

Zbigniew Bubliński*, Zbigniew Mikrut*, Piotr Pawlik*

Automatyczne wyznaczanie stref kolizji pojazdów

1. Wprowadzenie

Wobec szybko zwiększającej się liczby pojazdów poruszających się po drogach i stale rosnącego natężenia ruchu zagadnienia szeroko rozumianego bezpieczeństwa wszystkich uczestników ruchu drogowego nabierają szczególnie ważnego znaczenia. Istotną rolę w zwiększeniu bezpieczeństwa ruchu drogowego odgrywają systemy automatycznej analizy obrazów [1, 3] rejestrowanych w miejscach potencjalnie niebezpiecznych, np. na skrzyżowaniach. Przykładowy system [2] w pełni realizuje postawione przed nim cele, ale jego szersze stosowanie jest dosyć problematyczne. Ograniczeniem jest bowiem konieczność praco- i czasochłonnego „ręcznego” przygotowania niezbędnych do działania całego systemu obrazów zawierających linie stopów i strefy kolizji. Celowe więc byłoby zautomatyzowanie procesu tworzenia niezbędnych obrazów – przynajmniej w pewnym zakresie – pozwoliłoby to bowiem na upowszechnienie całkiem dobrze działającego systemu. Prezentowany w niniejszej pracy algorytm może stanowić podstawę w pełni automatycznego systemu analizy ruchu pojazdów na skrzyżowaniu z sygnalizacją świetlną wykorzystującągo informację wizyjną.

2. Detekcja stref kolizji

Idea automatycznego wyznaczania stref kolizji jest prosta: nałożenie na siebie torów, po których poruszają się (w tym samym kierunku) widoczne na poszczególnych klatkach zarejestrowanego filmu pojazdy, utworzy „korytarze”. Obszary wspólne tych „korytarzy” (ich przecięcia) są właśnie szukanymi strefami kolizji – a przynajmniej ich w miarę dobrym przybliżeniem, zarówno w sensie położenia (lokalizacji), jak i wielkości (powierzchni). Problemy, jakie pojawiły się przy próbie wykorzystania tej idei w praktyce, oraz sposoby ich przezwyciężenia, przedstawiono w dalszej części niniejszego artykułu.

Obraz przykładowego skrzyżowania z zaznaczonymi istotnymi elementami (liniami stopu oraz strefami kolizji) pokazano na rysunku 1.

* Katedra Automatyki, Akademia Górniczo-Hutnicza, Kraków

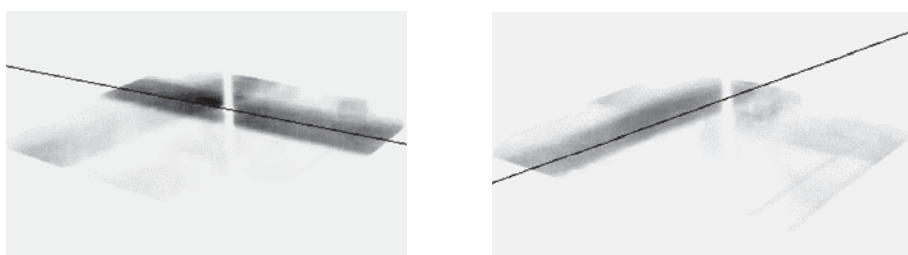


Rys. 1. Obraz przykładowego skrzyżowania

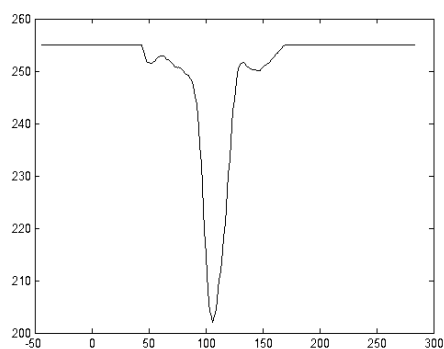
Pierwszym etapem algorytmu jest utworzenie czterech obrazów zawierających ślady poruszających się pojazdów. Do detekcji ruchu wykorzystano metodę przepływu optycznego (*optical flow*) [4, 5]. Każdy obraz zawiera ślady pojazdów przejeżdżających przez skrzyżowanie określonym pasem ruchu – samo wyznaczenie kierunku poruszania się pojazdu polega na sprawdzeniu kierunku i zwrotu wektora prędkości środka ciężkości obiektu. W utworzonych obrazach poziom jasności punktów odzwierciedla liczbę pojazdów, jakie przejechały przez określone miejsca. Kolejnym krokiem jest zredukowanie uzyskanych śladów do pojedynczych linii stanowiących osie poszczególnych pasów ruchu („korytarzy”), a stosowaną metodą jest regresja liniowa. W trakcie badań okazało się jednak, że taki sposób postępowania nie zawsze pozwala na poprawne zlokalizowanie osi – zdarzało się, że wyznaczone linie nie były osiami symetrii pasów ruchu. Z tego względu wprowadzono dodatkowy element algorytmu – iteracyjną korekcję wyliczonego w regresji liniowej współczynnika kierunkowego prostej. Z matematycznego punktu widzenia celem korekcji jest maksymalizacja stosunku liczby punktów ciemnych do jasnych, przy czym uwzględniane są tylko punkty leżące na prostej. Analizowane obrazy (dla dwóch kierunków ruchu) z naniesionymi osiami (już po etapie korekcji) pokazano na rysunku 2.

