

**A G H**

**AKADEMIA GÓRNICZO-HUTNICZA IM. STANISŁAWA STASZICA W  
KRAKOWIE**

**WYDZIAŁ ELEKTROTECHNIKI, AUTOMATYKI,  
INFORMATYKI I INŻYNIERII BIOMEDYCZNEJ**

KATEDRA AUTOMATYKI I ROBOTYKI

**Praca dyplomowa magisterska**

*Sprzętowo-programowy system wizyjny do detekcji  
obiektów z wykorzystaniem termowizji*

*Hardware-software vision system for object detection with  
the use of thermovision.*

Autor: Tomasz Kańka

Kierunek studiów: Automatyka i Robotyka

Opiekun pracy: dr inż. Tomasz Kryjak

Kraków, 2018

*Uprzedzony o odpowiedzialności karnej na podstawie art. 115 ust. 1 i 2 ustawy z dnia 4 lutego 1994 r. o prawie autorskim i prawach pokrewnych (t.j. Dz.U. z 2006 r. Nr 90, poz. 631 z późn. zm.): „Kto przywłaszcza sobie autorstwo albo wprowadza w błąd co do autorstwa całości lub części cudzego utworu albo artystycznego wykonania, podlega grzywnie, karze ograniczenia wolności albo pozbawienia wolności do lat 3. Tej samej karze podlega, kto rozpowszechnia bez podania nazwiska lub pseudonimu twórcy cudzy utwór w wersji oryginalnej albo w postaci opracowania, artystycznego wykonania albo publicznie zniekształca taki utwór, artystyczne wykonanie, fonogram, videogram lub nadanie.”, a także uprzedzony o odpowiedzialności dyscyplinarnej na podstawie art. 211 ust. 1 ustawy z dnia 27 lipca 2005 r. Prawo o szkolnictwie wyższym (t.j. Dz. U. z 2012 r. poz. 572, z późn. zm.): „Za naruszenie przepisów obowiązujących w uczelni oraz za czyny uchybiające godności studenta student ponosi odpowiedzialność dyscyplinarną przed komisją dyscyplinarną albo przed sądem koleżeńskim samorządu studenckiego, zwanym dalej « sądem koleżeńskim ».”, oświadczam, że niniejszą pracę dyplomową wykonałem(-am) osobistie i samodzielnie i że nie korzystałem(-am) ze źródeł innych niż wymienione w pracy.*

*Serdecznie dziękuję siostrze i ojcu i Ani za użyczenie swojego ciepła, oraz dr. inż. Tomaszowi Kryjakowi za stałą opiekę i nieocenioną pomoc w trakcie pisania pracy.*



## Streszczenie

Dobry system detekcji pieszych znajduje przykładowo zastosowanie w zaawansowanym wspomaganiu kierowcy bądź monitoringu. W niniejszej pracy został zaproponowany sprzętowo-programowy system wizyjny bazujący na obrazie multispektralnym. Obraz multispektralny jest otrzymywany z połączenia obrazów z kamery termowizyjnej i wizyjnej. Obszar zainteresowania jest wybierany na podstawie klasyfikatora wykorzystującego wzorzec probabilistyczny w termowizji. Następnie wskazany obszar jest ponownie klasyfikowany z wykorzystaniem HOG i SVM w przestrzeni multispektralnej by potwierdzić obecność przechodnia.

W pracy wykorzystano miniaturową kamerę termowizyjną Lepton o rozdzielcości 80x60 pikseli oraz wizyjną o rozdzielcości 640x480 px. Do realizacji systemu został użyty układ heterogeniczny Zynq-7000 firmy Xilinx, który zawiera układ FPGA oraz procesor ARM w jednej obudowie. Logika programowalna została wykorzystana do połączenia obrazów w multispektralny oraz do wykrycia przechodniów za pomocą wzorca probabilistycznego, natomiast w systemie procesorowym został zrealizowany HOG i SVM.

**Słowa kluczowe:** termowizja, ARM, FPGA, detekcja przechodniów, obraz multispektralny, HOG SVM, wzorzec probabilistyczny

## Abstract

A robust pedestrian detection system is crucial in many applications such as advances driver-assistance or surveillance. In this thesis a multispectral image based hardware-software vision system is proposed. The multispectral image is obtained by combining outputs of vision and infrared cameras. The ROIs (Regions of interest) are selected using a probabilistic template classifier from infrared image. The HOG descriptor and SVM classifier working in multispectral imaging domain is used to confirm that the selected area contains pedestrian.

The system is based on Lepton miniature infrared camera with a resolution of 80x60 pixels and color vision camera (640x480 pixels). To implement the system, Xilinx's SoC Zynq-7000, which integrates an FPGA and an ARM processor in a single device, is used. Programmable logic is used to combine images into a multispectral domain and to detect pedestrian with probabilistic template, while HOG and SVM is been implemented in processing system.

**Keywords:** infrared vision, ARM, FPGA, pedestrian detection, multispectral images, HOG SVM, probabilistic template



# **Spis treści**

<b>1. Wprowadzenie .....</b>	9
1.1. Cel pracy .....	11
1.2. Struktura pracy .....	11
<b>2. Multispektralny system wizyjny .....</b>	13
2.1. Podczerwień .....	13
2.2. Kamera termowizyjna.....	14
2.2.1. Sensor podczerwieni.....	14
2.2.2. Kamera termowizyjna FLIR Lepton.....	15
2.3. Rejestracja obrazu multispektralnego.....	17
2.3.1. Model geometryczny .....	19
2.3.2. Kalibracja .....	19
<b>3. Algorytmy detekcji pieszych .....</b>	23
3.1. Ustalenie regionu zainteresowania .....	23
3.2. Wyodrębnienie cech .....	24
3.3. Klasyfikator .....	26
3.4. Specyfikacja detekcji w podczerwieni.....	28
<b>4. Wykorzystanie układów FPGA i Zynq w przetwarzaniu i analizie obrazu .....</b>	31
4.1. Układ Zynq-7000.....	32
4.2. Interfejs AXI.....	33
<b>5. Przegląd metod detekcji pieszych.....</b>	35
5.1. Zaawansowana binaryzacja i segmentacja + HOG SVM.....	35
5.2. Stereotermowizja .....	36
5.3. Podejście sprzętowo-programowe. Stereowizja dla robotów.....	37
5.4. Podejście sprzętowo-programowe. System wspomagania kierowcy.....	38
5.5. Wzorzec probabilistyczny .....	39

5.6. Praca Inżynierska.....	40
<b>6. Zrealizowany system wizyjny do detekcji ludzi .....</b>	<b>43</b>
6.1. Koncepcja systemu .....	43
6.2. Model programowy .....	45
6.3. Wykorzystanie AXI-Stream do transmisji sygnału wideo. ....	47
6.4. AXI VDMA.....	47
6.5. Opis modułów zaimplementowanych w logice programowalnej .....	48
6.5.1. Kontroler kamery IR.....	48
6.5.2. Transformata projekcyjna.....	49
6.5.3. Interpolacja dwuliniowa .....	50
6.5.4. Łączenie strumieni.....	51
6.5.5. Koloryzacja i nakładanie .....	51
6.5.6. Obramowanie wyników.....	51
6.5.7. Moduł DPM.....	53
6.6. System procesorowy .....	53
6.7. Proces kalibracji .....	54
6.8. HOG i SVM.....	55
6.9. Wyniki .....	56
<b>7. Podsumowanie i możliwe dalsze kierunki rozwoju systemu .....</b>	<b>59</b>
<b>A. Dodatek A – zawartość płyty CD.....</b>	<b>65</b>
<b>B. Dodatek B – Instrukcja obsługi systemu.....</b>	<b>67</b>
<b>C. Dodatek C – Opis informatyczny projektu.....</b>	<b>69</b>

# 1. Wprowadzenie

Cyfrowa analiza obrazów znalazła szerokie zastosowanie w wielu dziedzinach życia. Dzięki niej możliwe jest automatyczne uzyskanie istotnych dla użytkownika informacji. Przez ostatnie kilkadziesiąt lat opracowano tysiące różnych technik i algorytmów wyspecjalizowanych do określonych zadań np. leśne fotopułapki do badania zachowań i migracji zwierząt, systemu kontroli jakości i przebiegu procesu przemysłowego, metody kontroli dostępu poprzez rozpoznanie twarzy m.in. w smartfonach, algorytmu analizy zdjęć satelitarnych ziemi umożliwiające prognozowanie pogody, sterowanie ruchem drogowym na podstawie obrazu z kamer zamontowanych nad skrzyżowaniami, a także systemy do badania przekroju żołędzia pozwalające określić czy jest on dobrym kandydatem na sadzonkę.

Wzrok ludzki umożliwia rejestrację obrazu w pewnym zakresie promieniowania elektromagnetycznego zwanego światłem widzialnym. Dzisiejsza technologia daje możliwość rejestracji poza tym widmem. Przykładem są kamery termowizyjne, które stają się coraz tańsze i przez to bardziej popularne. Dostarczają one информацию o temperaturze obserwowanych obiektów. Jest to coraz częściej wykorzystywane np. w weterynarii do określenia miejsc urazów zwierząt, w przemyśle do kontroli jakości artykułów spożywczych, w budownictwie do analizy strat cieplnych w budynkach, w systemach wspomagania kierowcy do detekcji obiektów w pobliżu drogi, przez ratowników do odnajdywania zasypanych ludzi w gruzowiskach, straż graniczną do monitorowania granic, przez wojsko do odnajdywania celów i zagrożeń podczas misji m.in. z wykorzystaniem dronów [1].

Większość systemów wizyjnych służących do detekcji przechodniów jest oparta o analizę obrazów z zakresu światła widzialnego bądź podczerwieni. W przypadku światła widzialnego można uzyskać bardzo dobre wyniki pod warunkiem, że wyszukiwane obiekty są dobrze oświetlone i wyróżniają się swoim kolorem od tła. Podczerwieni, a szczególnie termowizja, umożliwia detekcję w warunkach nocnych i ograniczonej widoczności. Oba podejścia mają swoje wady i zalety, które wzajemnie się uzupełniają (np. duże nasłonecznienia powoduje, że tło termiczne staje się dużo wyższe, co utrudnia wyodrębnienie pieszego, natomiast potencjalnie daje idealne warunki do uzyskania dobrej jakości obrazu w zakresie widzialnym) [2]. Połączenie tych dwóch

obrazów daje możliwość uzyskania jeszcze lepszej skuteczności rozpoznawania ludzi. W pracy [3] autorzy nazywają ten rozszerzony format jako RGBT („Red-Green-Blue-Thermal”), natomiast inna praca jako analizę multispektralną (ang. *Multispectral*) [4], albo po prostu jako połączony obraz z kamery termowizyjnej i wizyjnej [2].

Skuteczna detekcja obiektów często wymaga dużego zapotrzebowania na zasoby obliczeniowe. W wielu przypadkach nie da się uzyskać satysfakcyjującej wydajności – tak by można było uznać system za działający w czasie rzeczywistym – wykorzystując jedynie typowy komputer wyposażony w procesor ogólnego przeznaczenia (nawet najnowszej generacji). Stosuje się zatem różne metody akceleracji obliczeń. Jednym z podejść jest zastosowanie kart graficznych (GPU ang. *Graphics Processing Unit*). Pozwalają one na duże zrównoleglenie obliczeń, jednak wciąż charakteryzują się znacznym zużyciem energii (choć należy zaznaczyć, że obecnie trwają intensywne prace nad poprawą ich efektywności energetycznej). Tworzenie specjalizowanych układów scalonych (ASIC ang. *Application-Specific Integrated Circuit*) daje najlepsze rezultaty w implementacji systemu wizyjnego, ale ich opracowanie i produkcja wymaga bardzo dużych nakładów finansowych.

Dobre rozwiązanie stanowią układy rekonfigurowalne, które charakteryzują się podobnymi możliwościami w realizacji wyspecjalizowanych zadań co układy ASIC, ale nie wymagają tworzenia całkiem nowych układów scalonych. Układy FPGA (ang. *Field-Programmable Gate Array*) umożliwiają zrównoleglenie obliczeń i są szeroko stosowane w systemach wizyjnych. Szczególnie chętnie są wykorzystywane do realizacji operacji niskiego poziomu, przygotowując wstępnie obraz do dalszej analizy na wysokim poziomie. Przykłady takich operacji to: filtry konwolucyjne, filtry 2D, podpróbkowanie, wykrywanie krawędzi, obliczanie SAD (ang. *Sum Of Absolute Differences*) z regionu zainteresowania, obliczanie orientacji krawędzi i histogramów, obliczanie strumieniowo statystyk (wartość maksymalna, minimalna, średnia), zmiana przestrzeni barw [5]. Dodatkową zaletą układów FPGA jest mały pobór mocy, co czyni je niezwykle atrakcyjne dla aplikacji mobilnych – takich jak drony czy czujniki środowiskowe [6].

Układy heterogeniczne łączą w jednej obudowie dwa układy (zasoby obliczeniowe) o różnej architekturze i funkcjonalności. Przykładem takiego połączenia jest Zynq-7000 firmy Xilinx, który integruje w sobie układ FPGA oraz procesor ARM. Największą zaletą takiego rozwiązania jest wysoka przepustowość transferu danych między procesorem, a logiką programowaną, co pozwala na wydajniejszą pracę w przypadku rozdzielenia algorytmu między te dwa układy.

Niniejsza praca stanowi kontynuację i rozwinięcie pracy inżynierskiej autora.

## 1.1. Cel pracy

Celem pracy była realizacja wbudowanego systemu wizyjnego do detekcji wybranych obiektów (np. ludzi) na podstawie obrazu z kamery termowizyjnej. Założono, że jako platforma obliczeniowa zostanie użyty układ heterogeniczny (np. Zynq firmy Xilinx), który umożliwia implementację sprzętowo-programową algorytmów. W pierwszym etapie pracy należało dokonać przeglądu i oceny zrealizowanego w ramach pracy inżynierskiej algorytmu, a także przeprowadzić pogłębioną analizę literatury związanej z tematem. Należało przy tym zwrócić szczególną uwagę na systemy detekcji pieszych (wspomaganie kierowcy) oraz systemy dla autonomicznych pojazdów latających (dronów). Ponadto należało przeanalizować możliwość wykorzystania kontekstu czasowego tj. informacji zawartej w sekwencji obrazów. Na tej podstawie należało wytypować algorytmy, które zostaną zaimplementowane i przetestowane w aplikacji programowej (Matlab/C++/Python/OpenCV). Należało również zgromadzić bazę zdjęć lub sekwencji testowych.

W drugim etapie należało wybrane wspólnie z opiekunem pracy algorytmy zaimplementować w systemie programowo sprzętowym, uruchomić oraz sprawdzić ich skuteczność w różnych scenariuszach testowych.

Oczekiwanym rezultatem pracy był: opis wykorzystania informacji termowizyjnej w detekcji pieszych (systemy wspomagania kierowcy) oraz detekcji ludzi z wykorzystaniem dronów, implementacja i analiza kilku podejść (w aplikacji programowej), implementacja sprzętowo-programowo wybranych algorytmów detekcji.

## 1.2. Struktura pracy

Prace rozpoczyna wstęp, w którym przedstawiono możliwości i zastosowania kamer termowizyjnych. Wskazano również zalety wykorzystania podczerwieni do rozszerzenia spektrum konwencjonalnej kamery wizyjnej – analizy multispektralnej. W rozdziale 2 zostały omówione sposoby uzyskania obrazu multispektralnego. Rozdział rozpoczyna się od scharakteryzowania promieniowania podczerwonego oraz metody jego rejestracji. Następnie został zaprezentowany model kamery otworkowej oraz transformacja projekcyjna, które stanowią podstawę teoretyczną do prawidłowego połączenia obrazów z dwóch różnych kamer. W rozdziale 3 przedstawiono ogólny przegląd algorytmów służących do detekcji pieszych. W poszczególnych sekcjach zostały omówione kolejne etapy przetwarzania obrazu i klasyfikacji wraz z ogólnie stosowanymi rozwiązaniami. Następnie w rozdziale 4 zostały omówione sposoby zrównoleglenia obliczeń i możliwość FPGA na przykładzie układu heterogenicznego Zynq-7000 firmy Xilinx. Kolejny rozdział 5 zawiera streszczenia kilku wybranych artykułów, które poruszają tematykę

związaną z niniejszą pracą. W rozdziale 6 przedstawiono koncepcję zaproponowanego rozwiązania wraz ze szczegółami dotyczącymi jego wykonania. Pracę zakończono podsumowaniem.

## 2. Multispektralny system wizyjny

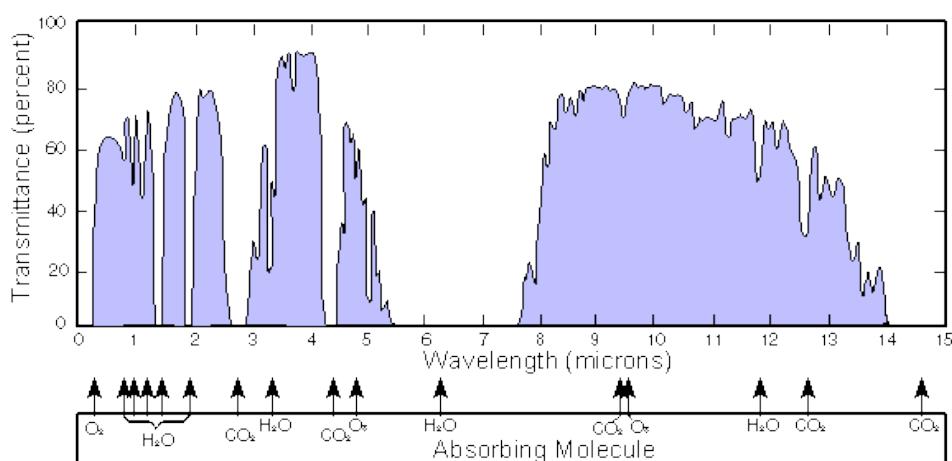
Rozdział rozpoczyna się od charakterystyki promieniowania podczerwonego oraz jego podział na zakresy. Następnie został zaprezentowany sposób rejestracji obrazu termowizyjnego na podstawie kamery Lepton firmy FLIR. Dalej został omówiony dwa sposoby uzyskiwania obrazu poprzez rozdzieleniem wiązki świetlnej na dwa sensory oraz za pomocą układu dwóch kamer równoległych. Następnie został przedstawiony model geometryczny kamery otworkowej oraz sposób kalibracji obrazów zarejestrowanych dwa różnymi kamerami.

