgodnieniu powiązań pomiędzy rekordami obu list znajdowane są rekordy listy pomocniczej niemające odpowiedników w liście głównej. Są one traktowane jako reprezentacje nowych pojazdów i otrzymują kolejne nowe identyfikatory.

Z kolei rekordy listy głównej bez odpowiedników w liście pomocniczej reprezentują samochody, które wyjechały z obszaru kontroli lub zatrzymały się. Są one odpowiednio oznaczane i usuwane po pewnym czasie.

Etap uzgadniania list ma na celu nadanie rekordom listy pomocniczej odpowiednich unikalnych identyfikatorów. Po tym kroku następuje reindeksacja obiektów aktualnego kadru nadanymi identyfikatorami. Na koniec uaktualniana jest lista główna. Pozostawiane są rekordy obiektów uczestniczących w dylatacji (czyli dołączonych do innych obiektów) oraz rekordy odpowiadające samochodom zatrzymanym lub wyjeżdżającym z obszaru zainteresowania. Reszta rekordów jest zastępowana przez rekordy listy pomocniczej. Po tym etapie konieczna jest weryfikacja (*postprocessing*) nowej listy głównej:

- 1) Może w niej wystąpić kilka rekordów z tym samym identyfikatorem (efekt podziału obiektu na kilka mniejszych). W tym przypadku sprawdzane są odległości pomiędzy obiektami reprezentowanymi przez te rekordy w celu znalezienia ewentualnych nowych samochodów. Także rekordom, opisującym zbyt odległe obiekty nadawane są nowe identyfikatory. Pozostałe rekordy łączone są w jeden, który powinien opisywać wszystkie pozostałe obiekty.
- 2) Należy uwzględnić zmiany w obiektach biorących udział w dylatacji.

Z obu powyższych powodów dokonywane jest ponowne wyliczenie parametrów wszystkich obiektów sceny z uwzględnieniem ich nowych indeksów (identyfikatorów).

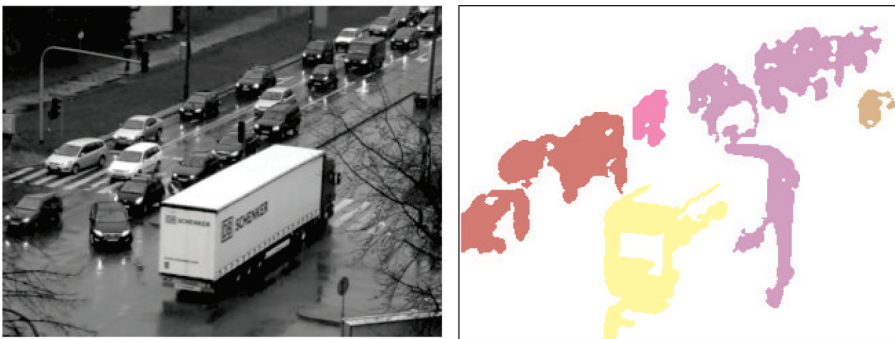
Po tej fazie następuje przejście do nowej klatki i cały proces jest powtarzany.

W wyniku działania powyższego algorytmu uzyskuje się listę aktualnie śledzonych pojazdów. Może ona być zaprezentowana w formie graficznej przez poindeksowanie obrazu binarnego unikalnymi identyfikatorami samochodów. Obserwacja filmów, na których w ten sposób zwizualizowano działanie algorytmu, pokazuje, że w przypadkach, kiedy pojazdy nie są przesłaniwane nie występują żadne problemy z ich śledzeniem (por. rys. 4).