Jak już wspomniano, wyznaczone proste nie zawsze są osiami symetrii torów ruchu pojazdów, co powoduje, że określanie „korytarzy”, a w konsekwencji stref kolizji, wyłącznie na podstawie tych prostych byłoby obciążone znacznym błędem. Warto jednak zaznaczyć, że współczynnik kierunkowy prostej jest wyliczony poprawnie. Biorąc to pod uwagę, przyjęto, że granice torów ruchu („korytarzy”) będą prostymi równoległymi do znalezionych wcześniej osi. Oznacza to, że jeden z parametrów szukanych prostych (współczynnik kierunkowy) jest już znany. Wyznaczenie drugiego parametru dokonywane jest na podstawie analizy histogramu średniej jasności punktów leżących na szukanych prostych i wybra-

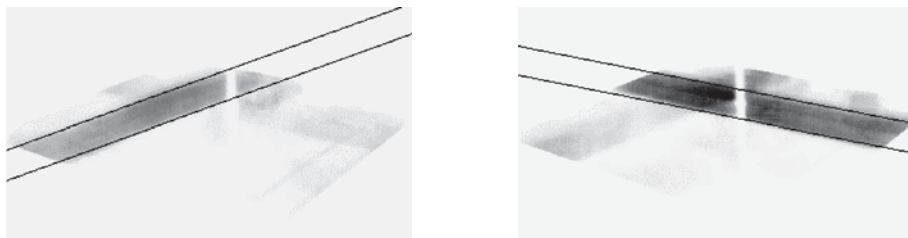
niu takiego parametru, przy którym średnia jasność przekracza ustalony doświadczalnie próg. Przykładowy histogram pokazano na rysunku 3. Widoczne w środku histogramu minimum jasności odpowiada prostej przechodzącej przez centralną część toru ruchu („korytarza”), a w miarę przesuwania prostej w stronę granic „korytarza” średnia jasność rośnie – punkty w obrazie są coraz jaśniejsze, bo mniej pojazdów poruszało się w tych miejscach. Oczywiście jest, że znalezione proste powinny znajdować się po przeciwnych stronach widocznego minimum. Na rysunku 4 pokazano obrazy torów ruchu po zaznaczeniu znalezionych granic.



Rys. 2. Ślady pojazdów z naniesionymi wykrytymi osiami (odpowiada konfiguracji skrzyżowania z rys. 1)



Rys. 3. Histogram średniej jasności punktów leżących na prostej (na osi poziomej przesunięcie)



Rys. 4. Ślady pojazdów z naniesionymi granicami torów ruchu

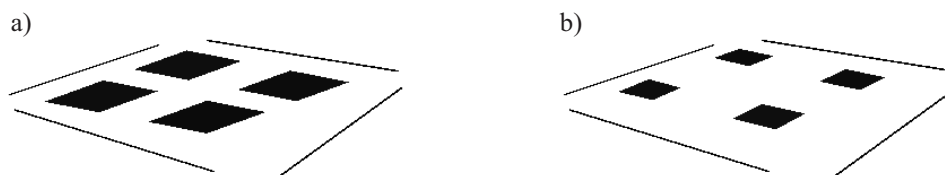
Ostatnim krokiem algorytmu jest „nałożenie” na siebie wyznaczonych w opisany powyżej sposób torów ruchu („korytarzy”). Części wspólne torów (ich przecięcia) utworzą strefy kolizji. Dla zachowania zgodności z systemem, w którym obraz stref kolizji tworzony jest „ręcznie”, konieczne jest odpowiednie zakodowanie znalezionych obszarów (określonym poziomem jasności, jednoznacznie identyfikującym daną strefę kolizji). Tak uzyskane obrazy stref kolizji mogą już być wykorzystane we wcześniej wspomnianym systemie automatycznej analizy ruchu drogowego.