### 2.1. Podczerwień

Każde ciało, które ma temperaturę wyższą niż zero absolutne, emituje swoją powierzchnią promieniowanie cieplne. Im wyższa temperatura ciała tym większe natężenie tego promieniowania. Dla każdej temperatury danego ciała istnieje charakterystyczna długość fali o najwyższej wartości mocy promieniowania. Wraz z jej wzrostem, ta częstotliwość przesuwa się w zakres fal widzialnych. Można to zaobserwować, gdy stal osiąga wysoką temperaturę, co skutkuje emisją światła. Zależność ta jest opisana prawem Plancka, które opisuje emisję promieniowania elektromagnetycznego przez ciało doskonale czarne. Ciało doskonale czarne to wyidealizowane ciało fizyczne, które całkowicie pochłania padające na nie promieniowanie oraz emisję promieniowania ściśle związane z jego temperaturą. Wykres na rysunku 2.1 przedstawia tę zależność.

Mianem podczerwieni określa się promieniowanie elektromagnetyczne w zakresie fal o długości od  $0,75 \mu m$  do  $1000 \mu m$ . Wyróżnia się następujące pasma podczerwieni:

- Bliska podczerwień (NIR ang. *near infrared*) w zakresie  $0,75 \mu m$  do  $1,4 \mu m$ .
- Podczerwień fal krótkich (SWIR ang. *short-wavelength infrared*) w zakresie  $1,4 \mu m$  do  $3 \mu m$ .
- Podczerwień fal średnich (SWIR ang. *mid-wavelength infrared*) w zakresie  $3 \mu m$  do  $8 \mu m$ .



**Rys. 2.1.** Wykres transmisyjności atmosfery dla promieniowania podczerwonego [8].

- Podczerwień fal długich (LWIR ang. *long-wavelength infrared*) w zakresie  $8\mu m$  do  $15\mu m$ .
- Daleka podczerwień (FIR ang. *long-wavelength infrared*) w zakresie  $15\mu m$  do  $1000\mu m$ .

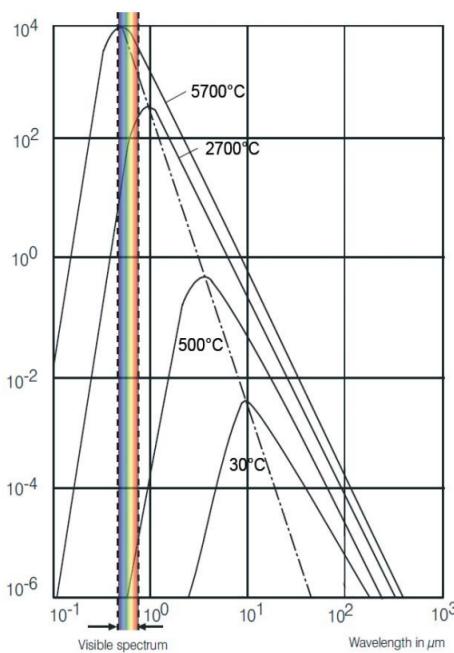
Bliska podczerwień znajduje się tuż za zakresem światła widzialnego ludzkiem wzrokiem i jest możliwa do rejestracji przez typowe dla kamer sensory CCD czy CMOS (często z zastosowaniem dodatkowych oświetlaczy IR). Wraz ze wzmacniaczem światła jest również stosowana w noktowizji.

SWIR i LWIR występują także pod nazwą termowizji. Promieniowanie podczerwone jest częściowo pochłaniane przez atmosferę ziemską. Na rysunku 2.2 przedstawiono tzw. transmisję atmosfery. W aparaturze rejestrującej w podczerwieni wykorzystuje się dwa zakresy, przy których transmisja jest największa:  $3 - 5 \mu m$  oraz  $8 - 14 \mu m$  [7].

## 2.2. Kamera termowizyjna

### 2.2.1. Sensor podczerwieni

W kamerach do rejestracji obrazu w termowizji są wykorzystywane sensory FPA (ang. *Focal Plane Array* – płaskie zespoły ogniskujące). Najbardziej popularne typy to: InSb (Ind Arsen), InGaAs (Ind Gal Arsen), HgCdTe (Rtęć Kadm Tellur) (w postaci fotodiód; wymagają kriogenicznych warunków pracy) and QWIP (ang. *Quantum well infrared photodetector*). Najsłynniejsze technologie wykorzystują niskobudżetowe, niewymagające chłodzenia mikrobolometry.



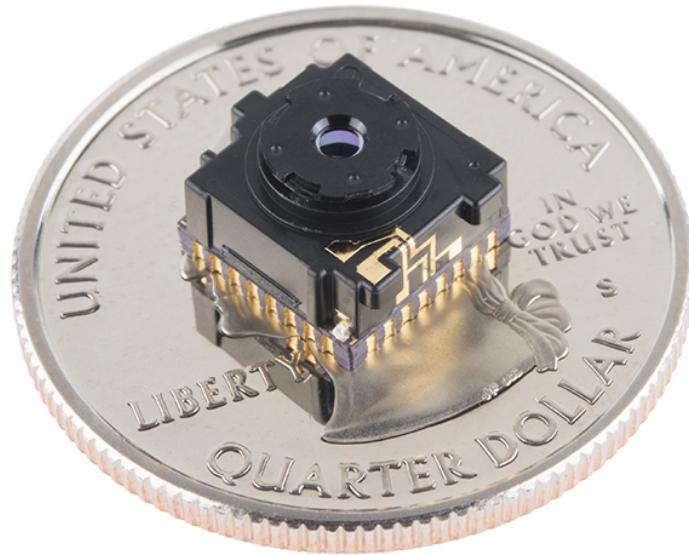
Rys. 2.2. Emisyjność ciała idealnie czarnego.

try. Mikrobolometr jest czujnikiem do pomiaru energii przenoszonej promieniowaniem elektromagnetycznym.

Firma FLIR wykorzystuje tlenek wanadu do budowy mikrobolometrów m.in. w kamerach Lepton. Tlenek wanadu cechuje się dużym temperaturowym współczynnikiem rezystancji (TWR) oraz małym szumem  $1/f$ , co zapewnia doskonałą czułość oraz jednolitość. W celu uzy-skania obrazu, zespół soczewek skupia promieniowanie z rejestrowanej sceny na macierz de-tektorów. W każdym z detektorów, w odpowiedzi na padającą na niego wiązkę promieniowania, zmienia się temperatura zawartego w nim tlenku wanadu. Zmiana temperatury wiąże się pro-porcjonalnie ze zmianą rezystancji. Rejestracja sceny polega na odczycie rezystancji każdego de-tektora poprzez przyłożenia napięcia i odczyt przepływającego przez nie prądu [9].

### 2.2.2. Kamera termowizyjna FLIR Lepton

Lepton jest miniaturową kamerą termowizyjną. W pojedynczym układzie został zintegro-wany kompletny system składający się soczewki, sensora podczerwieni fal długich (ang. LWIR – *Long Wave Infrared*) oraz elektroniki sterującej i przetwarzającej sygnał. Kamera cechuje się bardzo małymi wymiarami, co czyni ją idealnym rozwiązaniem do zastosowań mobilnych. Układ ma możliwość domontowania dodatkowej przesłony, która jest wykorzystywana do auto-matycznej optymalizacji procesu ujednolicania obrazu (kalibracji sensora). Układ jest prosty do integracji z dowolnym mikrokontrolerem dzięki zastosowaniu standardowych protokołów i in-



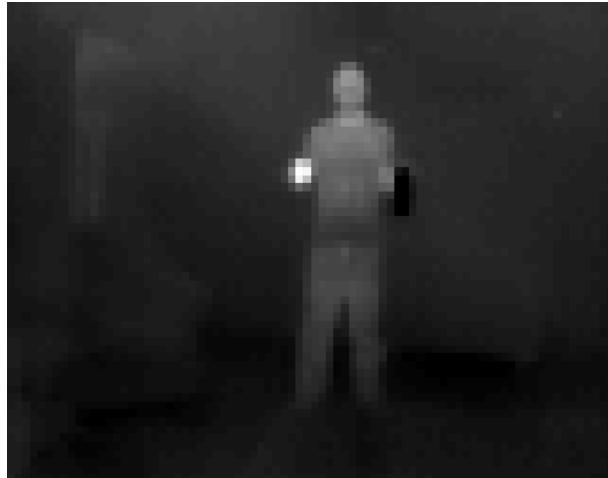
Rys. 2.3. Widok poglądowy na kamerę FLIR Lepton.

terfejsów. Na rysunku /reffig:leptonTermalImage zaprezentowano przykładowy obraz usyskany za pomocą kamery.

Lepton po podłączeniu zasilania od razu uruchamia się w domyślnym trybie pracy. Kamera jest konfigurowalna poprzez CCI (ang. *Camera Control Interface* – interfejs kontroli kamery). Zapewnia on dostęp do rejestrów zawierających konfigurację [9].

Parametry kamery:

- Wymiary: 11,8 x 12,7 x 7,2 mm,
- Sensor: niechłodzony mikrobolometr VOx (tlenek wanadu),
- Rejestrowany zakres: fale długie podczerwieni,  $8\mu m$  do  $14\mu m$ ,
- Wielkość piksela:  $17 \mu m$ ,
- Rozdzielcość: 80x60 pikseli,
- Liczba klatek na sekundę: 8,6,
- Zakres rejestrowanych temperatur:  $-10^{\circ}C$   $140^{\circ}C$  (tryb wysokiego wzmacnienia),
- Korekta niejednorodności matrycy: automatyczna na bazie przepływu optycznego,



**Rys. 2.4.** Obraz człowieka w termowizji wykonany kamerą Lepton. W prawej ręce widać gorący obiekt (kubek herbaty), w lewej zimny (butelka wody z lodówką).

- Kąt widzenia horyzontalny / diagonalny:  $51^{\circ}$   $66^{\circ}$ ,
- Głębia ostrości: od 10 cm do nieskończoności,
- Format wyjściowy: do wyboru: 14-bit, 8-bit (z AGC (ang. *automatic gain control* – automatyczna kontrola wzmacnienia)) 24-bit RGB (z ACG i koloryzacją),
- Interfejs wideo: VoSPI (Video over Serial Peripherial Interface),
- Interfejs sterujący: CCI (zbliżony do I2C).

## 2.3. Rejestracja obrazu multispektralnego

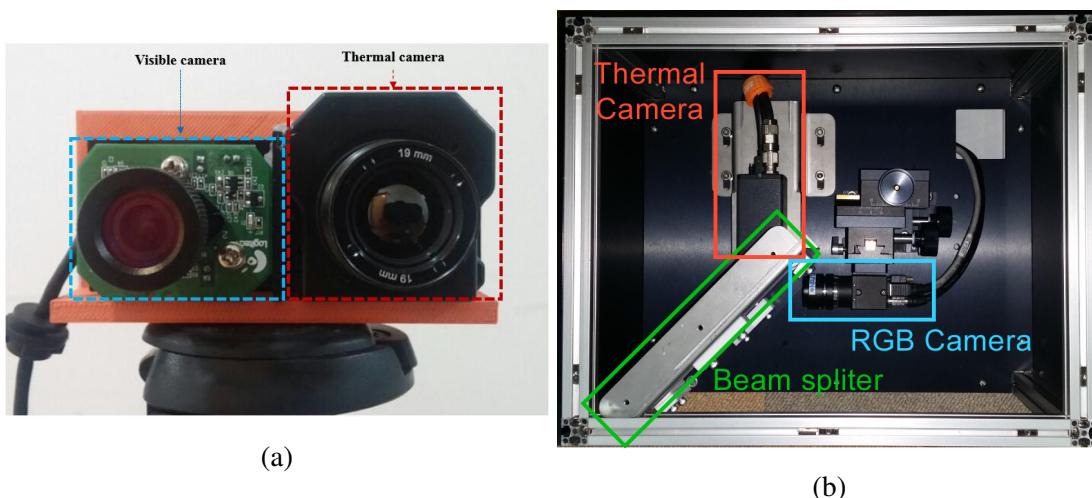
Widmo elektromagnetyczne docierające do kamery składa się fal o różnych długościach. Sensory w kamerach rejestrują obraz tylko w pewnym zakresie tego widma, więc aby uzyskać obraz w wymaganym paśmie należy odfiltrować niepożądane elementy widma, np. kolorowy obraz z kamery wizyjnej jest otrzymywany poprzez zastosowanie trzech filtrów: czerwonego, zielonego i niebieskiego. Ponieważ wszystkie trzy kolory mogą być zarejestrowane przez pojedynczą matrycę, filtry są nałożone bezpośrednio na sensor, a wartość koloru w danym punkcie jest interpolowana z sąsiadujących ze sobą pikseli (tzw. matryca Bayera). W przypadku, gdy nie jest możliwe zastosowanie jednego sensora do wszystkich pożądanych zakresów, należy rozdzielić wiązkę pomiędzy kilka różnych matryc aparatu, albo wykorzystać równoległy układ kamer.

W przypadku jednoczesnej rejestracji obrazu wizyjnego i termicznego większość rozwiązań wykorzystuje układ dwóch równoległych do siebie kamer – przykład przedstawia rysunek 2.5a. W tym przypadku została zastosowana kamera termowizyjna FLIR Tao 2 oraz kamera wizyjna Logitech Webcam c600.

Zazwyczaj obrazy z kamer różnią się, co wynika z ich budowy, różnej rozdzielczości, kąta widzenia oraz zniekształceń soczewkowych. Do poprawnego odwzorowania tej samej sceny w obu widmach należy zastosować algorytm mający na celu dopasowanie obu obrazów. Tworzony jest w ten sposób nowy obraz, na którym wszystkie piksele łączą informacje o kolorze i temperaturze.

Pierwszym z etapów poprawnego dopasowania obrazów jest kalibracja systemu wizyjnego. Wykonuje się ją z wykorzystaniem specjalnych plansz, które pozwalają określić położenie pewnych punktów w przestrzeni w obu rejestrowanych zakresach promieniowania. Punkty te pozwalają na obliczenie relacji między obrazami. Plansze mogą być aktywne (posiadają własne źródło ciepła) albo pasywne (przesłaniają obce źródło ciepła). W równoległym układzie kamer występuje również zjawisko paralaksy, które powiększa się wraz ze wzrostem odległości obiektu od punktu kalibracji. Ogranicza to zakres w jakim może działać taki system.

W pracy [4] autorzy zastosowali zwierciadło półprzezroczyste wykonane z wafla krzemowego pokrytego cynkiem do rozdzielenia obrazu wizyjnego od termicznego (rysunek 2.5b). Wykorzystując trójosiowy uchwyt, kamery zostały ustawione tak, by ich osie optyczne się pokrywały. Następnie obrazy z obu kamer zostały zrektyfikowane, aby miały tą samą wirtualną ogniskową.



**Rys. 2.5.** Sposoby akwizycji obrazów: (a) dwie kamery równolegle [2], (b) z wykorzystaniem zwierciadła półprzezroczystego [4].

### 2.3.1. Model geometryczny

Do opisu matematycznego systemu wykorzystuje się model kamery otworkowej. Dzięki niemu można opisać relację między trójwymiarową przestrzenią a dwuwymiarowym obrazem za pomocą projekcji perspektywicznej. Nie stanowi on najdokładniejszego opisu matematycznego kamery, nie ma w nim uwzględnionych zakłóceń soczewkowych, jednakże zapewnia dobre rezultaty w większości aplikacji. Model składa się z 2 zestawów parametrów: zewnętrznych oraz wewnętrznych. Parametry zewnętrzne definiują lokację kamery względem zewnętrznego układu współrzędnych. Są reprezentowane przez wektor translacji  $T$  między układem związanym z kamerą  $(X_c, Y_c, Z_c)$  a zewnętrznym  $(X, Y, Z)$ . Drugim parametrem jest macierz rotacji  $R$  (między osiami tych dwóch układów). Punkt  $P = [X, Y, Z]^T$  będący w zewnętrznym układzie współrzędnym ma swój odpowiednik w układzie wewnętrznym, który można określić zależnością:

$$P_c = RP + T \quad (2.1)$$

Właściwości optyczne kamery można przedstawić w postaci macierzy kamery:

$$K = \begin{bmatrix} f_x & 0 & x_0 \\ 0 & f_y & y_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.2)$$

gdzie:

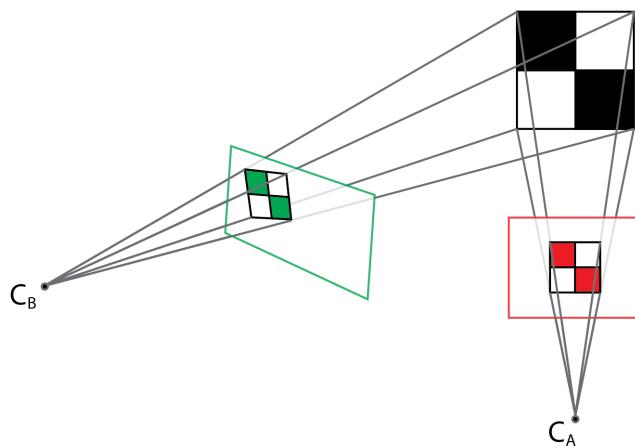
- $f_x, f_y$  = ogniskowa kamery wyrażona w liczbie pikseli,
- $x_0, y_0$  = współrzędne punktu głównego.