Rys. 4. Przykład prawidłowego śledzenia obiektów nie przesłaniających się

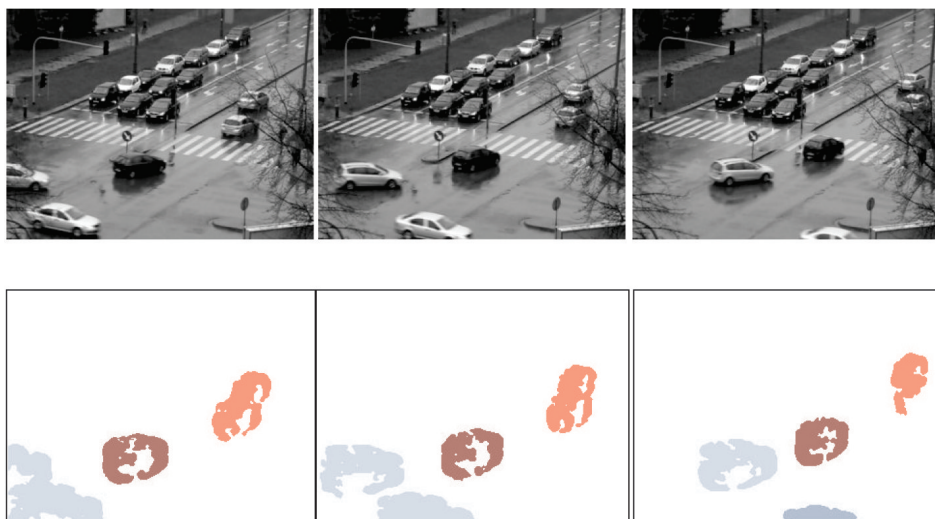
Rzadko występują tu nieprawidłowe sytuacje podzielenia jednego pojazdu na kilka obiektów (w szczególności dotyczy to dużych samochodów ciężarowych z dużymi jednolitymi powierzchniami karoserii – por. rys. 5).



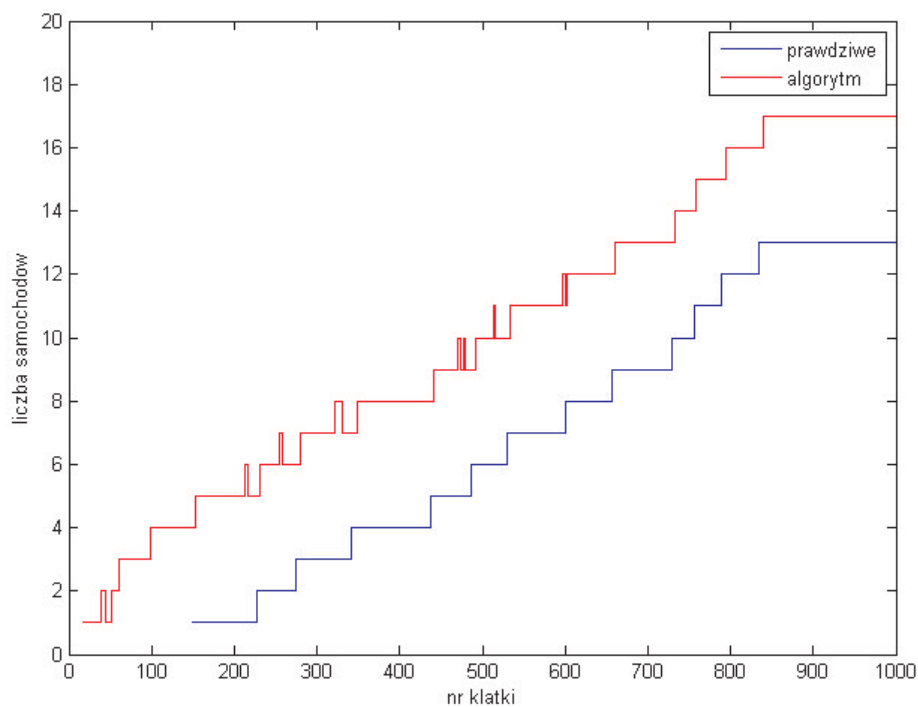
Rys. 5. Przykład nieprawidłowego podziału dużego samochodu na kilka obiektów wizyjnych, oraz połączenia w jeden obiektów nakładających się, reprezentujących blisko siebie jadące (lub przesłaniające się) pojazdy

Prawidłowo działa również rozdzielanie dwóch torów ruchu na pas do jazdy na wprost i pas do skrętu. Odpowiada to sytuacji początkowego uznania dwóch przesłaniających się pojazdów jako jeden obiekt, który następnie jest poprawnie dzielony na dwa – po oddaleniu się samochodów na odpowiednią odległość (por. rys. 6).