3. Realizacja algorytmu

Przedstawiony powyżej algorytm w wersji finalnej został zaimplementowany jako zewnętrzny filtr działający w programie VirtualDub [6] – wszystkie funkcje zostały napisane w języku C i skompilowane w środowisku MS VisualC++ 6.0. Użytkownik powinien dokonać przetworzenia filmu zarejestrowanego na skrzyżowaniu umieszczając skonstruowany filtr w łańcuchu wykorzystywanych filtrów. Liczbę analizowanych klatek (długość filmu) należy dobrać tak, aby co najmniej kilkanaście pojazdów przejechało w każdym kierunku. Czas przetwarzania filmu jest kilka razy dłuższy od jego czasu trwania (długości filmu) – zależy to głównie od szybkości wykorzystywanego procesora. Po zakończeniu analizy wszystkich klatek zapisywane są obrazy zawierające ślady wykrytych pojazdów, następnie wyznaczane są strefy kolizji, a w ostatnim kroku następuje odpowiednie ich zakodowanie i utworzenie gotowego obrazu stref. Zarówno w przypadku tworzonych obrazów, jak i wykorzystywanych w trakcie przetwarzania (np. zawierających obiekty częściowo zasłaniające skrzyżowanie), obowiązującym formatem jest BMP. Obok implementacji metody przepływu optycznego oraz opisanego algorytmu wyznaczania stref kolizji, w filtrze umieszczono też kod związany z realizacją innych operacji przetwarzania obrazów (np. konwersje RGB do poziomów szarości i odwrotnie, detekcja krawędzi, indeksacja obiektów), których szczegółowe opisy można znaleźć w pracach [7, 8, 9]. Jak już wspomniano, program VirtualDub użyto do zbudowania końcowej, użytkowej wersji narzędzia, zaś na etapie testowania koncepcji i tworzenia prototypów korzystano ze środowiska Matlaba.

4. Uzyskane wyniki

Szczegółowe badania opracowanego algorytmu w oparciu o zgromadzone filmy z różnych skrzyżowań generalnie potwierdzają słuszność zaproponowanej koncepcji. W zdecydowanej większości przypadków uzyskiwane rezultaty są bardzo zbliżone do tych, uzyskanych podczas obserwacji skrzyżowania przez człowieka, a następnie „ręcznego” zaznaczenia stref kolizji. Różnice związane są z przyjętą metodologią postępowania przy wyznaczaniu stref kolizji – automatycznie wyznaczone strefy są większe od tworzonych „ręcznie”, co można zobaczyć na rysunku 5. Nie stanowi to jednak większego problemu, gdyż nie wpływa w zauważalny sposób na rezultaty uzyskiwane w analizie sytuacji niebezpiecznych.



Rys. 5. Porównanie wyników: a) strefy wyznaczone automatycznie; b) strefy wyznaczone „ręcznie”

W pewnych sytuacjach opisana metoda nie daje jednak dobrych wyników. Dzieje się tak, gdy skrzyżowanie zawiera więcej niż jeden pas ruchu w każdym kierunku lub gdy ma nietypowy przebieg (np. w kształcie litery T). Wtedy liczba automatycznie wykrytych stref kolizji będzie inna, niż gdyby je zaznaczać „ręcznie” – zwykle mniejsza. Wynika to wprost z założeń przyjętych na etapie tworzenia algorytmu (klasyczne skrzyżowanie z pojedynczymi pasami ruchu w każdą stronę). Należy zwrócić uwagę, że problemem jest nie tyle samo zmniejszenie liczby stref kolizji, co ograniczona precyzja rozpoznania sytuacji kolizyjnych. W praktyce ogromna większość skrzyżowań „pasuje” do założeń, a w sytuacjach szczególnych zawsze możliwe jest przygotowanie obrazów stref kolizji w sposób „ręczny”.