Macierz  $K$  określa związek między znormalizowanymi współrzędnymi w układzie odniesienia kamery danych wzorem  $x_n = \frac{X_c}{Z_c}, y_n = \frac{Y_c}{Z_c}$  a odpowiadającym im współrzędnymi punktów na obrazie  $u, v$ :

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = K \begin{bmatrix} x_n \\ y_n \\ 1 \end{bmatrix} \quad (2.3)$$

### 2.3.2. Kalibracja

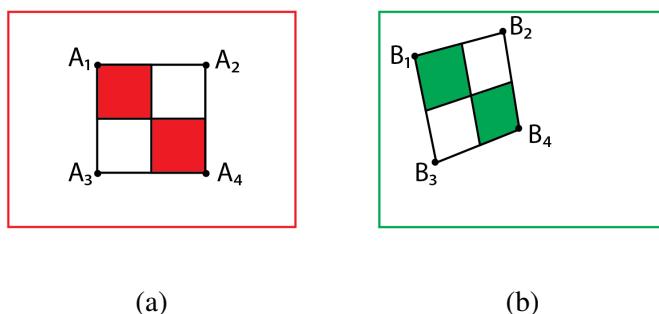
Obrazy, które przedstawiają tę samą scenę, ale zostały wykonane dwoma różnymi kamerami w innych położeniach różnią się. Na rysunku 2.6 czarna szachownica jest uchwycona



**Rys. 2.6.** Dwie kamery rejestrujące jeden obiekt.

przez dwie kamery ustawione w punktach  $C_A$  (na wprost obiektu) oraz  $C_B$  (po skosie i lekko obrócona).

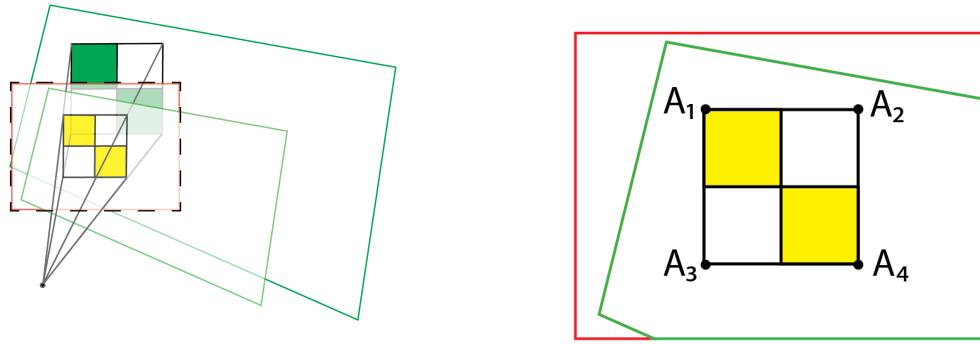
Aby dopasować te dwa obrazy, tak by szachownice były ujęte tak samo (tj. w tym samym miejscu na obrazie), należy na jednym z nich przeprowadzić transformację projekcyjną. Jest to przekształcenie pomiędzy dwoma płaszczyznami, które wykorzystuje model geometryczny kamery. Wymaga to uprzedniego wyznaczenia macierzy transformacji  $A$  na podstawie co najmniej 4 punktów kalibracyjnych.



**Rys. 2.7.** Zarejestrowane obrazy: (a) przez kamerę  $C_A$ , (b) przez kamerę  $C_B$ .

Punkty  $A$  i  $B$  są punktami kalibracyjnymi.

Na rysunkach 2.7a i 2.7b punkty  $A_1$  do  $A_4$ , będące czterema rogami zarejestrowanej szachownicy przez kamerę  $C_A$ , odpowiadają punktom  $B_1$  do  $B_4$ , będącymi tymi samymi czterema rogami zarejestrowanymi kamerą  $C_B$ . Macierz transformacji  $A$  można obliczyć rozwiązując



(a) Interpretacja transformacji projekcyjnej: rzutowanie płaszczyzny.

(b) Wynik transformacji.

**Rys. 2.8.** Transformacja projekcyjna.

równanie (2.4):

$$A = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & 1 \end{bmatrix} \quad (2.4)$$

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ \dots \\ u_n \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ \dots \\ v_n \end{bmatrix} = \begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -u_1x_1 & -u_1y_1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 & -u_2x_2 & -u_2y_2 \\ x_3 & y_3 & 1 & 0 & 0 & 0 & -u_3x_3 & -u_3y_3 \\ x_4 & y_4 & 1 & 0 & 0 & 0 & -u_4x_4 & -u_4y_4 \\ \dots & \dots & & & & & & \\ x_n & y_n & 1 & 0 & 0 & 0 & -u_nx_n & -u_ny_n \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -v_1x_1 & -v_1y_1 \\ 0 & 0 & 0 & x_2 & y_2 & 1 & -v_2x_2 & -v_2y_2 \\ 0 & 0 & 0 & x_3 & y_3 & 1 & -v_3x_3 & -v_3y_3 \\ 0 & 0 & 0 & x_4 & y_4 & 1 & -v_4x_4 & -v_4y_4 \\ \dots & \dots & & & & & & \\ 0 & 0 & 0 & x_n & y_n & 1 & -v_nx_n & -v_ny_n \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \end{bmatrix} \quad (2.5)$$

$u_n, v_n$  = współrzędne punktu kalibracji  $n$ , na obrazie bazowym

$x_n, y_n$  = współrzędne punktu kalibracji  $n$ , na obrazie dopasowywanym

Transformację projekcyjną można zinterpretować jako rzutowanie płaszczyzny, co obrazuje rysunek 2.8a. Wynikiem transformacji (a zarazem rzutowania) jest obraz dopasowany do obrazu bazowego (2.8b)



## 3. Algorytmy detekcji pieszych

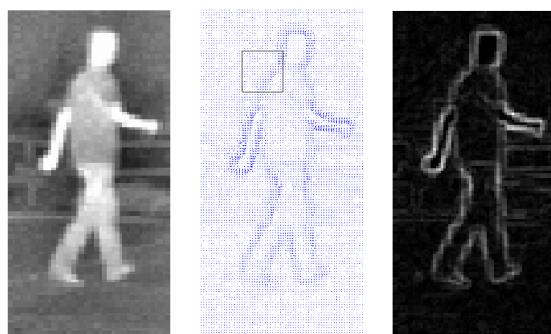
W cyfrowej analizie obrazu detekcja pieszych jest jedną z najbardziej aktywnie rozwijanych dziedzin. W ciągu kilkudziesięciu lat powstało ponad tysiąc artykułów poruszających to zagadnienie [10], w których zaproponowano wiele różnych metod. Większość z nich opiera się na analizie obrazu tylko w jednym spektrum: widzialnym albo podczerwieni. Praca [4] pokazała, iż połączenie obu obrazów może dać lepsze wyniki. Podobnie w artykule [11] wykazano, że analiza multispektralna jest skuteczniejsza w dzień niż w nocy (o około 5% AMR (ang. *Avrange Miss Rate*)). W artykule [12] autorzy podsumowują osiągnięcia w dziedzinie detekcji pieszych w latach 2004 – 2014. Wyróżniono ponad 40 różnych podejść do problemu. Eksperymenty w artykule są oparte o bazę danych Caltech-USA, która zawiera obrazy w kolorze. Jednym z wniosków jest to, że przez ostatnie dziesięć lat największy postęp został osiągnięty głównie dzięki dopracowaniu cech, które są wyodrębniane z obrazu, niż ulepszanie klasyfikatora. Dodatkowo autorzy połączycyli cechy dające najlepsze wyniki i stworzyli własną metodę, która pozwoliła poprawić o ok 12% AMR względem najlepszej badanej wcześniej metody.

Coraz większą popularnością cieszą się rozwiązania oparte o głębokie sieci neuronowe (DNN ang. *Deep Neural Networks*). W artykule [28] został zaprezentowany system wykrywania pieszych operaty na fuzji kilkunastu dobrze wyuczonych sieci neuronowych, dając jeden z lepszych wyników detekcji z wykorzystaniem bazy Caltech-USA.

Dla typowego algorytmu detekcji pieszych można wyróżnić trzy podstawowe etapy:

### 3.1. Ustalenie regionu zainteresowania

Jest to obszar zwany ROI (ang. *Region Of Interest*), w którym potencjalnie mogą znajdować się plesi. Wiele podejść uznaje cały obraz jako ROI i stosuje okno przesuwne, sprawdzając każdy możliwy fragment obrazu. Jeżeli scena jest rejestrowana przez nieruchomą kamerę, ROI można określić poprzez różnicę między zapamiętanym tłem a aktualnym obrazem (tzn. modelowanie i odejmowanie tła). Analiza przepływu optycznego również pozwala na wyodrębnienie obszaru, w których obserwowany jest ruch inny niż na pozostałej części sceny. Inną metodą



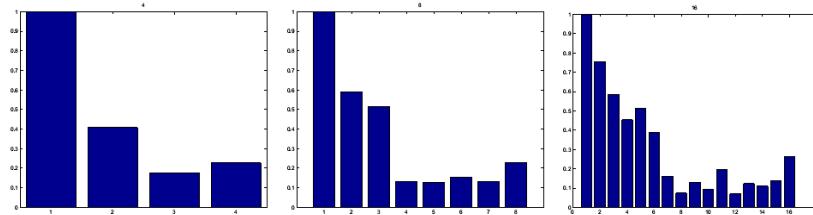
**Rys. 3.1.** Obliczanie gradientów dla obrazu. Po lewej oryginalny obraz, pośrodku kierunki gradientów, po prawej wartości modułów [14].

jest zastosowanie słabszego, bardziej ogólnego, ale mniej wymagającego obliczeniowo klasyfikatora. Wyodrębnienie ROI jest bardzo istotne w przypadku pracy w czasie rzeczywistym, ze względu na ograniczony maksymalny czas analizy pojedynczego obrazu.

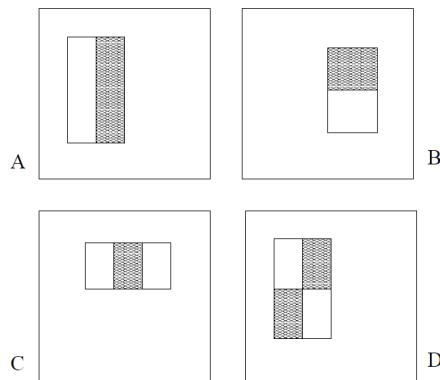
## 3.2. Wyodrębnienie cech

Do najbardziej popularnych cech można zaliczyć:

1. Histogramy zorientowanych gradientów (HOG ang. *Histogram of Oriented Gradients*). Algorytm został zaproponowany przez N.Dalala i B. Triggsa w pracy [13] i stał się jedną z najbardziej popularnych metod w dziedzinie detekcji ludzi. Jest cały czas rozwijany i modyfikowany w wielu pracach naukowych. W pierwszej kolejności zostają obliczone gradienty dla obrazu, za pomocą dwóch masek kierunkowych  $[-1 \ 0 \ 1]$  i  $[-1 \ 0 \ 1]^T$ . Gradienty mogą być ze znakiem lub bez. Na rysunku 3.1 przedstawiono przykład uzyskiwania gradientów. Następnie tworzone są histogramy orientacji na podstawie gradientów uzyskanych z pkt. 1. Histogram ma z góry zadaną liczbę przedziałów i zawiera dane z komórki (ang. *cell*) o określonej wielkości (np. kwadrat  $6 \times 6$ ). Wagi orientacji poszczególnych pikseli wynikają z wartości wypadkowej gradientu. Na rysunku 3.2 zaprezentowano przykładowe histogramy. Komórki łączone są w bloki, w obrębie których następuje normalizacja. Ma ona na celu wyrównanie kontrastów pomiędzy sąsiadującymi blokami komórek. Bloki mają z góry ustalony rozmiar i nakładają się na siebie. Połączenie wszystkich histogramów we wszystkich blokach w jeden wektor cech.
2. Lokalne wzorce binarne LBP (ang. *Local Binary Patterns*). Oryginalnie deskryptory te zostały zaproponowane do opisu tekstur [15]. Analizowany obraz zostaje podzielony na bloki. Następnie, do każdego piksela w bloku zostaje przypisany wzorzec binarny na



Rys. 3.2. Przykłady histogramów o różnej ilości przedziałów [14].

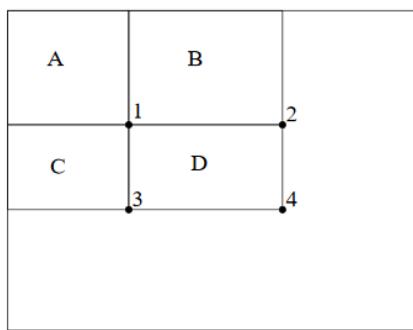


Rys. 3.3. Przykładowe prostokąty służące do określenia cech. Od sumy pikseli zawartych w białym prostokącie zostaje odjęta suma pikseli w szarym. [16].

podstawie wartości pikseli w jego sąsiedztwie. Jeżeli wartość sąsiadującego piksela jest większa od centralnego, to przyjmuje on wartość 1. W ten sposób do każdego piksela przypisywany jest wzorzec binarny (np. 1100110). Następnie zostaje obliczony histogram wzorców dla każdego bloku. Histogramy z wszystkich bloków wchodzących w skład obrazu (okna detekcji) tworzą wektor cech.

3. Falki Haara. Określają różnicę w kontraste między dwoma przylegającymi prostokątnymi obszarami. W oryginalnej pracy P.Viola i M.Jones z 2001 [16] autorzy rozważali 3 rodzaje cech:

- dwa obszary mające ten sam rozmiar i kształt oraz przylegające do siebie horyzontalnie bądź wertykalnie, gdzie cechę stanowi różnica sumy pikseli zawartych w każdym z regionów (Układ A i B na rysunku 3.3),
- obszar składający się z 3 prostokątów przylegających do siebie, gdzie od sumy środkowego elementu jest odejmowana suma dwóch zewnętrznych (układ C),
- układ 4 prostokątów, gdzie suma jest różnicą między obszarami po przekątnej (układ D).



Rys. 3.4. Obliczanie sumy pikseli za pomocą *integral image*. [16].

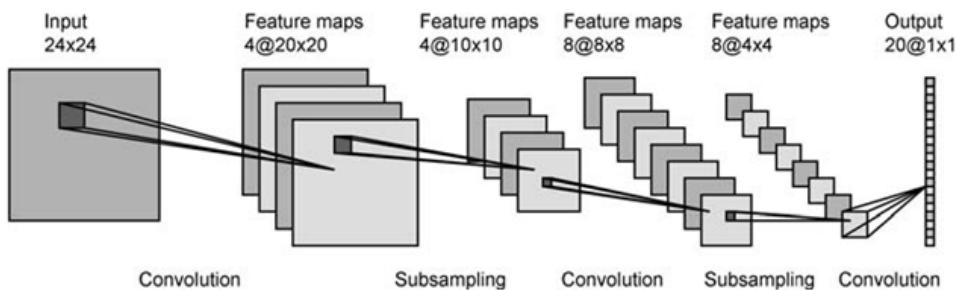
Cechy są łatwe do skalowania i nie wymagają dużych nakładów obliczeniowych. W celu szybkiego obliczania tych cech stosuje się tzw. *integral image*. Dla każdego punktu o współrzędnych  $x, y$  zostaje obliczona suma wartości pikseli znajdujących się na lewo i nad pozycją  $x, y$ . Dzięki temu, aby obliczyć sumę pikseli w danym kwadracie wystarczą jedynie cztery punkty referencyjne. Na przykład, by obliczyć sumę pikseli w prostokącie D na rysunku 3.4 wystarczy rozwiązać równanie  $D = 4 + 1 - (2 + 3)$ .

4. Kolor. W analizie obrazów wykorzystuje różne przestrzenie barw np. RGB, HSV oraz LUV. Głównie wtedy, gdy kolor wykrywanego obiektu jest kluczowy (np. znaki drogowe, światła na skrzyżowaniu). Jako cechę można go wykorzystać w kilku formach. Momenty koloru (ang. *Color Moments*) jest to średnia, wariancja i odchylenie standardowe występowania danego koloru na obrazie. Histogram określaczęstość występowania danego koloru, a wektor koherencji koloru (CCV ang. *Color Coherence Vectors*) określa w jakim stopniu piksele danego koloru są częścią obszaru o podobnym kolorze (np. obraz zielonej łąki na którym pasie się jedna fioletowa krowa. Kolor zielony na obrazie byłby rozłożony równomiernie natomiast fioletowy byłby skupiony w pojedynczym rejonie koherencji – krowy) [17].

### 3.3. Klasyfikator

Otrzymany wektor cech jest następnie poddany klasyfikacji, której wynik decyduje czy obraz zawiera człowieka. W pracy [12] autorzy wyróżnili 3 dominujące rodziny metod:

1. Rodzina DPM (ang. *Deformable Part Model*) Technika zakłada, że obiekty mogą być zamodelowane poprzez części ułożone w deformowanych konfiguracjach. Model składa się z głównego, globalnego filtra, który stanowi punkt odniesienia dla pozostałych części.



Rys. 3.5. Schemat przykładowej sieci konwolucyjnej.

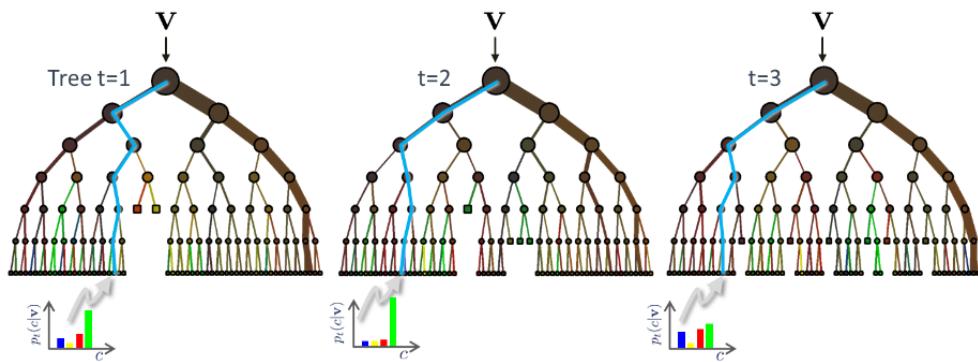
Każda część zawiera swój własny filtr wraz z zestawem dozwolonych pozycji względem okna detekcyjnego oraz koszt deformacji dla każdej z tych pozycji. Suma wyniku uzyskanego z filtra głównego wraz z jego częściami stanowi o wyniku detekcji [18].