Na rysunku 4 kolorem czerwonym zaznaczono linię pomiarową, na której zliczano pojazdy jadące w prawo. Wykonano także oprogramowanie, za pomocą którego można manualnie zliczyć przejeżdżające pojazdy z równoczesnym zapamiętaniem momentu zliczenia. Porównanie działania algorytmu (linia czerwona) z danymi rzeczywistymi (linia niebieska) zostało przedstawione na rysunku 7.



Rys. 6. Przykład sekwencji śledzenia obiektu (szary, w lewym dolnym rogu), który powstał na wjeździe na skrzyżowanie z dwóch samochodów jadących w różnych kierunkach



Rys. 7. Porównanie wyników detekcji (i zliczania) pojazdów z danymi rzeczywistymi

Opisany wyżej algorytm wykrywa pojazdy w momentach bardzo zbliżonych do czasów zarejestrowanych manualnie. Fałszywe wykrycie kilku pierwszych pojazdów przez algorytm było spowodowane niestabilnością kamery (drgania w czasie od 0 do 150 klatki). Wąskie impulsy, które są generowane przez algorytm (np. w okolicy ramek 210, 260, 350 itp.) są spowodowane podziałem jednego obiektu na dwie lub więcej części. Podobną sytuację zilustrowano na rysunku 5. Jeżeli wykryte fragmenty znajdują się blisko siebie, to algorytm potrafi wykryć taką sytuację i – jak widać – prawidłowo ją zinterpretować.

4. Podsumowanie i wnioski

W artykule przedstawiono koncepcję i realizację algorytmu wykrywania i zliczania pojazdów, opartego na analizie pola ruchu (*optical flow*). Porównano efektywność i czas obliczeń trzech algorytmów. Wybrano algorytm Horna–Schuncka i zastosowano go do wydzielania ruchomych obiektów. Stwierdzono, że algorytm dobrze wydziela obiekty po zastosowaniu binaryzacji stałoproęgowej. Skonstruowano podstawowy algorytm detekcji pojazdów, którego działanie i wyniki przedstawiono w rozdziale 3.

Obecnie algorytm nie ma możliwości rozdzielenia pojazdów jadących obok siebie po równoległych pasach ze zbliżoną prędkością. Będą one traktowane jako jeden obiekt. Co więcej zdarzają się także nieprawidłowe przypadki łączenia obiektów w jedną reprezentację pojazdu, mimo że w rzeczywistości należą one do różnych samochodów (por. rys. 5). Jest to efekt podobny do przedstawionego problemu niemożności rozdzielenia nakładających się na siebie obiektów wzmocniony przez algorytm łączenia. Ta część algorytmu wymaga dalszego dopracowania. Być może będzie konieczne zastosowanie reguł semantycznych [17] uwzględniających kształt i kolor obiektów.

Podsumowując działanie proponowanego algorytmu, można stwierdzić że:

- działa on poprawnie przy braku przesłaniania pojazdów;
- prawidłowo rozdzielane są początkowo złączone na skutek przesłaniania, pojazdy poruszające się po pasach, z których jeden jest pasem do skrętu;
- zdarzają się nieprawidłowe podzielenia jednego pojazdu na kilka obiektów (dotyczy to specyficznych pojazdów);
- nie rozdzielane są pojazdy poruszające się obok siebie po równoległych pasach ze zbliżoną prędkością;
- część algorytmu odpowiedzialna za analizę samochodów zatrzymujących się znajduje się w fazie testowania.

Z analizy wykresów przedstawionych na rysunku 7 wynika, że algorytm należy uzupełnić o neutralizowanie drgań kamery. Rozbudowy wymaga także oprogramowanie do konfigurowania obszarów zainteresowania, w których pojazdy mają być śledzone przez algorytm oraz rejestrowane manualnie.

Podziękowania

Praca współfinansowana przez Unię Europejską z Europejskiego Funduszu Rozwoju Regionalnego w ramach Programu Operacyjnego Innowacyjna Gospodarka – projekt INSIGMA nr POIG.01.01.02-00-062/09.

Autorzy dziękują zespołowi studentów w składzie: Szymon Bigaj i Jacek Kościow za napisanie pierwszej wersji oprogramowania do manualnego zliczania pojazdów.

