

POLITECHNIKA WARSZAWSKA
WYDZIAŁ ELEKTRYCZNY
INSTYTUT STEROWANIA I ELEKTRONIKI PRZEMYSŁOWEJ

PRACA MAGISTERSKA
na kierunku ELEKTROTECHNIKA



Daniel CHADE
Nr imm.189269

Rok. akad. 2007/2008
Warszawa, 12.10.2007

SAMOŁOKALIZACJA ROBOTA MOBILNEGO NA PODSTAWIE ANALIZY OBRAZU Z
POJEDYNCZEJ KAMERY

Zakres pracy:

1. *Wstęp*
2. *Wybrane elementy przetwarzania obrazów*
3. *Przegląd oraz analiza porównawcza wybranych metod wizyjnej lokalizacji robotów mobilnych*
4. *Implementacja wybranej metody wizyjnej samolokalizacji robota mobilnego*
5. *Podsumowanie i wnioski*

Podpis i pieczęć

Kierownika Zakładu Dydaktycznego

Opiekun naukowy:
Dr inż. Witold Czajewski

Termin wykonania: 15.09.2008
Praca wykonana i zaliczona pozostaje
własnością Instytutu i nie będzie
zwrócona wykonawcy

Samolokalizacja robota mobilnego na podstawie obrazu z pojedynczej kamery

Streszczenie

Głównym celem pracy było stworzenie systemu wizyjnego lokalizacji robota z użyciem pojedynczej kamery. Do lokalizacji robota została użyta metoda sztucznych znaczników z rodziny metod bezwzględnych. Polega ona na rozmieszczeniu rozróżnialnych dla algorytmu symboli geometrycznych w środowisku robota mobilnego. Robot wyposażony w kamerę porusza się po otoczeniu. Jeśli w polu widzenia kamery pojawi się odpowiedni symbol, wtedy robot jest w stanie określić swoje położenie względem niego. Dzięki bazie danych o położeniu poszczególnych znaczników, staje się możliwe również określenie pozycji robota w pomieszczeniu.

Szczególny nacisk w całym systemie, został położony na stworzenie odpowiedniego algorytmu przetwarzania obrazu, który jest odpowiedzialny za wyznaczenie przestrzennej lokalizacji oraz identyfikację znacznika. Wymagało to udoskonalenia pierwotnej wersji programu, w stosunku do której został zautomatyzowany proces progowania. Udoskonalano również proces filtracji o użycie metody momentów geometrycznych, a także wprowadzono możliwość rozpoznawania zestawu ośmiu znaczników. Algorytm został napisany w języku C++, przy zastosowaniu zbioru bibliotek do przetwarzania obrazu ze środowiska OpenCV 2.0.

Drugim aspektem pracy było zastosowanie zaproponowanego systemu wizyjnego w praktyce. W tym celu został zaimplementowany system sterowania robotem mobilnym, łączący dane otrzymywane z kamery razem z danymi otrzymywanymi poprzez czujniki odometryczne. Jako robot mobilny użyty został Pioneer 3DX wyposażony w kamerę Creative Live Cam Optia AF. Algorytm sterowania robotem został napisany od podstaw i umożliwia jego samolokalizację w nieznanym miejscu w pomieszczeniu, a następnie przemieszczanie się po ustalonej trasie na podstawie sensorów odometrycznych. Ponieważ pozycja wyznaczana przez odometrię jest obciążona błędem podczas pokonywania trasy, algorytm koryguje ją używając precyzyjniejszej kamery. Algorytm sterowania, podobnie jak algorytm rozpoznawania obrazu, został napisany przy użyciu języka C++, wykorzystując dodatkowo zbiór bibliotek do sterowania robotem ARIA (*ang. Advanced Robotics Interface for Applications*).

Przeprowadzone testy działania potwierdziły skuteczność systemu. Spełnia on swoją rolę i umożliwia samolokalizację oraz korygowanie położenia, dzięki czemu robot jest w stanie poruszać się cały czas po wyznaczonej trasie. Podczas testów ujawniła się również zawodność metody odometrycznej, której błąd kumuluje się wraz z pokonanym dystansem. Poleganie jedynie na odczytach z sensorów odometrycznych z pewnością uniemożliwiłoby zachowanie poprawnego toru jazdy, co wskazuje na praktyczność zastosowania korekcji wizyjnej.

Wykonana praca ukazała, iż kamera może z powodzeniem stanowić precyzyjny czujnik lokalizujący robota oraz wspomagający działanie odometrii. Stworzony system sterowania robotem umożliwia jego adaptację zarówno do różnych pomieszczeń, jak i typów robotów mobilnych. Dodatkowo istnieje możliwość rozbudowy systemu, zarówno od strony sprzętowej jak i programowej. Przykładami udoskonaleń mogą być: zaimplementowanie algorytmu doboru wstępnych parametrów obrazu, zastosowanie sonarów oraz użycie dwóch kamer jednocześnie.