## 2. Deep networks.

Głębokie sieci neuronowe posiadają kilkanaście warstw ukrytych między warstwą wejściową i wyjściową. Ich działanie polega na tym, że po podaniu wektora cech na warstwę wejściową wytrenowanej sieci, w warstwie wyjściowej aktywuje się neutron odpowiedzialny za detekcję danej klasy. W analizie obrazu szczególnie chętnie wykorzystywane są sieci konwolucyjne. W odróżnieniu od klasycznego podejścia głębokie sieci konwolucyjne łączą w sobie oba zadania: wyodrębnianie cech i klasyfikacje. Neurony pierwszej warstwy ukrytej są podłączone jedynie do wybranego fragmentu warstwy wejściowej (np. okna 24x24). Jest to tzw. warstwa konwolucyjna. Neurony w tej warstwie dzielą wspólne wagi dla swoich wejść i bias. Sieć posiada zazwyczaj kilkanaście takich warstw – każda wykrywająca pojedynczą cechę. Pozwala to na redukcję liczby neutronów i parametrów potrzebnych do uzyskania w procesie uczenia. Do warstw konwolucyjnych dochodzą warstwy sumujące (ang. *Polling Layers*). Ich zadaniem jest generalizacja informacji z poprzedniej warstwy. Sieć zamyka warstwa wyjściowa. Na rysunku 3.5 przedstawiono schemat przykładowej sieci konwolucyjnej.

## 3. Decision forests

Lasy decyzyjne to zbiory nieskorelowanych drzew decyzyjnych. Pojedyncze drzewo jest graficznym odwzorowaniem procesu decyzyjnego. Na podstawie wektora cech drzewo zwraca rozkład prawdopodobieństwa przynależności obiektu do klas. Wynikiem ostatecznym jest średnia prawdopodobieństw z wszystkich drzew w lesie. Rysunek 3.6 przedstawia schemat koncepcyjny dla lasu o trzech drzewach. Algorytm uczenia drzew wykorzystuje przykłady (wektor cech) i związane z nimi konsekwencje (klasyfikacja obiektu).

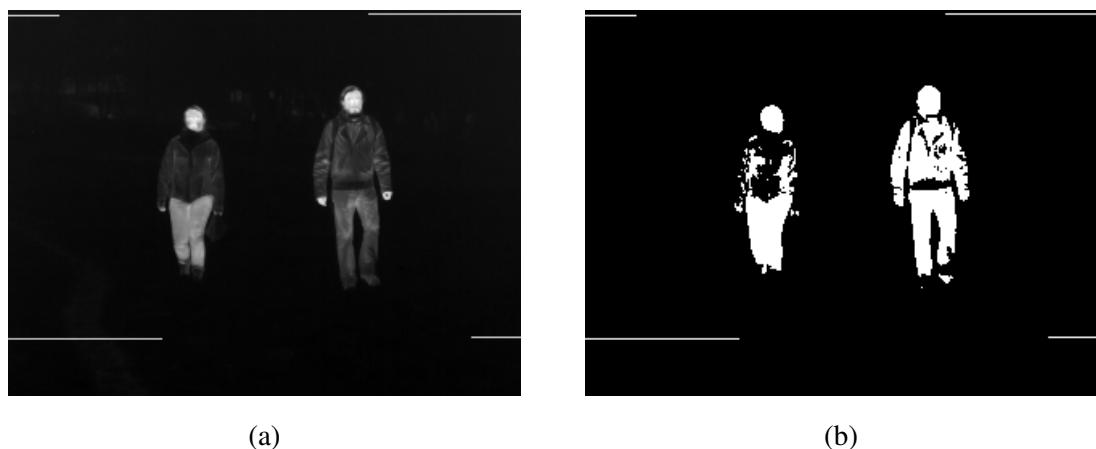


Rys. 3.6. Schemat lasu decyzyjnego składającego się z trzech drzew [19].

4. inne: np. SVM (ang. Support Vector Machine – maszyna wektorów nośnych), AdaBoost itp.

### 3.4. Specyfikacja detekcji w podczerwieni

Człowiek jest istotą żyjącą, co za tym idzie, wytwarza ciepło. Przeważnie ciało człowieka ma wyższą temperaturę niż otoczenie, co jest wyraźnie widoczne na obrazie termowizyjnym. Termowizja najlepiej sprawdza się w nocy, kiedy kontrast między człowiekiem a otoczeniem jest największy. Ta cecha powoduje, że podczerwień jest chętnie wykorzystywana do systemów wspomagania kierowcy, aby zwiększyć bezpieczeństwo poruszania się po drodze w nocy. Sylwetka przechodnia może zostać w bardzo prosto wyodrębniona, wykonując prostą operację binaryzacji. Rysunek 3.7 przedstawia przykładowy proces binaryzacji. Daje to możliwość zastosowania wielu algorytmów, które opierają się na kształcie obiektu (np. wzorzec probabilistyczny albo obliczenie proporcji wysokości do szerokości).



(a)

(b)

**Rys. 3.7.** Binaryzacja: (a) Obraz zarejestrowany kamerą termowizyjną, (b) Obraz po binaryzacji[20].



## 4. Wykorzystanie układów FPGA i Zynq w przetwarzaniu i analizie obrazu

Tradycyjne systemy wizyjne zwykle bazują na architekturze sekwencyjnej. W tym rozwiązańiu obraz jest sukcesywnie poddawany kolejnym przekształceniom, a wyniki pośrednie zapisywane są w pamięci operacyjnej lub podręcznej. W aplikacji procesorowej operacje te są wykonywane przez układ arytmetyczno-logiczny. Kolejne kroki algorytmu są komplikowane w ciąg instrukcji dla procesora, który oprócz operacji matematycznych dużą część pracy poświęca na pobieranie i dekodowanie rozkazów oraz na odczytywanie i zapisywanie danych do pamięci. By taka aplikacja mogła pracować w czasie rzeczywistym, cała procedura musi wykonać się szybciej niż przychodzące dane wizyjne, co wymusza wysokie taktowanie procesora sięgające kilku GHz. To podejście jednak ma swoje ograniczenia. Wraz ze wzrostem częstotliwości pracy procesora wzrasta jego moc a tym samym ilość ciepła, które która musi zostać rozproszona. Najszybsze CPU są taktowane z częstotliwością zegara sięgającą nawet 4,4 GHz (choć przy zastosowaniu chłodzenia ciekłym azotem jest możliwe uzyskanie ponad 8 GHz). Dodatkowo, efektywność energetyczna takiego rozwiązania pozostawia wiele do życzenia. Wzrost szybkości obliczeń uzyskuje się coraz częściej poprzez zwiększanie liczby rdzeni w procesorze.

W przypadku podejścia równoległego, implementacja poszczególnych kroków algorytmu odbywa się w osobnych procesach. Jeżeli wykonywany algorytm jest głównie sekwencyjny, tzn. kolejne kroki algorytmu wymagałyby danych otrzymanych z poprzednich, to zysk takiego zabiegu byłby równy zero. W celu uzyskania dobrej implementacji w układzie równoległym istotne jest by znaczna część algorytmu mogła być wykonywana równolegle. Maksymalne do uzyskania przyspieszenie jest określone przez prawo Amdahla:

$$P_w = \frac{1}{s + \frac{1-s}{n_w}} \quad (4.1)$$

gdzie:

$P_w$  = przyspieszenie algorytmu w systemie wieloprocesorowym,

$s$  = część algorytmu niepodlegająca zrównolegleniu (wartość od zera do jeden),

$n_w$  = liczba elementów obliczeniowych.

Algorytmy przetwarzania obrazów są w dużej mierze równoległe, szczególnie te niskiego i średniego poziomu. W wielu przypadkach, każdy piksel obrazu można przetwarzać niezależnie, np. we wszelkich operacjach kontekstowych, przekształceniach przestrzeni barw, binaryzacji itp. Między innymi dlatego układy FPGA są chętnie stosowane w systemach wizyjnych. Teoretycznie jedynym ograniczaniem w możliwości zrównoleglenia obliczeń jest liczba dostępnych zasobów w układzie, jednak innym istotnym aspektem jest sposób dostarczania danych do modułów obliczeniowych. Dostęp do pamięci często wymaga czasu, a ilość danych przekazana podczas jednego transferu jest ograniczona. Stanowi to wąskie gardło w tego rodzaju rozwiązaniach. Z tego powodu przetwarzanie danych odebranych bezpośrednio z czujnika wizyjnego w czasie jego akwizycji jest chętnie wykorzystywane, gdyż zmniejsza to liczbę operacji odczytu i zapisu [6].

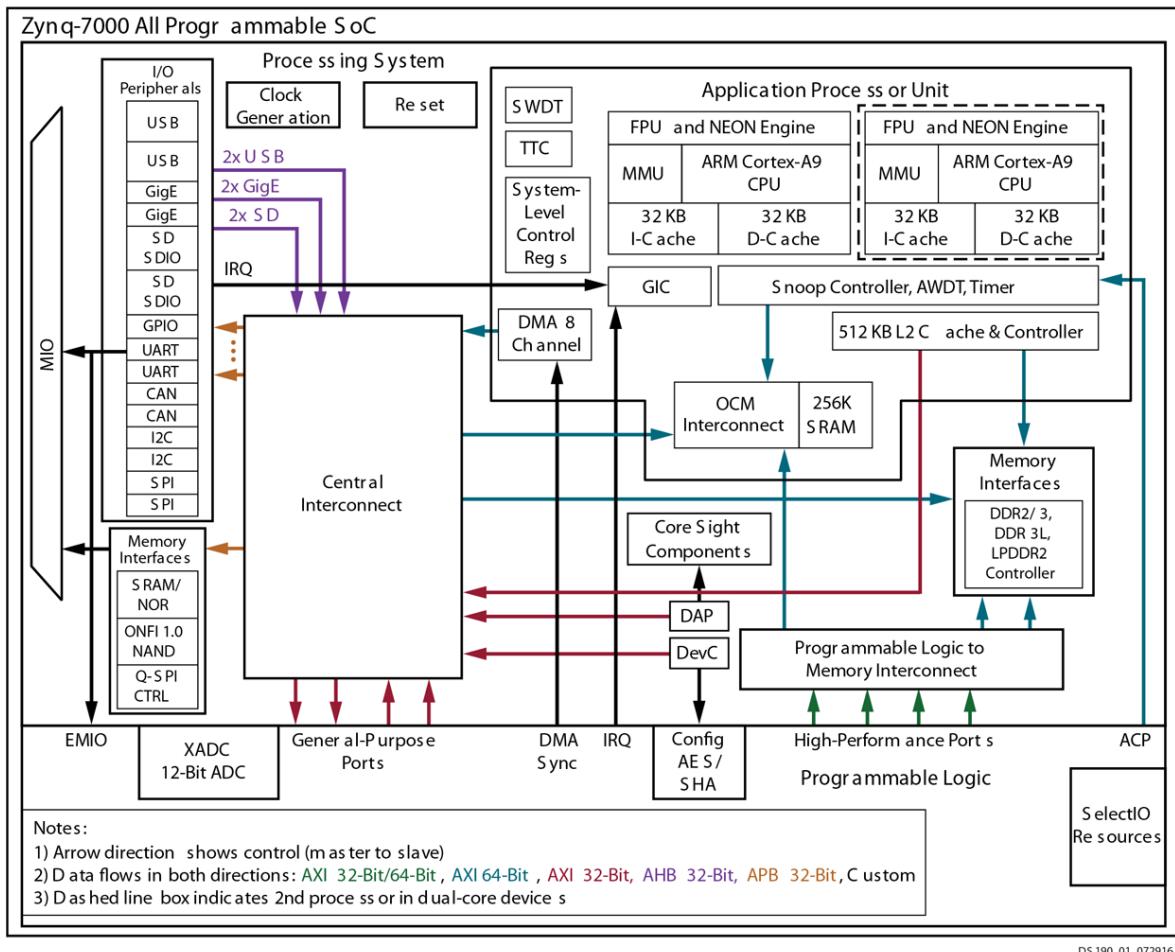
Układy GPU (ang. *Graphics Processing Unit* też dają możliwość zrównoleglania obliczeń. W odróżnieniu od FPGA mają sztywną strukturę. Swoją architekturę przypominają rozwiązania SIMD (ang. *Single Instruction Multiple Data*) z tą różnicą, że zamiast pojedynczego rozkazu wykonywany jest ten sam kod (tzw. *kernel*) na dużej liczbie mniejszych procesorów.

## 4.1. Układ Zynq-7000

Rodzina układów Zynq-7000 bazuje na architekturze SoC (ang. *System on Chip*). W pojedynczym układzie scalonym został zawarty kompletny system, w skład którego wchodzą układy spełniające różne funkcje. Został on podzielony na dwie główne części: system procesorowy (PS ang. *Porcessing System*) bazujący na procesorze ARM Cortex-A9 oraz logikę programowalną (PL ang. *Programable Logic*) – FPGA. Na rysunku 4.1 przedstawiono schemat architektury układu.

Część procesorowa, oprócz samego ARM-a, posiada wbudowaną pamięć, kontroler pamięci zewnętrznej oraz szereg interfejsów dla układów peryferyjnych takich jak USB, GigEthernet, CAN, I2C, SPI. W części logiki programowej znajdują się bloki logiki konfigurowalnej (CLB ang. *configurable logic block*), 36Kb bloki pamięci RAM, moduły DSP48, układ JTAG, układy zarządzania zegarami oraz dwa 12-bitowe przetworniki analogowo-cyfrowe.

Komunikacja między częścią procesorową a logiką programową odbywa się za pośrednictwem interfejsu AXI (ang. *Advanced Extensible Interface*), bezpośrednio wykorzystując porty ogólnego przeznaczenia oraz przerwania.



Rys. 4.1. Schemat ogólny architektury układu Zynq-7000.

## 4.2. Interfejs AXI

AXI (ang. *Advanced eXtensible Interface* – zaawansowany rozszerzalny interfejs) jest częścią ARM AMBA (ang. *Advanced Microcontroller Bus Architecture*) – otwartego standardu, będącego specyfikacją do zarządzania połączeniami między blokami funkcjnymi w SoC. Aktualnie jest stosowana AMBA 4.0 która wprowadziła drugą wersję AXI – AXI4. Występują trzy typy interfejsów dla AXI4:

- AXI4 – stosowany w wysokowydajnych transferach w przestrzeni pamięci (ang. *memory-mapped*),
- AXI4-Lite – stosowany dla prostszych operacji w przestrzeni pamięci (na przykład do komunikacji z rejestrami kontrolnymi i statusu),
- AXI4-Stream – stosowany do transmisji strumieniowych (wysokiej prędkości).

Specyfikacja interfejsu zakłada komunikację pomiędzy pojedynczym AXI *master* i pojedynczym AXI *slave*, która ma na celu wymianę informacji. Kilkanaście interfejsów AXI *master* i *slave* mogą zostać połączone między sobą za pomocą specjalnej struktury zwanej *interconnect block* (blok międzymodułowy), w której odbywa się trasowanie połączeń do poszczególnych bloków.

AXI4 i AXI4-Lite składają się z 5 różnych kanałów:

- Kanał adresu odczytu,
- Kanał adresu zapisu,
- Kanał danych odczytyanych,
- Kanał danych do zapisania,
- Kanał potwierdzenia zapisu.

Dane mogą płynąć w obie strony pomiędzy *master* a *slave* jednocześnie. Ilość danych, które można przesyłać w jednej transakcji w przypadku AXI4 wynosi 256 transferów, zaś AXI4-Lite pozwala na tylko 1 transmisję.

AXI4-Stream nie posiada pola adresowego, a dane mogą być przesyłane nieprzerwanie.

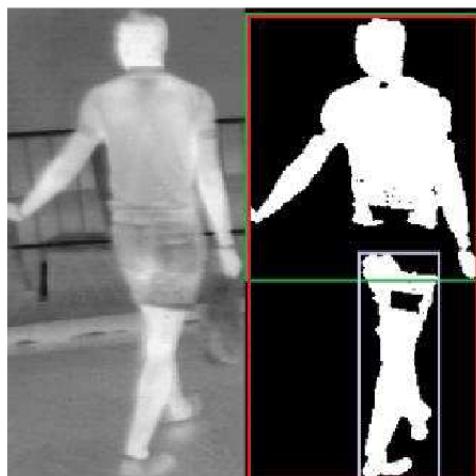
## **5. Przegląd metod detekcji pieszych**

### **5.1. Zaawansowana binaryzacja i segmentacja + HOG SVM**

W pracy [21] autorzy opracowali algorytm pozwalający na szybką i efektywną detekcję przechodniów w czasie rzeczywistym. Termowizja pozwala na uzyskanie dobrego kontrastu między poszukiwanym przechodniem a otoczeniem. Zaproponowany system jest dedykowany do pracy w nocy, kiedy kontrast między człowiekiem a otoczeniem pozwala na jednoznaczne ich rozróżnienie. Rozwiążanie bazuje na ulepszonym algorytmie progowania i segmentacji obrazu.

Pierwszym etapem jest wyodrębnienie obszarów zainteresowań (ROI). Pozwala to na znaczne ograniczenie obszaru analizowanych fragmentów obrazu. Obraz w odcieniach szarości zostaje poddany binaryzacji z użyciem dwóch progów: mniejszym i większym. Pozwala to na detekcję przechodniów w różnych rejonach obrazu o różnym kontraście. Progi zmieniają się wraz z dynamiką obrazu wejściowego. W obrazie termicznym człowieka często w okolicy bioder znajduje się chłodniejszy obszar, który jest poniżej progu binaryzacji, co skutkuje przerwę między dwoma połówkami człowieka. Aby połączyć je w jedną całość, dla każdego obszaru wyłonionego podczas binaryzacji zostają wytypowane dodatkowe ROI, które obejmuje ten obszar wraz z innymi znajdującymi się nad lub pod. Przykładowy wynik segmentacji jest zaprezentowany na rysunku 5.1.

Następnym krokiem jest filtracja wyników. Ma ona na celu zredukowanie liczby obszarów przed końcową analizą. Autorzy zastosowali filtrację opierającą się na proporcji obszaru zainteresowań. Pozytywnie zakwalifikowane zostały jedynie obszary o odpowiednich proporcjach wysokości do szerokości (1:1.3 do 1:4). Ponieważ analizowany obraz pochodził z kamery zamontowanej na stałe na samochodzie, autorzy wykorzystali filtrację perspektywiczną, która uwzględnia możliwą wysokość ROI w różnych fragmentach obrazu. Filtracja jednorodnych regionów pozwoliła na odrzucenie kandydatów będących częścią szerszych obiektów niemających nic wspólnego z przechodnimi. Obraz człowieka cechuje się dużą rozpiętością wartości



Rys. 5.1. Obraz IR po binaryzacji i segmentacji [21]

temperatur. Obliczając odchylenie standardowe ROI w odcieniach szarości można wyeliminować obszary, które są poniżej progu określonego przez autorów.