Innym przypadkiem, w którym należy liczyć się z możliwymi błędami, jest sytuacja, gdy przez skrzyżowanie przejeżdżają pojazdy o znacznej wielkości: autobusy, samochody ciężarowe, pojazdy specjalne. Wtedy wykryte tory ruchu są znacznie większe niż faktyczna część jezdni zajmowana przez pojazd (analizowany ślad jest rzutem na płaszczyznę trójwymiarowego obiektu). Pewnym sposobem na zminimalizowanie tego problemu jest uwzględnianie tylko niewielkich pojazdów (samochodów osobowych), szczególnie że pojawienie się na skrzyżowaniu pojazdu o znacznych rozmiarach skutkuje znacznymi błędami podczas detekcji sytuacji kolizyjnych – tym bardziej nie ma sensu zajmowanie się takimi pojazdami na etapie wstępnym, jakim jest tworzenie stref kolizji.

Również w sytuacji, gdy natężenia ruchu w poszczególnych kierunkach są bardzo zróżnicowane, może to stanowić przyczynę nie do końca poprawnego zlokalizowania stref kolizji, choć w tym przypadku wprowadzenie odpowiednich poprawek do algorytmu detekcji jest stosunkowo łatwe i proste.

5. Podsumowanie

Przeprowadzone eksperymenty w pełni potwierdziły przydatność opracowanej metody automatycznej detekcji stref kolizji na skrzyżowaniach o klasycznej budowie, z pojedynczymi pasami ruchu w każdą stronę. Uzyskiwane rezultaty były zgodne z tymi, jakie otrzymano w przypadku, gdy w systemie analizy wykorzystano z obrazów stref kolizji przygotowanych „ręcznie”. Eliminacja dosyć czasochłonnego i żmudnego procesu przygotowywania obrazów ze strefami kolizji to nie jedyna korzyść wynikająca z zastosowania opisanego algorytmu. Inną, równie – o ile nie bardziej – istotną jest wyeliminowanie możliwości popełnienia błędu przez użytkownika.

W pojedynczych przypadkach zaobserwowano nieprawidłowe działanie metody. Przyczyny takiego stanu rzeczy, jak i sposoby na poprawienie uzyskiwanych wyników przedstawiono powyżej. Warto jednak zauważyć, że w pewnych sytuacjach nawet korzystanie z idealnie przygotowanych obrazów ze strefami kolizji nie gwarantuje poprawnych rezultatów końcowej analizy sytuacji kolizyjnych – dotyczy to szczególnie ruchu dużych pojazdów.

Dalsze badania będą się koncentrować na rozszerzeniu opracowanego algorytmu tak, aby zapewnić jego prawidłowe działanie dla możliwie jak największej liczby skrzyżowań. Celowe byłoby też zaimplementowanie i przetestowanie alternatywnych metod wyznaczenia granic torów ruchu („korytarzy”).

Podziękowania

Przedstawione rezultaty zostały uzyskane w ramach badań własnych – umowa AGH nr 10.10.120.39.

Autorzy pragną podziękować Panom Markowi Szelidze i Dominikowi Szymosze za pomoc w przeprowadzeniu opisanych w niniejszej pracy eksperymentów na filmach z zarejestrowanym ruchem pojazdów w obrębie skrzyżowań.

Literatura

- [1] Adamski A.: *Inteligentne systemy transportowe: sterowanie, nadzór i zarządzanie*. Kraków, UWND AGH 2003
- [2] Adamski A., Bubleński Z., Mikrut Z., Pawlik P.: *Image Analysis Based Automatic Safety Monitoring System For Traffic Intersections*. IV Konferencja „Telematyka i Bezpieczeństwo Ruchu Drogowego”, 19-20 XI 2004, Katowice-Ustroń
- [3] Hoose N.: *Computer Image Processing In Traffic Engineering*. Taunton, RSP Ltd., 1991
- [4] Horn B.K.P., Schunck B.G.: *Determining Optical Flow*. Artificial Intelligence, 17, 1981, 185–204
- [5] Horn B.K.P., Schunck B.G.: *Determining Optical Flow: A Retrospective*. Artificial Intelligence, 59, 1993, 81–87
- [6] Lee A.: *VirtualDub project homepage*. <http://www.virtualdub.org>, Internet, 2004
- [7] Parker J.R.: *Algorithms for Image Processing and Computer Vision*. New York, Wiley Computer Publishing 1997
- [8] Tadeusiewicz R., Korohoda P.: *Komputerowa analiza i przetwarzanie obrazów*. Kraków, Wydawnictwo Fundacji Postępu Telekomunikacji 1997
- [9] Umbaugh S.E.: *Computer Vision and Image Processing*. Upper Saddle River, Prentice Hall 1998