Literatura

- [1] Adamski A., Bubliński Z., Mikrut Z., Pawlik P., *The image-based automatic monitoring for safety traffic lanes intersections*. [w:] J. Piecha (Ed.), Transactions on TRANSPORT SYSTEMS TELEMATICS, Wyd. Politechniki Śląskiej, Gliwice 2004, 92–102.
- [2] Adamski A., Mikrut Z., *The Cracovian prototype of videodetectors feedback in transportation systems*. [w:] J. Piecha (Ed.), Transactions on TRANSPORT SYSTEMS TELEMATICS, Wyd. Politechniki Śląskiej, Gliwice 2004, 140–151.
- [3] Beauchemin S.S., Barron J.L., *The Computation of Optical Flow*. ACM Computing Surveys, 27(3): 1995, 433–467.
- [4] Barron J.L., Beauchemin S.S., Fleet D.J., *On Optical Flow*. 6th Int. Conf. on Artificial Intelligence and Information-Control Systems of Robots. Bratislava, Slovakia, Sept 12–16: 1994, 3–14.
- [5] Barron J.L., Fleet D.J., Beauchemin S.S., *Performance of optical flow techniques*. Int. Journal of Computer Vision, 12(1), 1994, 43–77.
- [6] Brox T., Bruhn A., Papenberg N., Weickert J., *High accuracy optical flow estimation based on a theory for warping*. European Conference on Computer Vision (ECCV), 2004.
- [7] Galvin B., McCane B., Novins K., Mason D., Mills S., *Recovering Motion Fields: An Evaluation of Eight Optical Flow Algorithms*. Proc. of the British Machine Vision Conference (BMVC), September 1998.
- [8] Horn B.K.P., Schunck B.G., *Determining optical flow*. Artificial Intelligence, 17, 1981, 185–204.
- [9] Horn B.K.P., Schunck B.G., *Determining optical flow: a retrospective*. Artificial Intelligence, 59, 1993, 81–87.
- [10] Kotula K., Mikrut Z., *Detection and segmentation of vehicles based on a hierarchical „optical flow” algorithm*. Mat. Konf. TELEMATYKA I BEZPIECZEŃSTWO TRANSPORTU Katowice, październik 2006, 34–46.
- [11] Liu H., Hong T., Herman M., Camus T., Chellappa R., *Accuracy vs. Efficiency Trade-offs in Optical Flow Algorithms*. Computer Vision and Image Understanding (CVIU), 72(3), 1998, 271–286.
- [12] Lucas B.D., Kanade T., *An iterative image registration technique with an application to stereo vision*. Proc. 7th Intl. Joint Conf. on Artificial Intelligence (IJACAI), August 24–28, Vancouver, 1981, 674–679.
- [13] Mikrut Z., *Road Traffic Measurement Using Videodetection*. Image Processing and Communications, vol. 3, No. 3–4, 1997, 19–30.
- [14] Mikrut Z., *The Cracovian Videodetector – from Ideas to Embedding*. Proc. Int. Conf. Transportation and Logistics Integrated Systems ITS-ILS 07, Kraków, October 2007, 29–37.

- [15] Mikrut Z., Pałczyński K., *Segmentacja sekwencji obrazów z wideodetektora na podstawie przepływu optycznego*. Automatyka (półrocznik AGH), t. 7, z. 3, 2003, 371–384.
- [16] Sand P., Teller S., *Particle video*. IEEE Computer Vision and Pattern Recognition (CVPR), 2006.
- [17] Tadeusiewicz R., Ogiela M.R., *Semantic Content of the Images*. [w:] R.S. Choras, A. Zabłudowski (Eds.), Image Processing & Communications Challenges, Academy Publishing House EXIT, Warsaw 2009, 15–29.
- [18] Visesh Chari., *High Accuracy Optical Flow Using a Theory for Warping*. <http://perception.inrialpes.fr/~chari/myweb/Software/> (ostatni dostęp 4 lipca 2010).
- [19] Projekt INSIGMA. AGH, Kraków, 2011. <http://insigma.kt.agh.edu.pl>.