Ostatnim krokiem algorytmu jest klasyfikacja wytypowanych kandydatów. Autorzy wykorzystują histogram zorientowanych gradientów jako deskryptor, tworząc wektor 3780 cech, które są następnie klasyfikowane przez SVM.

W celu zbadania dokładności algorytmu został przeprowadzony test na zbiorze CVC-14, który zawiera obrazy nagrane kamerą FIR podczas nocnego przejazdu samochodem. Testy wykazały, że metoda podwójnego progowania daje trzy razy lepsze rezultaty, niż przy wykorzystaniu pojedynczego progu. Wraz z zaproponowanymi technikami filtracji zaowocowało to bardzo efektywnym mechanizmem segmentacji. Ponad 95% występujących przechodniów zostało poprawnie wytypowanych jako kandydaci do klasyfikacji.

Cała procedura detekcji przechodniów osiągnęła wysoki poziom wydajności na poziomie 33 klatek (o wymiarach 640x471 px) na sekundę przy wykorzystaniu pojedynczego rdzenia CPU. Dokładność detekcji wyniosła 37,3% AMR (ang. *Average Miss Rate*), która jest porównywalny do innych metod opartych o HOG+SVM.

## 5.2. Stereotermowizja

W pracy [14] autorzy zaproponowali wykorzystanie dwóch kamer termowizyjnych tworząc system stereowizyjny. W obrazie termowizyjnym człowieka najcieplejszym obszarem jest zazwyczaj głowa. By wyodrębnić obszary zainteresowania, w których potencjalnie znajdują się przechodnie, zgrupowano piksele o wartościach powyżej kilku różnych progów. Każdy z tych obszarów zostaje następnie uznany za głowę i stanowi górną część okna detekcji. Wielkość

okna jest estymowana na podstawie odległości źródła ciepła od kamer. Odległość jest ustalana na podstawie mapy dysparcji (ang. *Disparity Map*). Do obliczania dysparcji, wykorzystywana jest różnica między oryginalnym oknem a oknem przesuwnym w drugim obrazie – algorytm SAD (ang. Sum of Absolute Difference). Na koniec każde okno zostaje przeskalowane do wielkości 64x128, wyznaczony deskryptor HOG i poddane klasyfikacji za pomocą liniowego SVM.

W pracy autorzy skupili się na optymalnym doborze parametrów deskryptora HOG. Wykorzystany zestaw danych zawierał 4400 obrazów: 2200 próbek z pieszymi oraz 2200 bez pieszych. Autorzy przeprowadzili po 10 procedur uczenia klasyfikatora dla każdej kombinacji wykorzystując różne zestawy danych do nauki i testów. W tabeli 5.1 zaprezentowane są wyniki tych testów. Badanie parametrów dla klasyfikacji SVM wykazało, że im większy zestaw uczący, tym lepszą można uzyskać skuteczność detekcji.

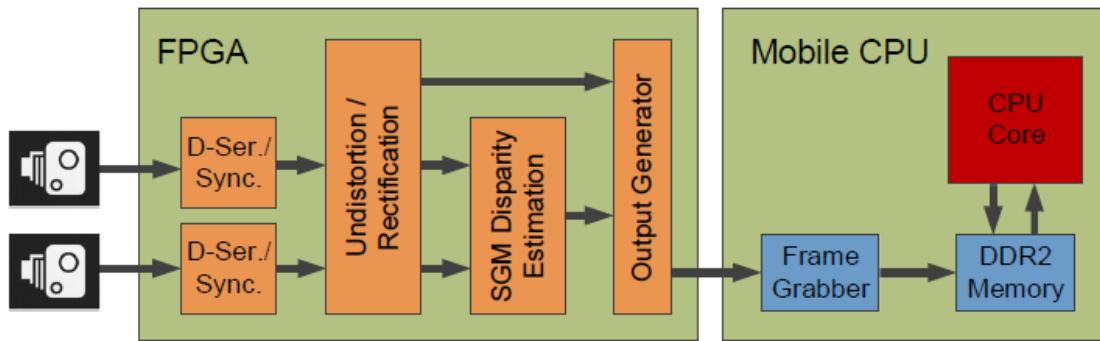
**Tabela 5.1.** Parametry HoG.

Parametr	Zestaw do testów	Najlepszy wynik
Rozmiar komórki	4x4, 8x8, 16x16	8x8
Rozmiar bloku	1x1, 2x2, 4x4	2x2
Nakładanie się komórek między blokami	1,2	1
Schemat normalizacji bloków	brak, L1, L2	L2
Liczba przedziałów histogramu	4, 8, 16	8
Rodzaj histogramu	ważony, nie ważony	ważony

### 5.3. Podejście sprzętowo-programowe. Stereowizja dla robotów.

W pracy [22] autorzy wykorzystali układ FPGA oraz CPU małej mocy do skonstruowania systemu wizyjnego dla robotów. System analizował obraz stereoskopowy z dwóch kamer wizyjnych tworząc mapę głębi. Obie kamery zostały bezpośrednio podpięte do układu FPGA za pomocą interfejsu LVDS (ang. *Low-Voltage Differential Signal* – niskonapięciowy sygnał różnicowy), w którym obrazy były następnie przetwarzane.

Schemat systemu jest przedstawiony na rysunku 5.2. W pierwszej kolejności obrazy z kamer są synchronizowane z wykorzystaniem bufora opóźniającego jedną linię danych. Przychodzące, zsynchronizowane piksele są zapisywane do bufora. Korekcja zniekształceń soczewkowych oraz rektyfikacja zostały połączone w jedną operację. Po obliczaniu współrzędnych



Rys. 5.2. Schemat systemu wizyjnego zaproponowanego w pracy[22].

piksela uwzględniającym korektę, jego wartość jest pobrana z właściwej lokalizacji w buforze wejściowym. Kolejnym krokiem było obliczanie dysparycji. W tym celu kontekst piksela z lewego obrazu jest porównywany z okolicznymi kontekstami pikseli w prawym obrazie. Kandydat jest wyłaniany na podstawie lokalnej funkcji kosztu i stałych globalnych bazujących na algorytmie SGM (ang. *Semi-Global Matching*).

Następnie dwa oryginalne obrazy oraz obliczona mapa głębi są przesyłane do CPU za pomocą dedykowanej magistrali. Moduł *frame grabbera* przechwytywał ten obraz i wykorzystując DMA (ang. *Direct Memeory Acces*) zapisywał do pamięci systemu. System pracował w rozdzielcości 752x480 pikseli i 60 klatkach na sekundę. Całość, włącznie z kamerami, układem FPGA, CPU oraz konwerterami napięcia pobierała mniej niż 5W mocy. Całkowita latencja podana przez autorów rozwiązania wynosi około 2ms. W opisywanej wersji systemu CPU nie realizowało żadnego algorytmu detekcji.

## 5.4. Podejście sprzętowo-programowe. System wspomagania kierowcy.

W pracy [23] autorzy wykorzystali układ SoC (ang. System on Chip) do detekcji pieszych dla zaawansowanego systemu wspomagania kierowcy (ADAS ang. *Advanced Driver Assistance System*). Głównym wyzwaniem było opracowanie systemu, która działa w czasie rzeczywistym, ma mały pobór mocy oraz niski koszt wykonania. Zwykle najbardziej skuteczne algorytmy wymagają znacznych zasobów obliczeniowych. Autorzy dokonali zatem relaksacji problemu poprzez zastosowanie prostszego deskryptora, jakim jest LBP oraz klasyfikatora SVM. Po każdej stronie pojazdu została zamontowana inteligentną kamerę o szerokim, 180° horyzontalnym kącie widzenia, by jak najlepiej monitorować przestrzeń wokół niego. W kamerach została prze-

prowadzona wstępna obróbka obrazu (rektyfikacja i skalowanie). Przetworzony obraz z kamer był transmitowany do "Fusion-Box", gdzie odbywała się generacja kandydatów, klasyfikacja, weryfikacja oraz śledzenie. Wyniki były przesyłane do wbudowanego komputera PC. Rozwiążanie nie zostało jeszcze w pełni zaimplementowane, ale pierwsze testy dawały obiecujące rezultaty.

## 5.5. Wzorzec probabilistyczny

W pracy [24] autorzy wykorzystali układ FPGA i naiwny klasyfikator Bayesa do detekcji przechodniów na obrazie termowizyjnym. W tym klasyfikatorze zakłada się, że wszystkie predyktatory są niezależne od siebie, co znacznie upraszcza obliczenia. Z jego wykorzystaniem można określić przynależność obrazu w badanym oknie do jednej z dwóch klas: zawierającego przeodnia albo który nie zawiera (czyli tła). W tym przypadku predyktorami są poszczególne piksele. Dla każdego piksela w oknie określa się prawdopodobieństwo jego przynależności do danej klasy. Klasyfikacja sprowadza się do zależności (5.1).

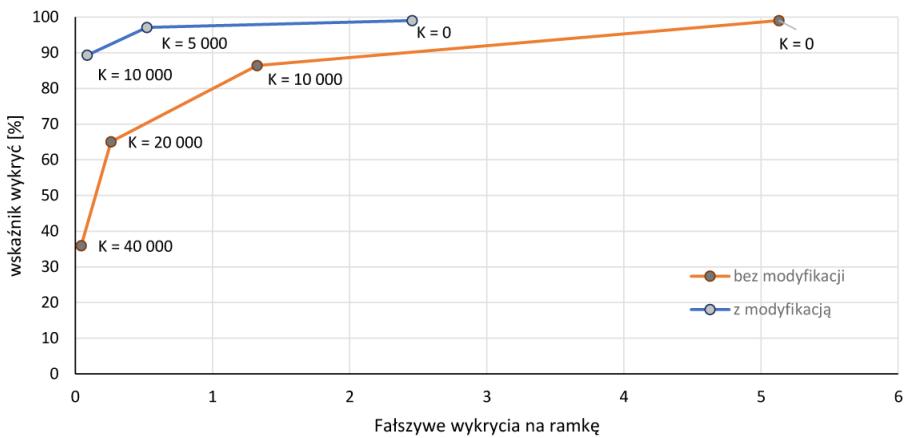
$$\sum_{x,y} \ln p(w_{x,y}|P) > \sum_{x,y} \ln p(w_{x,y}|\bar{P}) \quad (5.1)$$

gdzie  $p(w_{x,y}|P)$  to prawdopodobieństwo, że piksel  $w_{x,y}$  przynależy do obrazu człowieka  $p(w_i|\bar{P})$  przynależy do tła.

Jeżeli nierówność jest prawdziwa wtedy okno zostaje sklasyfikowane jako zawierające przeodnia. Obrazy wykorzystane w systemie są binarne, co oznacza, że  $p(w_{x,y}|P)$  przyjmuje dwie wartości, w zależności czy jest to piksel czarny czy biały. Można w ten sposób stworzyć macierz rozkładu prawdopodobieństwa (PDM ang. *probability distribution matrix*), która określa prawdopodobieństwo wystąpienia białego piksela w oknie (prawdopodobieństwo wystąpienia czarnego jest równe  $1 - p(w_{x,y}|P)$ ). Inną nazwą tej macierzy jest wzorzec probabilistyczny. Autorzy utworzyli PDM na podstawie 60 pozytywnych próbek.

W celu usprawnienia obliczeń w układzie FPGA macierz została przeskalowana na wartości całkowitoliczbowe w zakresie od 1 do 127. Następnie obliczono logarytm o podstawie dwa z uzyskanego wzorca i jego negacji (poprzez odjęcie od 128 wartości macierzy pozytywnej) i pomnożono przez 32. Utworzono tak dwie macierze: LPDW i LPDB dla białych i czarnych pikseli. (ang. *Logarithmic Probability Matrix* – logarytmiczna macierz prawdopodobieństwa). Przyjęto, że prawdopodobieństwo przynależności piksela do tła jest stałe i wynosi 50%, co daje wartość 192 po uprzednich przekształceniach. Ostatecznie klasyfikacji dokonuje się wg. wzoru:

$$L_p = \sum_{x=1}^j \sum_{y=1}^k (th(x, y) * LPMW(x, y) + (1 - th(x, y)) * LPMB(x, y)) \quad (5.2)$$



Rys. 5.3. Wyniki symulacji przy różnych wartościach parametru K.[20]

$$L_b = j * k * 192 \quad (5.3)$$

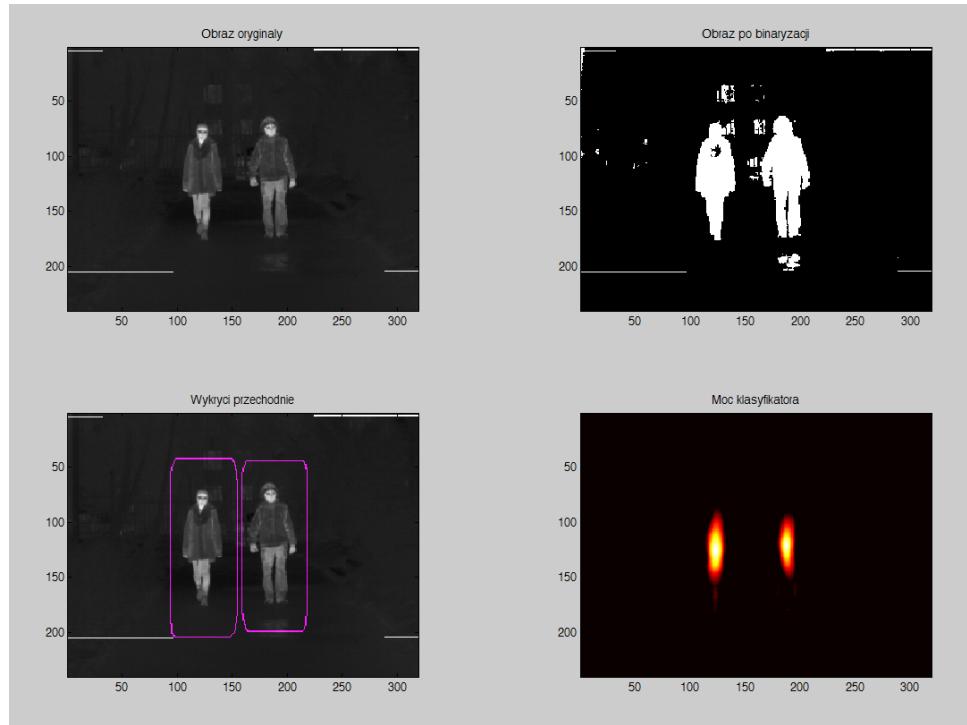
$$IsPedestrian = \begin{cases} 1 & \text{gdy } L_p \geq L_b + K \\ 0 & \text{gdy } L_p < L_b + K \end{cases} \quad (5.4)$$

gdzie  $j$  i  $k$  stanowią wysokość i szerokość okna przesuwnego. Wartości  $L_p$  i  $L_b$  odnoszą się do sum prawdopodobieństw przynależności białego/czarnego piksela do obrazu człowieka.  $K$  jest parametrem, którym określono minimalną wartość dla której klasyfikator daje poprawne pozytywne wyniki.

W omawianej pracy autorzy opracowali wzorce dla 3 różnych wielkości okna: 10x15, 8x12 i 6x9 pikseli. Do ich stworzenia wykorzystali jedynie górne połówki obrazu termicznego ludzkiej sylwetki, co miało na celu uzyskanie bardziej niezawodnego wzorca dla różnych wysokości zarejestrowanych przechodniów.

## 5.6. Praca Inżynierska

Niniejsza praca magisterska jest kontynuacją pracy inżynierskiej [20] oraz artykułu opracowany na jej podstawie[25]. W pracy inżynierskiej została rozwinięta koncepcja systemu zaproponowana w rozdziale 5.5. Zastosowano 4 wzorce o wymiarach 48x120, 32x80, 24x60 oraz 19x48. Wzorzec został podzielony na 4 części: górną, dolną, prawą i lewą. Następnie, podobnie jak w DPM, układ tych 4 części decydował o wyniku klasyfikacji. Każda część tworzyła maskę wielkości całego okna. Jeżeli co najmniej 3 takie maski pokrywały się, obszar ten był klasyfikowany pozytywnie. Pozwoliło to na znaczną poprawę działania algorytmu, co pokazuje wykres ROC (ang. *Receiver Operating Characteristic*) przedstawiony na rysunku 5.3.



**Rys. 5.4.** Proces detekcji wzorcem probabilistycznym. [20]

Poprawiło to również detekcję w różnych skalach. Pojedynczy wzorzec zapewniał detekcję w pewnym zakresie wysokości przechodniów. Im dalej odbiegał od swojej nominalnej wysokości, tym bardziej należało zmniejszyć próg detekcji  $K$ , by w pojedynczym oknie było możliwe wykrycie mniejszych i większych sylwetek. Powodowało to również zwiększenie liczby fałszywych detekcji. Po wprowadzeniu modyfikacja każda część ma możliwość deformacji, co w dynamiczny sposób pozwala na zmianę wysokości wzorca.

Dodatkową ciekawą właściwość można zauważyć na rysunku 5.4, gdzie zaprezentowano implementację systemu w programie Matlab. W oknie przedstawiającym „moc klasyfikacji” ( $L_p - (L_b + K)$ ) wyraźnie widać dwa maksima lokalne wskazujące dokładną pozycję przechodniów. Ta właściwość została wykorzystana w niniejszej pracy w celu ustalenia ROI.



## **6. Zrealizowany system wizyjny do detekcji ludzi**

### **6.1. Koncepcja systemu**

Zadaniem systemu jest detekcja osób na obrazie multispektralnym. Do uzyskania obrazu multispektralnego została wykorzystana kamera wizyjna, dająca obraz o rozdzielcości 640x480 pikseli i 50 klatek na sekundę oraz termowizyjna: Lepton – 80x60 pikseli i 9 klatek na sekundę. Obraz z kamery termowizyjnej (IR) jest dopasowywany do wizyjnego (RGB) za pomocą projekcji perspektywicznej. Wynikiem ich połączenia jest obraz multispektralny (RGBIR). Wybór ROI odbywa się tylko z wykorzystaniem obrazu termowizyjnego. W tym celu został wykorzystany moduł PDM (ang. *Probability Density Matrix*) zaczerpnięty z pracy inżynierskiej autora [20]. Tworzy on listę kandydatów, z której jest wybierany najbardziej prawdopodobny wynik. Następnie z ROI o wymiarach 80x192 piksele wyodrębniane są deskryptory HOG oraz przeprowadzana jest klasyfikacja z wykorzystaniem SVM. Wynik detekcji jest prezentowany na ekranie, poprzez obramowanie sylwetki przechodnia. Do realizacji tego systemu została wykorzystana płytka deweloperska ZYBO firmy Digilent. Bazuje ona na omówionym wcześniej w rozdziale 4 układzie Zynq-7000. Jest to układ heterogeniczny, co daje możliwość realizacji poszczególnych elementów systemu wizyjnego w logice programowalnej (PL) lub systemie procesorowym (PS). Zaproponowano następujący podział zadań:

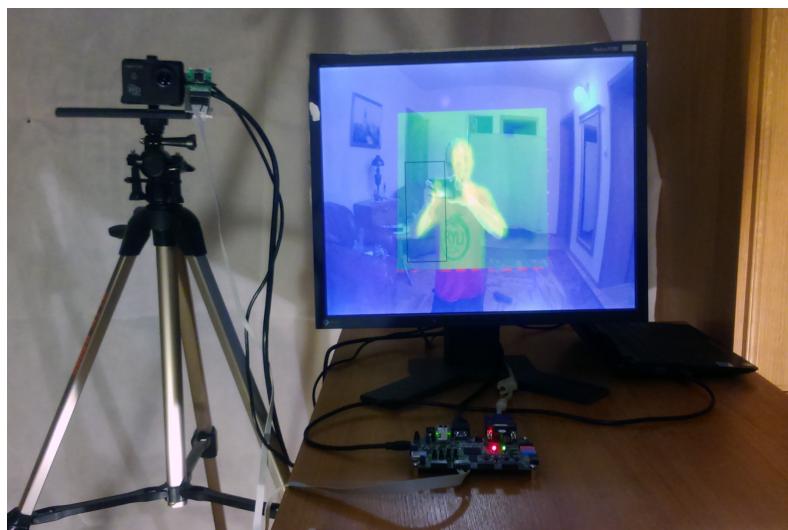
Logika programowalna:

- Akwizycja obrazu poprzez HDMI (RGB) i VoSPI (IR),
- Transformata projekcyjna i interpolacja obrazu IR,
- Nałożenie i synchronizacja obrazu IR do obrazu RGB,
- Prezentacja wyników,
- Detekcja kandydatów za pomocą wzorca probabilistycznego.

System procesorowy:

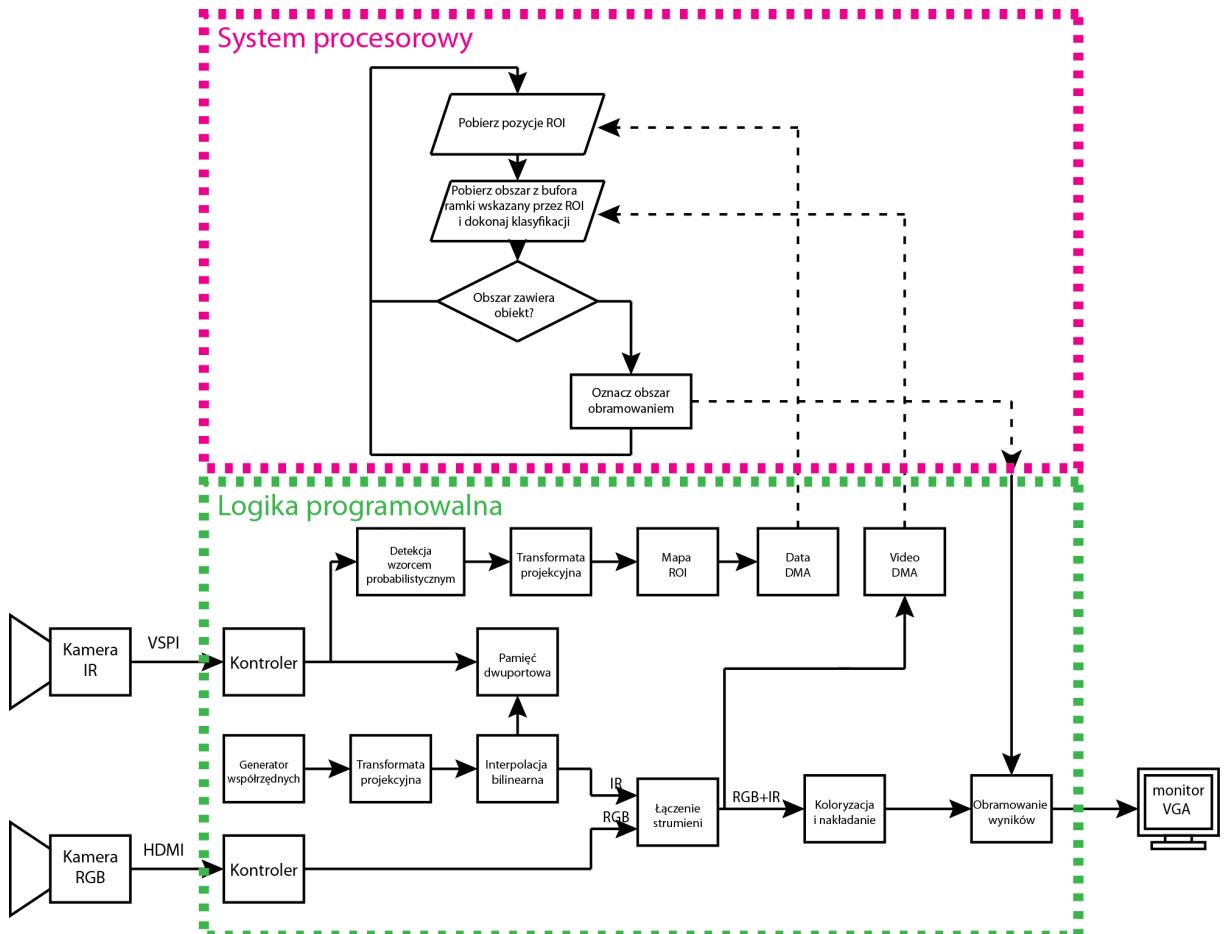


**Rys. 6.1.** Wykorzystany system kamer. Po lewej stronie znajduje się kamera wizyjna, po prawej termowizyjna Lepton.



**Rys. 6.2.** Widok na kompletny system do detekcji obiektów w przestrzeni multispektralnej. Obraz jest rejestrowany przez zespół kamer znajdujący się na statywie. Kamery są połączone do płytki deweloperskiej Zybo. Wynik jest prezentowany na monitorze.

- konfiguracja parametrów systemu wizyjnego w logice programowej poprzez interfejs AXI4-Lite,
- Klasyfikacja obszarów wytypowanych przez wzorzec probabilistyczny (HOG+SVM),
- Generowanie wyników.



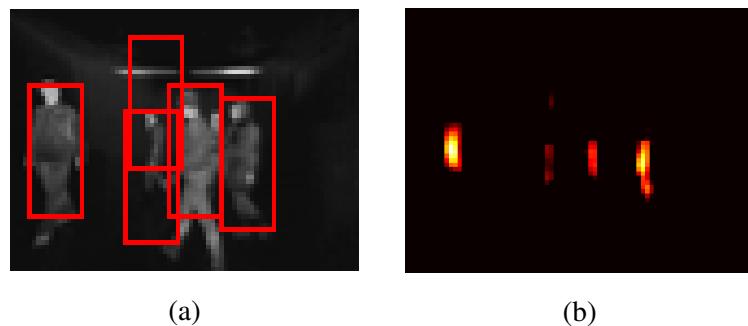
Rys. 6.3. Schemat blokowy systemu detekcji.

Na rysunku 6.3 został przestawiony ogólny schemat rozwiązania.

## 6.2. Model programowy

W celu sprawdzenia koncepcji systemu została wykonana jego implementacja w pakiecie Matlab. Do testów został wykorzystany zestaw filmów pochodzący z pracy [26]. Przedstawia on pomieszczenie, po którym porusza się od 1 do 5 aktorów. Obraz został nagrany dwoma kamerami: termowizyjną FLIR Thermovision A40M oraz wizyjną Sony XCD-710CR. Rozdzielcość obu klipów wynosi 480x360 piksele. W pierwszym etapie obraz z kamery termowizyjnej został

pomniejszony do rozmiarów 80x60 piksele. Następnie, za pomocą wzorca probabilistycznego o wymiarze 30x12 piksele została obliczona mapa rozkładu LP (ang. *Logarithmic Probability*) na obrazie jak na rysunku 6.4b. Maksima lokalne występujące w tym rozkładzie stanowią punkty środkowe ROI do dalszej analizy (rysunek 6.4a).

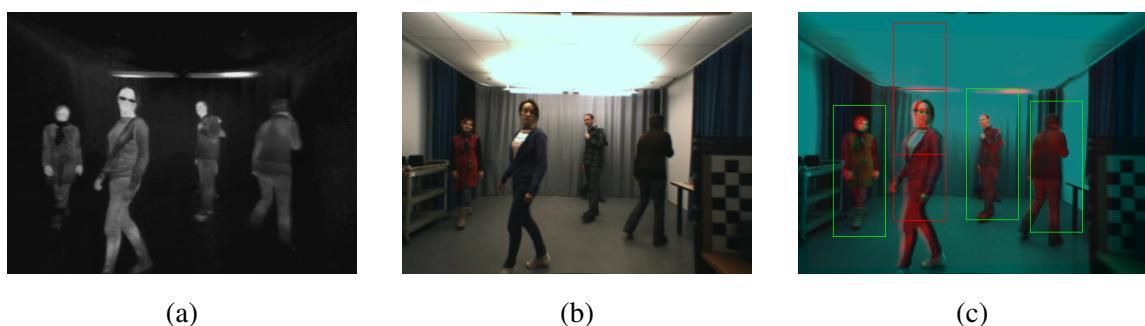


**Rys. 6.4.** Poszukiwanie kandydatów za pomocą wzorca probabilistycznego.

(a) wytypowani kandydaci (b) mapa LP.

Obraz termowizyjny został dopasowany do obrazu wizyjnego, tworząc obraz multispektralny. Punkty kalibracyjne zostały wyznaczone na podstawie elementów sylwetek (stopy, dłoń i głowa), od których aktorów wysokość na obrazie wizyjnym wynosiła około 180 pikseli, uzyskując minimum dysparcji między dwoma obrazami dla obiektów poszukiwanych wzorcem o wysokości 30 pikseli.

Następnie współrzędne znalezionych kandydatów podczas badania wzorcem probabilistycznym zostały przeniesione na układ współrzędnych obrazu multispektralnego z wykorzystaniem parametrów transformacji uzyskanej podczas kalibracji. Wokół tych punktów zostały wyznaczone ROI o wymiarach 80x192 pikseli. Dalej został obliczony wektor cech HOG w każdym z okien, które następnie zostały poddane klasyfikacji za pomocą liniowego SVM.



**Rys. 6.5.** (a) Obraz z kamery termowizyjnej, (b) obraz z kamery wizyjnej, (c) połączone obrazy z zaznaczonymi wynikami detekcji. Zielona ramka oznacza pozytywny wynik klasyfikacji HOG SVM, czerwona – negatywnie sklasyfikowani kandydaci.

Do nauczenia SVM zostały wykorzystane 141 ROI wytypowanych przez wzorzec probabilistyczny. Każdemu ROI została przypisana klasa: 1 – jeżeli zawiera obraz człowieka o wysokości 160+/-20 px , 0 – jeżeli jest to element tła bądź część człowieka (np. sama głowa, nogi).

Na 141 próbek 78 stanowiły próbki pozytywne a 62 negatywne. SVM sklasyfikował poprawnie pozytywnie 88,9% próbek zaś poprawnie negatywnie 88,5%.

### 6.3. Wykorzystanie AXI-Stream do transmisji sygnału wideo.

W odróżnieniu od standardowego sposobu przetwarzania strumieniowego wideo, w AXI4-Stream przesyłane są jedynie aktywne piksele. Linie synchronizacji poziomej i pionowej są odrzucane albo przekierowywane do specjalnego bloku, w którym są mierzone parametry wchodzącego strumienia wizyjnego (liczba pikseli w linii, liczba aktywnych linii, czas wygaszania itd.). W celu wyświetlenia obrazu wykorzystuje się ten sam blok, który ma możliwość generacji nowych sygnałów synchronizacji (ich odtworzenia).

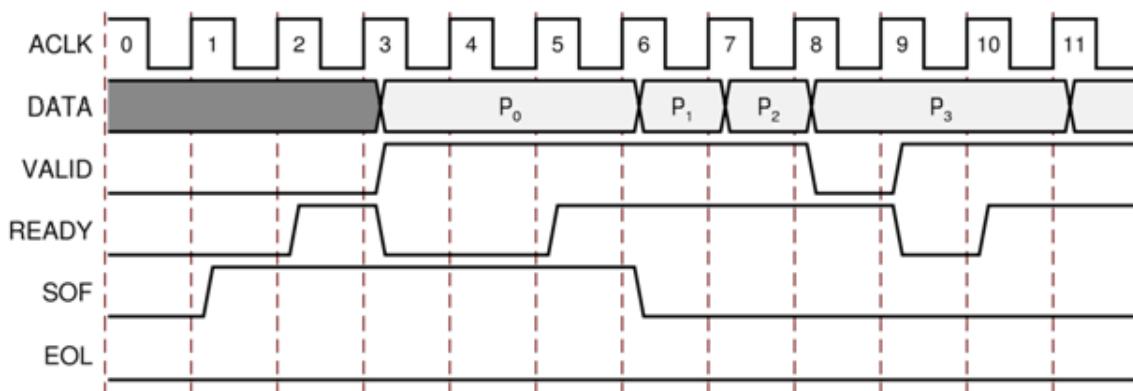
Do transmisji wykorzystane jest 6 sygnałów: dane i pięć kontrolno-sterujących. W nawiasach są podane linie, które wykorzystuje dany sygnał w interfejsie AXI4-Stream.

- *Video Data (tdata)* – linia danych o szerokości jednego (albo dwóch) pikseli. Szerokość tej linii powinna być wielokrotnością liczby osiem (16, 24, 48 itd.)
- *Valid(tvalid)* – linia określająca czy dane piksela są poprawne,
- *Ready (tready)* – linia kontrolna informująca urządzenie master, że *slave* jest gotowy do transmisji danych,
- *Start Of Frame (tuser)* – linia sygnalizacyjna pierwszego piksela nowej ramki,
- *End Of Line (tlast)* – linia sygnalizacyjna ostatniego piksela w linii.

Aby mógł wystąpić poprawny transfer danych linie *Valid* i *Ready* muszą być w stanie wysokim podczas rosnącego zbocza zegara. Przykładowe nawiązanie transmisji przedstawia rysunek 6.6.

### 6.4. AXI VDMA

Wiele aplikacji wizyjnych wymaga przechowania całej ramki obrazu w celu jej dalszej obróbki np. podczas skalowania, przycinania bądź dopasowania liczby klatek na sekundę. Część



Rys. 6.6. Przykład rozpoczęcia transmisji Ready/Valid.

programowalna układu Zynq zazwyczaj nie posiada wystarczającej liczby wewnętrznych zasobów pamięciowych do przechowywania pełnej klatki obrazu. Aby stworzyć taki bufor jedną z możliwości jest wykorzystanie mechanizmu bezpośredniego dostępu do pamięci, który pozwala na przesyłanie i wczytywanie danych z logiki programowej do pamięci RAM bez konieczności angażowania procesora. Należy w tym miejscu zaznaczyć, że dla rozważanej platformy Zybo zewnętrzna pamięć RAM DDR podłączona jest do kontrolera w systemie procesorowym.

Dostęp ten realizuje się poprzez IP-Core AXI VDMA. Zapewnia on przejście między interfejsem AXI4-Stream, a AXI4 Memory Map w obu kierunkach. Przed rozpoczęciem przesyłania IP-Core jest konfigurowany poprzez interfejs AXI4-Lite. Konfiguracja zawiera adres w pamięci RAM do którego ma być zapisana bądź wczytana ramka obrazu. Po wgraniu do pamięci ramki kontroler może wywołać przerwanie dla systemu procesorowego.

## 6.5. Opis modułów zaimplementowanych w logice programowalnej

### 6.5.1. Kontroler kamery IR

Kamera Lepton przesyła obraz za pomocą interfejsu VoSPI (ang. *Video over Serial Peripheral Interface*). W zastosowanym rozwiążaniu jest on urządzeniem typu *slave*, zaś układ Zynq jest *masterem*. Są wykorzystywane 3 z 4 linii typowego kanału SPI: SCK (ang. *Serial Clock* – zegar), /CS (ang. *Chip Select* – wybór układu)(aktywny stanem niskim) oraz MISO (ang. *Master In/Slave Out* – wejście master/wyjście slave). Transmisja rozpoczyna się od podania stanu niskiego przez kontroler na linii /CS. Powoduje to aktywację linii. Następnie, *master* rozpoczyna taktowanie zegarem na linii SCK. *Slave* wystawia kolejne bity danych począwszy od MSB (ang.

*Most Significant Bit) na linię MISO.* Kontroler szczytuje te bity przy każdym rosnącym zboczu zegara do 16 bitowego rejestru przesuwnego. Dane przesyłane z kamery są zorganizowane w pakiety po 164 bajty. Pakiet rozpoczyna się nagłówkiem składającym się z 2 bajtów pola identyfikacyjnego oraz 2 bajtami sumy CRC. Pole identyfikacyjne spełnia dwa zadania. Po pierwsze, stanowi 12-bitowy numer pakietu (a zarazem numer linii obrazu). Po drugie, w przypadku błędnego pakietu zawiera wartość 0xFFXX (X to obojętna wartość) co wskazuje, że nadchodzący pakiet powinien zostać zignorowany przez kontroler. Zawartość pakietu stanowi 160 bajtów zawierających wartości 80 pikseli linii. W systemie wykorzystywany jest format RAW14, zaś temu każdy piksel jest przesyłany w postaci 2 bajtów zawierających 14 bitową wartość piksela. Cała ramka obrazu składa się z 63 pakietów. 60 pierwszych pakietów stanowią linie obrazu zaś ostatnie 3 są przeznaczone na telemetrię, która zawiera m.in. temperaturę matrycy i obudowy, 32-bitowy licznik ramek obrazu i bity stanu. Pierwsze 3 przesłane pakiety są błędne i służą do synchronizacji transmisji. W przypadku prawidłowego pakietu kontroler szczytuje numer linii obrazu i wystawia go na wyjściu *row*. Następnie na wyjściu *data* wystawiane są wartości kolejnych 80 bitów wraz z ich pozycją na wyjściu *column*. Wysoki stan *we* informuje, że dane są poprawne i powinny być zapisane. Wartości *row* i *column* są zamieniane na adres w pamięci dwuportowej BRAM, będącą bufor ramki IR. Do zaadresowanej komurki pamięci zapisywana jest wartość piksela z wyjścia *data*.

### 6.5.2. Transformata projekcyjna

Moduł służy do dopasowania obrazu IR do RGB. W tym celu zamienia współrzędne w układzie odniesienia obrazu RGB na odpowiadające im w układzie IR. Na wejściu podawany jest strumień AXI4-Stream służący do synchronizacji ramek oraz 12-bitowe współrzędne X i Y. Moduł realizuje operację:

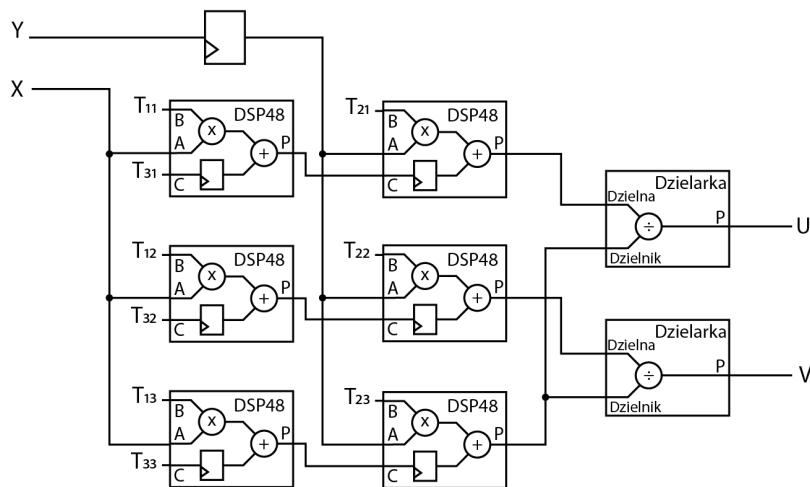
$$\begin{bmatrix} u_n & v_n & n \end{bmatrix} = \begin{bmatrix} x & y & 1 \end{bmatrix} T \quad (6.1)$$

$$u = \frac{u_n}{n} \quad (6.2)$$

$$v = \frac{v_n}{n} \quad (6.3)$$

gdzie *x*, *y* to współrzędne obrazu w układzie odniesienia kamery wizyjnej, *u*, *v* to odpowiadające im współrzędne w układzie odniesienia kamery termowizyjnej. *T* to macierz transformacji.

Moduł wystawia na wyjściu strumień wizyjny AXI4-Stream, 12-bitowe wartości U i V oraz ich części ułamkowe w U\_fraction i V\_fraction (14 bitów). W module zostały wykorzystane 34 z 80 dostępnych w układzie Zynq (wersji na karcie Zybo) modułów DSP48 do wykonania



Rys. 6.7. Schemat modułu transformacji projekcyjnej.

operacji arytmetycznych, z czego większość wychodzi w skład IP-Core dzielarki dostarczonej przez producenta układu (14 na pojedyńczą dzielarkę). Dzielenie nie odbywa się w pełni pote- kowo. Użyty w dzielarce algorytm *High\_Radix* wymaga zatrzymania strumienia na czas obliczeń. Zmniejszenie liczby instancji dzielarek pozwoliło na zaoszczędzenie pewnej liczby (jak się później okazało istotnej) zasobów logicznych. Jednak dzięki zastosowaniu wyższej częstotliwości niż zegar pikseli obrazu RGB oraz bufora (250 MHz) nie stanowi to wąskiego gardła systemu. Macierz transformacji T jest zapisana w dziewięciu 32-bitowych rejestrach i konfigurowalna poprzez interfejs AXI4-Lite. Elementy macierzy są 25 liczbami w notacji stałoprzecinowej: 1 bit znaku 10 – część całkowita, 14 – część ułamkowa. Taka dokładność pozwala na maksymalne wykorzystanie pojedynczych modułów DSP48 (gdyż wejście B jest 25-bitowe).

Rysunek 6.7 przedstawia schemat modułu.

### 6.5.3. Interpolacja dwuliniowa

Moduł pozwala na interpolację wartości piksela wskazanego przez koordynaty na wejściu. Podobnie jak reszta systemu używa AXI4-Stream do przekazywania danych między poszczególnymi modułami. Dane na wejściu to współrzędne U i V oraz ich części ułamkowe  $U_{fraction}$  ( $U_f$ ) i  $V_{fraction}$  ( $V_f$ ). Moduł został wyposażony w 4 rejesty, w których przechowywane są współrzędne oraz wartości 4 ostatnio użytych pikseli. Zabieg ten znacznie redukuje liczbę potrzebnych zapytań do pamięci. Podczas powiększania obrazów istnieje duża szansa, że kolejne koordynaty na wejściu U, V odwołają się do tych samych czterech otaczających ich pikseli (np. [0,0], [1,0], [2,0], podczas zwiększenia 10-krotnego, wynikiem transformacji byłyby punkty: [0,0], [0.1,0], [0.2,0], ... więc w celu interpolacji odwoływałyby się do wartości otaczających ich pikseli: [0,0] , [1,0], [0,1], [1,1]). W module następuje sprawdzenie, czy w pamięci są już war-

tości z koordynatów  $[U, V]$ ,  $[U+1, V]$   $[U, V+1]$ ,  $[U+1, V+1]$ . Jeżeli któregoś piksela brakuje, jest on pobierany z pamięci i zapisywany w rejestrze przechowującym niepotrzebny piksel. Jeżeli wszystkie koordynaty się zgadzają, obliczana jest wartość piksela wyjściowego zgodnie ze wzorem (6.4).

$$Ir = A(1 - U_f)(1 - V_f) + B(1 - V_f) + C(1 - U_f)V_f + DU_fV_f \quad (6.4)$$

gdzie:  $A, B, C, D$  odpowiadają wartościom pikseli w  $[U, V]$ ,  $[U+1, V]$   $[U, V+1]$ ,  $[U+1, V+1]$ , a  $Ir$  to wartość wyjściowa piksela wyjściowego.

Moduł działa potokowo. W przypadku gdy wymagana jest aktualizacja rejestrów, strumień jest wstrzymywany na czas pobrania stosownych wartości z bufora ramki IR. Jeżeli koordynaty wejściowe wychodzą poza zakres obrazu termowizyjnego, to ich wartość wyjściowa odgórnie wynosi zero.

#### 6.5.4. Łączenie strumieni

Moduł posiada dwa wejścia AXI4-Stream. Strumień RGB jest nadrzedny i do niego jest dołączany strumień IR. Do synchronizacji została wykorzystana możliwość wstrzymania transmisji poprzez linię *tready* w interfejsie AXI4-Stream. Piksele z dołączanego obrazu są odrzucone do momentu pojawiienia się sygnału SOF, reprezentowanym przez wysoki stan linii *tuser*. Następnie, w momencie pojawiienia się sygnału SOF w strumieniu głównym transmisja zostaje ponownie wznowiona. Po przejściu całej ramki strumienie są ponownie synchronizowane.

#### 6.5.5. Koloryzacja i nakładanie

Strumień RGBIR zostaje połączony w jeden obraz. Obraz IR zostaje poddany koloryzacji na podstawie wartości zapisanych w 12-bitowym LUT (ang. *Lookup table*). Rysunek 6.9 przedstawia użytą paletę do wizualizacji temperatury. Obrazy nakładają się w proporcjach 50 na 50. Jeżeli wartość piksela IR jest równa zero to nie jest on wyświetlany. Na wyjściu jest podany 24-bitowy strumień RGB. Przykładowy wynik operacji przedstawiono na rysunku 6.8.

#### 6.5.6. Obramowanie wyników

Moduł ma na celu wskazanie na obrazie lokalizacji wykrytego przechodnia, poprzez obramowanie tego obszaru ramką określonego koloru. Kolor, rozmiar i lokalizacja ramki jest zapisana w dwóch 32-bitowych rejestrach, konfigurowalnych poprzez interfejs AXI4-Lite.



Rys. 6.8. Obraz IR po koloryzacji i nakładaniu.



Rys. 6.9. Paleta kolorów użyta do wizualizacji temperatury.

### 6.5.7. Moduł DPM

Moduł został zaczerpnięty z pracy inżynierskiej autora w celu selekcji kandydatów w obrazie multispektralnym i jest opisany w rozdziale 5.5. Do detekcji wykorzystuje bezpośredni strumień pikseli kamery termowizyjnej. Pomocniczy moduł *data grabber* znajdujący się tuż za kontrolerem kamery IR ma za zadanie rozdzielenie sygnału do dwóch komponentów: bufora ramki oraz, po zbinaryzowaniu, do modułu DPM wraz z jego koordynatami. Moduł składa się z okna kontekstowego o wymiarach 16x40 pikseli, gdzie odbywa się porównanie z macierzą wzorcową. Do każdego piksela w oknie przypisywana jest wartość ze wzorca LPBW – jeżeli piksel jest biały albo LPMB w przypadku czarnego. Następnie, wszystkie wartości są sumowane za pomocą drzewa sumacyjnego. Wynikiem jest wartość wyjściowa LP (ang. *Logarithmic Probability*). Jeżeli przekroczy ona wartość progową (ustaloną na podstawie sumy LP policzoną dla tła i parametru K) zostaje przesłana do listy kandydatów wraz ze współrzędnymi tego okna (w układzie odniesienia kamery IR). Lista kandydatów jest na bieżąco przesyłana za pomocą AXI4-Stream do pamięci systemu procesorowego poprzez AXI DMA. Po sprawdzeniu ostatniego okna zostaje przesłany sygnał *tlast* i moduł AXI DMA wygeneruje przerwanie w systemie procesorowym. Wartość progowa binaryzacji i LP jest konfigurowalna za pomocą AXI4-Lite.

## 6.6. System procesorowy

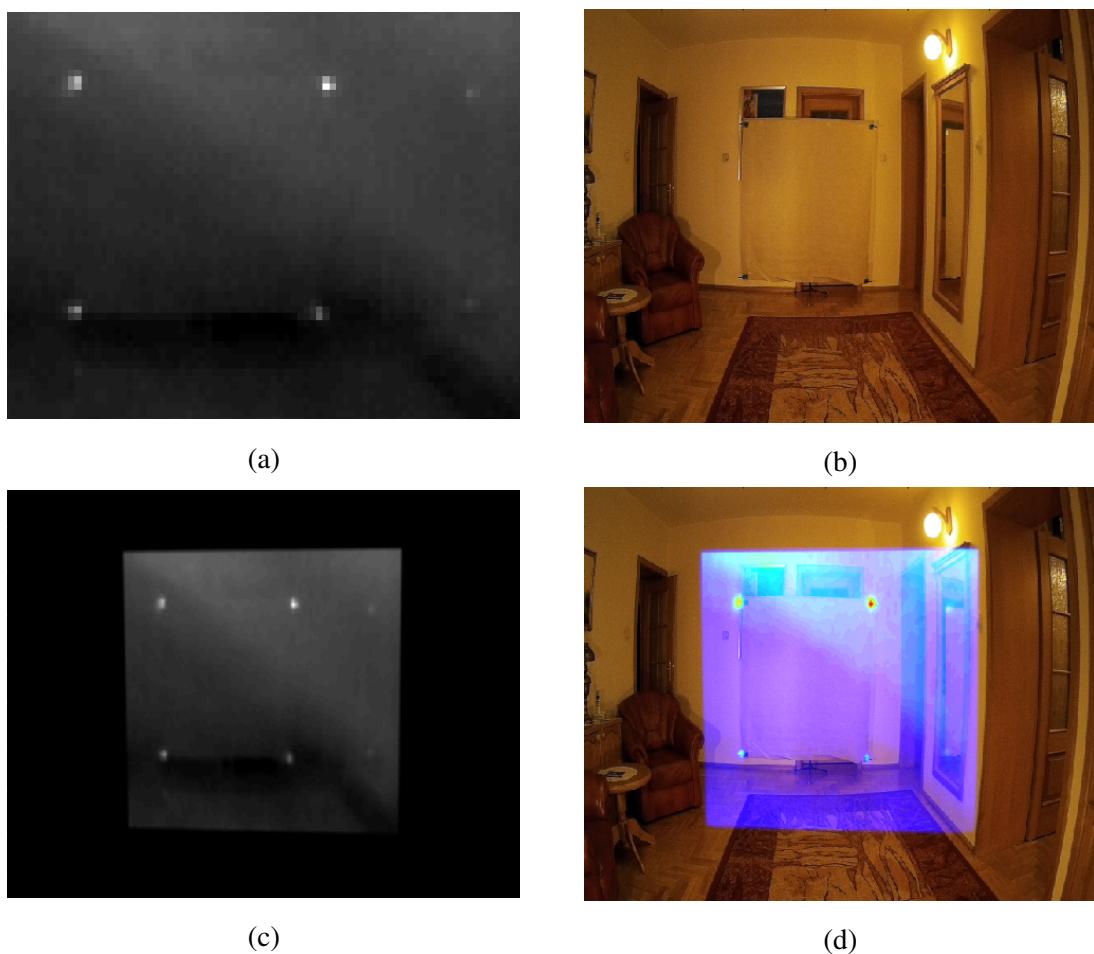
Program działający w części procesorowej układu Zynq spełnia dwa podstawowe zadania. Po pierwsze pozwala na konfigurację modułów znajdujących się w PL za pomocą AXI4-Lite. Użytkownik za pośrednictwem konsoli może wprowadzać własne parametry dla każdego z modułów. Program pozwala również na zapis na karcie SD pojedynczej klatki obrazu multispektralnego, następnego wykrytego ROI oraz ROI znajdującego się pośrodku sceny. Ta opcja ułatwia tworzenie bazy do nauczenia klasyfikatora SVM.

Drugim zadaniem jest klasyfikacja wytypowanego przez DPM kandydata. Po otrzymaniu przerwania przez moduł AXI DMA powiązany z DPM, sprawdzana jest lista kandydatów i wybierany jest ten o największej wartości LP. Współrzędne kandydata są podane w układzie odniesienia kamery IR. Po przeliczaniu ich za pomocą macierzy transformacji projekcyjnej i dodaniu pewnego *offsetu* w buforze ramki obrazu RGBIR zostaje wskazane ROI o wymiarach 80x192 pikseli. Następnie, obliczany jest wektor cech HOG z tego obszaru. Potem na jego podstawie odbywa się klasyfikacja przy użyciu wytrenowanego SVM. Wynik klasyfikacji jest wyświetlany w konsoli wraz z jego współrzędnymi na obrazie i wartością LP. Dodatkowo obszar ten zostanie zaznaczony zieloną ramką na wyjściowym strumieniu wizyjnym. Jeżeli kandydat

nie zostanie zakwalifikowany pozytywnie przez SVM ramka przybierze kolor czerwony. Brak kandydatów wskazuje czarna ramka w miejscu ostatniej detekcji.

## 6.7. Proces kalibracji

W celu kalibracji, zostaje wykonane zdjęcie specjalnej planszy kalibracyjnej za pomocą obu kamer (rysunki 6.10a, 6.10b). Pozwala to na identyfikację czterech punktów kalibracyjnych w obu przestrzeniach: RGB i IR. W użytej planszy zastosowano rozgrzane kawałki metalu jako punkty odniesienia. Następnie na ich podstawie zostaje obliczona macierz transformacji projekcyjnej. Na rysunku 6.10c przedstawiono obraz z kamery termowizyjnej po transformacji, który pokrywa się z obrazem wizyjnym (rysunek 6.10d).



**Rys. 6.10.** Proces kalibracji: (a) obraz z kamery termowizyjnej, (b) obraz z kamery wizyjnej, (c) obraz z kamery termowizyjnej po transformacji projekcyjnej, (d) obraz z kamery termowizyjnej po transformacji projekcyjnej nałożony na obraz z kamery wizyjnej.

## 6.8. HOG i SVM

W pierwszym etapie tworzony jest wektor cech HOG. Dla każdego piksela w ROI zostaje obliczony gradient składający się z kierunku oraz modułu. Następnie okno detekcji zostaje podzielone na 60 komórek o wielkości 16x16 pikseli.

Dla każdej z komórek zostaje obliczony histogram ważony o 9 przedziałach.

Oznacza to, że wartość moduł gradientu jest dzielona na dwa przedziały, pomiędzy którymi się znajduje w proporcjach określonych wzorem (6.6), (6.7).

$$Bin1_{dir} < G_{dir} < Bin2_{dir} \quad (6.5)$$

$$Bin1_{mag} = \frac{Bin2_{dir} - G_{dir}}{Bin2_{dir} - Bin1_{dir}} \quad (6.6)$$

$$Bin2_{mag} = \frac{G_{dir} - Bin1_{dir}}{Bin2_{dir} - Bin1_{dir}} \quad (6.7)$$

gdzie:  $G_{dir}$  to kierunek badanego gradientu,  $G_{mag}$  to wypadkowa badanego gradientu,  $Bin1_{dir}, Bin2_{dir}$  to kierunki gradientów powiązane z danym przedziałem histogramu,  $Bin1_{mag}, Bin2_{mag}$  to wypadkowe gradientu przypisana do poszczególnego z przedziału.

Dla każdej z komórek dodatkowo obliczana jest suma kwadratów wartości przedziałów histogramu. Następnie komórki są łączone w bloki 2 na 2, w których obrębie dokonywana jest normalizacja, wykorzystując wcześniej obliczone sumy kwadratów. Zastosowano normalizację L2 wyrażoną wzorem (6.8).

$$norm = \sqrt{\sum_i (Bin(i)_{mag})^2 + c} \quad (6.8)$$

gdzie:  $Bin(i)_{mag}$  to wartość przedziału histogramu, a  $c$  to mała wartość stała. Następnie wartości wszystkich 36 przedziałów w 4 histogramach są dzielone przez  $norm$ . Bloki nakładają się na siebie, zatem w pojedynczym oknie można wyodrębnić 44 bloki. Suma histogramów z wszystkich bloków tworzy 1584-elementowy wektor cech.

Maszyna wektorów nośnych jest modelem służącym do określania przynależności wektora cech do jednej z klas. W procesie nauczania tworzona jest superpłaszczyzna oddzielająca przykładowe próbki tych dwóch klasy z największym możliwym marginesem między nimi. Próbki znajdujące się na marginesie nazywane są wektorami nośnymi. W celu wytrenowania klasyfikatora zostało wykonane 60 obrazów – 30 pozytywnych zawierających przechodnia i 30 przedstawiają elementy tła lub niekompletnej sylwetki człowieka (np. sama ręka). Po nauczeniu do każdego elementu wektora cech przypisana jest jego waga oraz wartość progowa –  $bias$ . Wynik sumy iloczynów wag i cech wektora oraz  $biasu$  stanowi podstawę do klasyfikacji. Jeżeli jest większy od zera świadczy to o pozytywnym wyniku klasyfikacji.



Rys. 6.11. Działający system.

## 6.9. Wyniki

Aby sprawdzić działanie i dokładność systemu została zaimplementowana możliwość zapisu obliczonego wektora cech na karcie SD. Następnie został obliczony przykładowy błąd względny między wektorem cech wyliczonym w pakiecie Matlab, a uzyskanym z systemu wizyjnego. Błąd oscyluje w granicy  $10^{-6}$  co czyni go marginalnym i najprawdopodobniej wynika z różnic w użytych bibliotekach numerycznych. Świadczy to o prawidłowym działaniu systemu.

Na przebadanie jednego okna zaproponowany system potrzebuje 75 ms (dla porównania te same obliczenia w pakiecie Matlab zajmują około 23 ms przy użyciu komputera z procesorem Pentium Core i7 i 8GB pamięci RAM). Dzięki zastosowaniu sprzętowego wyszukiwania ROI zadanie systemu procesorowego zostało ograniczone do analizy pojedynczego ROI. Kamera termowizyjna, będąca źródłem sygnału dla wzorca probabilistycznego, pracuje z prędkością 9 klatek na sekundę co zapewnia 111 ms na analizę jednej ramki obrazu. Ponieważ analiza jednego okna zajmuje 75 ms możliwe sprawdzenie tylko jednego ROI na ramkę. Na rysunku 6.11 przedstawiono zdjęcie działającego systemu.

W tabeli 6.1 zostało przedstawione wykorzystanie zasobów logiki programowej. Najwięcej zasobów LUT zostało wykorzystane w module DPM. Spowodowane jest to implementacją dużego okna kontekstowego wraz z drzewem sumacyjnym. Z kolei większość modułów DSP48 została wykorzystana do dzielarek w module transformaty projekcyjnej, gdzie odbywa się najczęściej obliczeń arytmetycznych.

**Tabela 6.1.** Wykorzystane zasoby logiki programowalnej.

Element	Wykorzystane	Dostępne	%
LUT	12583	17600	71,49
LUTRAM	617	6000	10,28
FF	19924	35200	56,60
BRAM	25,50	60	42,50
DSP	36	80	45,00
IO	43	100	43,00
BUFG	7	32	21,88
MMCM	1	2	50,00
PLL	1	2	50,00



## **7. Podsumowanie i możliwe dalsze kierunki rozwoju systemu**

Zgodnie z celem pracy został opracowany system detekcji przechodniów na podstawie obrazu z kamery termowizyjnej. Mając do dyspozycji kamerę Lepton o bardzo małej rozdzielczości (80x60 pikseli) zdecydowano się na wykorzystanie obrazu wizyjnego i detekcję obiektów w przestrzeni multispektralnej (RGBIR). W tym celu, zgodnie z założeniem, wykorzystano heterogeniczny układ Zynq-7000 umożliwiający sprzętowo-programową implementację algorytmów. Zadaniem części logiki programowej (FPGA) było połączenie strumieni wizyjnych z kamer w jeden obraz multispektralny oraz określenie obszaru zainteresowania (ROI) do klasyfikacji. W celu określenia ROI został wykorzystany słabszy klasyfikator: wzorzec probabilistyczny – opracowany przez autora w ramach pracy inżynierskiej. Wytypowany kandydat był następnie klasyfikowany przy użyciu deskryptora HOG i SVM. Proces ten odbywał się w systemie procesorowym układu Zynq. Drugim zadaniem systemu procesorowego była konfiguracja parametrów modułów zaimplementowanych w logice programowej oraz wizualizacja wyników. Dodatkowa funkcja zapisania obrazów na karcie SD dała możliwość stworzenia własnej bazy próbek do nauczenia klasyfikatora.

Zastosowana kamera termowizyjna umożliwia akwizycję 9 klatek na sekundę i jest źródłem obrazu dla modułu odpowiedzialnego za określanie ROI. Ponieważ system procesorowy potrzebuje 75 ms na klasyfikację pojedynczego ROI, a kolejne ramki obrazu pojawiają się co około 111 ms możliwa jest tylko jedna detekcja na klatkę obrazu IR. Nie jest to imponujący wynik.

W celu poprawy szybkości działania można by przenieść obliczanie deskryptora HOG związaną z systemem procesorowym do logiki programowej, ale nie pozwalają na to ograniczone zasoby logiczne używanego układu Zynq. Na podstawie modelu programowego można stwierdzić, że w porównaniu do poprzedniego systemu spadła liczba fałszywych pozytywnych detekcji.

Aktualnie próg binaryzacji obrazu termowizyjnego jest stały i wymaga ręcznej korekty w przypadku zmiany warunków pracy kamery. Kolejnym krokiem w rozwoju systemu byłoby dodanie modułu automatycznego dobierania tego progu. Przez małą ilość zasobów układu za-

stosowanego w platformie Zybo, detekcja wzorcem probabilistycznym jest wykonywana tylko w jednej skali. Realizacja systemu z użyciem większego układu pozwoliłaby na dodanie kolejnych wzorców, co poszerzyłoby zakres detekcji o inne skale. Dodanie śledzenia pozwoliłoby określić kierunek poruszania się obiektu. Na jego podstawie system wspomagania kierowcy mógłby zareagować, gdyby wykryty przechodźień mógł wtargnąć przed pojazd.

## Bibliografia

- [1] Rikke Gade i Thomas B Moeslund. „Thermal cameras and applications: A survey”. W: *Machine vision and applications* 25.1 (2014), s. 245–262.
- [2] Ji Hoon Lee i in. „Robust pedestrian detection by combining visible and thermal infrared cameras”. W: *Sensors* 15.5 (2015), s. 10580–10615.
- [3] Louis St-Laurent, Xavier Mal dague i Donald Prévost. „Combination of colour and thermal sensors for enhanced object detection”. W: *Information Fusion, 2007 10th International Conference on*. IEEE. 2007, s. 1–8.
- [4] Soonmin Hwang i in. „Multispectral pedestrian detection: Benchmark dataset and baseline”. W: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, s. 1037–1045.
- [5] Branislav Kisacanin, Shuvra S Bhattacharyya i Sek Chai. *Embedded computer vision*. Springer Science & Business Media, 2008.
- [6] Gabriel J García i in. „A survey on FPGA-based sensor systems: towards intelligent and reconfigurable low-power sensors for computer vision, control and signal processing”. W: *Sensors* 14.4 (2014), s. 6247–6278.
- [7] Frank Niklaus, Christian Vieider i Henrik Jakobsen. „MEMS-based uncooled infrared bolometer arrays: a review”. W: *Photonics Asia 2007*. International Society for Optics i Photonics. 2007, s. 68360D–68360D.
- [8] Wikipedia. *Infrared*, Wikipedia, The Free Encyclopedia. [Dostęp: 9 stycznia 2016]. 2016.
- [9] LEPTON® Long Wave Infrared (LWIR) Datasheet. Lepton. FLIR Commercial Systems. 2015.
- [10] Shanshan Zhang, Rodrigo Benenson i Bernt Schiele. „Filtered channel features for pedestrian detection”. W: *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*. IEEE. 2015, s. 1751–1760.

- [11] Alejandro González i in. „Pedestrian detection at day/night time with visible and FIR cameras: A comparison”. W: *Sensors* 16.6 (2016), s. 820.
- [12] Rodrigo Benenson i in. „Ten years of pedestrian detection, what have we learned?” W: *arXiv preprint arXiv:1411.4304* (2014).
- [13] Navneet Dalal i Bill Triggs. „Histograms of oriented gradients for human detection”. W: *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*. T. 1. IEEE. 2005, s. 886–893.
- [14] Frédéric Suard i in. „Pedestrian detection using infrared images and histograms of oriented gradients”. W: *Intelligent Vehicles Symposium, 2006 IEEE*. IEEE. 2006, s. 206–212.
- [15] Timo Ojala, Matti Pietikainen i Topi Maenpaa. „Multiresolution gray-scale and rotation invariant texture classification with local binary patterns”. W: *IEEE Transactions on pattern analysis and machine intelligence* 24.7 (2002), s. 971–987.
- [16] Paul Viola i Michael Jones. „Rapid object detection using a boosted cascade of simple features”. W: *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*. T. 1. IEEE. 2001, s. I–I.
- [17] SR Kodituwakku i S Selvarajah. „Comparison of color features for image retrieval”. W: *Indian Journal of Computer Science and Engineering* 1.3 (2004), s. 207–211.
- [18] Pedro Felzenszwalb, David McAllester i Deva Ramanan. „A discriminatively trained, multiscale, deformable part model”. W: *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE. 2008, s. 1–8.
- [19] A Criminisi, J Shotton i E Konukoglu. „Decision Forests for Classification, Regression, Density Estimation, Manifold Learning and Semi-Supervised Learning [Internet]”. W: *Microsoft Research* (2011).
- [20] Tomasz Kańska. „Wbudowany system wizyjny do detekcji pieszych w warunkach nocnych”. Akademia Górnictwo-Hutnicza im. Stanisława Staszica w Krakowie, 2016.
- [21] Karol Piniarski i Paweł Pawłowski. „Efficient pedestrian detection with enhanced object segmentation in far IR night vision”. W: wrz. 2017, s. 160–165.
- [22] Dominik Honegger, Helen Oleynikova i Marc Pollefeys. „Real-time and low latency embedded computer vision hardware based on a combination of FPGA and mobile CPU”. W: *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*. IEEE. 2014, s. 4930–4935.
- [23] Songlin Piao i in. „Real-time multi-platform pedestrian detection in a heavy duty driver assistance system”. W: *Proc. Int. Commercial Veh. Technol. Symp.* 2016, s. 61–70.

- [24] Han Xiao i in. „Real-time shape and pedestrian detection with FPGA”. W: *Mechatronics and Automation (ICMA), 2015 IEEE International Conference on*. IEEE. 2015, s. 2381–2386.
- [25] Tomasz Kańka, Tomasz Kryjak i Marek Gorgon. „FPGA Implementation of Multi-scale Pedestrian Detection in Thermal Images”. W: *Image Processing & Communications* 21.3 (2016), s. 55–67.
- [26] Guillaume-Alexandre Bilodeau i in. „Thermal-visible registration of human silhouettes: A similarity measure performance evaluation”. W: *Infrared Physics & Technology* 64 (2014), s. 79–86.
- [27] Harsh Nanda i Larry Davis. „Probabilistic template based pedestrian detection in infrared videos”. W: *IEEE Intelligent Vehicle Symposium*. T. 1. 2002, s. 15–20.
- [28] Xianzhi Du i in. „Fused DNN: A deep neural network fusion approach to fast and robust pedestrian detection”. W: *Applications of Computer Vision (WACV), 2017 IEEE Winter Conference on*. IEEE. 2017, s. 953–961.



## **A. Dodatek A – zawartość płyty CD**

- /doc – tekst pracy dyplomowej,
- /src – użyte skrypty i funkcje napisane w języku Matlab,
- /bootimage – plik rozruchowy dla platformy Zybo,
- /vivado – projekt w programie Vivado.



## B. Dodatek B – Instrukcja obsługi systemu

Aplikacja uruchamia się z karty SD. Kamerę lepton należy podłączyć do następujących pinów portu PMOD JD Zybo:

- CS – JD1\_P,
- MOSI – JD1\_N,
- MISO – JD2\_P,
- CLK – JD2\_N.

Po podpięciu platformy Zybo przez złącze mikro USB uzyskujemy dostęp do terminalu poprzez UART. Parametry portu: baud rate 115200, 8 bitów, jeden bit stopu, brak kontroli parzystości.

Wciśnięcie klawisza 'h' powoduje pokazania się karty pomocy:

```
HELP:  
1 VDMA status  
2 Start detection  
3 Restart DMA  
  
5 read T matrix  
6 write T matrix  
7 write bin threshold  
8 update LP threshold  
  
w save next detected ROI  
s save centered ROI  
x save center ROI and HOG discriptor  
d save whole frame  
m mount SD card  
q get calibration image
```

Wprowadzenie znaku z powyższej listy do konsoli powoduje wykonanie następującej akcji:

- 1 – wyświetlenie statusu VDMA,
- 2 – rozpoczęcie detekcji przechodniów,
- 3 – restart DMA,
- 5 – wyświetlenie parametrów macierzy transformacji projekcyjnej,
- 6 – pozwala na wprowadzenie nowej macierzy T. Dane należy wprowadzić w postaci 9 liczb z przecinkiem (kropka) np.: wprowadzenie "1.0 0.0 0.0 0.0 1.0 0.0 0.0 0.0 1.0" zamienia macierz T na macierz jednostkową,
- 7 – pozwala na wprowadzenie progu binaryzacji dla modułu DPM,
- 8 – pozwala na wprowadzenie progu LP dla modułu DPM,
- w – zapisanie następnego wykrytego ROI w postaci pliku ROInnn.CIR gdzie nnn to kolejne numery plików.
- s – zapisanie ROI znajdującego się na środku obrazu w postaci pliku ROInnn.CIR gdzie nnn to kolejne numery plików.
- x – zapisanie ROI znajdującego się na środku obrazu w postaci pliku ROInnn.CIR oraz wektora cech w postaci pliku FVnnn.FLO nnn to kolejne numery plików.
- d – zapisanie całej ramki obrazu w postaci pliku FRAMEnnn.CIR gdzie nnn to kolejne numery plików.
- m – zamontowanie karty SD w celu zapisu plików,
- q – zapis obrazu kalibracyjnego w postaci pliku CALIBR.CIR. Spowoduje to zapisanie obrazu wizyjnego wraz z niepoddanym transformacji obrazem termowizyjnym.

Pliki .CIR można otworzyć za pomocą funkcji [IrImage, rgbImage] = cir2image( filename ) w pakiecie MatLab (znajdującym się na płycie CD w folderze /src). Zwraca ona dwie macierze: IrImage – zapisany obraz w podczerwieni oraz rgbImage zapisany obraz wizyjny.

Pliki .FLO składa się z 1584 32-bitowych liczb zmiennoprzecinkowych typu float. Reprezentuje on obliczony przez procesor wektor cech związanego z nim ROI.

## C. Dodatek C – Opis informatyczny projektu

Projekt w pakiecie Matlab:

- cir2image.m – funkcja odczytująca pliki obrazu multispektralnego .CIR,
- cir2imageLegacy.m – funkcja odczytująca pliki obrazu multispektralnego .CIR w starszej wersji,
- cir2bmp – skrypt tworzący obraz RGB z nałożonym obrazem termowizyjnym w pseudokolorze,
- /PHOTOS – przykładowe zdjęcia,
- /KALIBRACJA – folder zawierający skrypt do kalibracji wraz z przykładowym obrazem,
- /HOGSVM – folder zawierający skrypty do obliczania deskryptora, oraz do nauki na przykładowym zestawie danych,
  - HOGSVM.m – skrypt do nauki klasyfikatora,
  - extractFeatVectfromROI.m – funkcja do obliczania wektora cech z ROI,
  - getHistogram.m – funkcja do obliczania histogramów,
  - showCIR.m – funkcja do szybkiego podglądu pliku .CIR,
  - HOGverification.m – skrypt do sprawdzenia różnicy między wektorami cech obliczonymi przez ZYBO a PC,
  - /ROIs – folder zawierający ROI do nauki klasyfikatora,
  - /Verification – folder zawierający pliki do sprawdzenia przez skrypt HOGverification,
- /SampleCIR – folder zawierający przykładowe pliki .CIR,

Projek w środowisku Vivado:

- IRcam\_Driver.v – kontroler kamery termowizyjnej Lepton,

- box.v – moduł dodający do strumienia wizyjnego kolorowe, prostokątne obramowanie wokół wskazanego obszaru,
- ir12\_rgb24\_to\_rgb24\_colorised.v – moduł zamieniający obraz strumień obrazu RGBIr na RGB z nałożonym Ir w pseudokolorze,
- axis\_video\_combiner.v – moduł łączący strumień RGB z IR,
- AXI\_projection\_module – IP-Core zawierający moduł realizujący transformację projekcyjną,
- PDM40 – IP-Core realizujący detekcję za pomocą wzorca probabilistycznego,
- DPM\_data\_graber.v – moduł przygotowujący dane dla modułu DPM40,
- bilinear\_interpolation – IP-Core realizujący interpolację obrazu IR,
- coordinates\_counter.v – moduł generujący strumień współrzędnych dla modułu transformacji projekcyjnej,
- mono16b\_signal\_normaliser – IP-Core realizujący moduł normalizujący strumień IR.