Self-localization of a mobile robot using a view from single camera

Abstract

Main aim of the thesis was to develop a computer vision system resolving mobile robot self-localization issue, using a view from single camera. For that purpose the method of artificial landmarks was used. The idea of the method was to place a set of recognizable geometric symbols in the mobile robot environment. The robot equipped with a digital camera, using image processing operations, was able to localize itself against the landmarks. Due to created database, that contained environment map and information, about location of different landmarks, it was possible to estimate robot's position in the surroundings.

Main emphasis of work was placed on the development of computer vision algorithm with a set of different fully recognizable landmarks. All system was made, based on basic version of the image processing algorithm, which needed some improvements. During the thesis work process of thresholding was fully automated, filtration process was improved with using the geometric moments and set of different fully recognizable landmarks was created. Entire algorithm was developed using C++ language with a set of libraries for image processing OpenCV 2.0.

Second aspect of the thesis was to show practical usage of the system in the real robot environment. This aim needed the development of control algorithm for the robot, which was built from scratch, using C++ language with a set of libraries ARIA (*Advanced Robotics Interface for Applications*). As mobile robot, Pioneer 3DX equipped with camera Creative Live Cam Optia AF was used. Algorithm purpose was to localize the robot in the environment and subsequently move it through projected route. The entire system used both vision and odometry sensor readings. As part of the work map of the environment and database containing information about position of particular landmarks were created.

Conducted tests confirmed effectiveness of the vision system. It made possible self-localization of the robot and verifying its position, when following the route. During the tests, results received from odometry sensors were compared to these obtained from the vision. This comparison showed accumulation of error for the odometry and very precise verification of position with the vision system.

Entire work leaves system open for further improvements. Examples of possible extensions are: usage of ultrasonic sensors to avoid obstacles during the movement, installation of additional camera to analyze two views simultaneously. It also worth to underline, that created system can be adapted both to different robot environment and another types of mobile robots.

Spis treści

1. Wstęp.....	8
1.1. Cel pracy	8
1.2. Układ pracy.....	8
2. Metody lokalizacji robotów mobilnych.....	10
2.1. Względne metody lokalizacji.....	10
2.2. Bezwzględne metody lokalizacji.....	14
2.3. Metody jednoczesnej lokalizacji i mapowania (SLAM).....	20
3.1. Cyfrowa reprezentacja obrazów.....	24
3.2. Histogram obrazu cyfrowego.....	28
3.3. Filtracja obrazu cyfrowego.....	29
3.4. Operacje morfologiczne.....	36
3.5. Segmentacja	39
3.6. Momenty geometryczne bezwładności.....	42
3.7. Kamera cyfrowa.....	43
4. Opis algorytmu przetwarzania obrazu użytego do lokalizacji robota mobilnego	51
4.1. Budowa znacznika.....	51
4.2. Akwizycja obrazu	53
4.3. Kondycjonowanie obrazu.....	53
4.4. Segmentacja sceny wizyjnej.....	54
4.5. Analiza obrazu.....	58
4.6. Testy działania algorytmu	60
5. Zastosowanie systemu wizyjnego w środowisku robota mobilnego	66
5.1. Opis robota mobilnego Pioneer 3DX.....	66
5.2. Opis algorytmu sterowania robotem mobilnym w pomieszczeniu laboratorium	67
5.3. Testy działania algorytmu sterowania robotem mobilnym	74
6. Podsumowanie	77
Referencje	80
Spis Załączników.....	82

1. Wstęp

Wzrok jest powszechnie uznawany za najbardziej istotny zmysł człowieka. Dostarcza on ogromu informacji do mózgu o otaczającym świecie: poprzez rozpoznawanie kształtów, ocenianie odległości położenia obiektów czy rozróżnianie barw. Zwykle ludzie nie zdają sobie sprawy, jak bardzo skomplikowanym procesem jest to co nazywamy patrzeniem i postrzeganiem. Wynika to po części z faktu, iż odbywa się on w sferze podświadomości w sposób naturalny od momentu narodzin. Dopiero próba stworzenia sztucznego odpowiednika takiego procesu, ujawnia całą jego złożoność. W każdym sztucznym systemie wizyjnym, oprócz kamery działa skomplikowany algorytm analizy obrazu, którego zadaniem jest identyfikacja szukanych cech na obrazie, a jego funkcję można porównać do działania mózgu człowieka.

System wizyjny, jako sztuczny odpowiednik wzroku jest w stanie realizować dokładnie te same funkcje czyli: rozpoznawać barwy, kształty, szacować odległość na potrzeby systemów sterowania najróżniejszego typu. Obecnie najpopularniejsze zadania zautomatyzowanych systemów wizyjnych to: kontrola jakości w produkcji przemysłowej, systemy biometryczne analizujące kształt linii papilarnych lub siatkówki oka, identyfikacja obiektów, szacowanie względnego położenia oraz tworzenie map otoczenia w systemach mobilnych [15].

1.1. Cel pracy

Zadaniem pracy było zbudowanie systemu mającego na celu szacowanie pozycji robota mobilnego w terenie, używając do tego metody sztucznych znaczników. W tym celu należało udoskonalić podstawową wersję algorytmu przetwarzania obrazu o automatyzację procesu doboru progu, usprawnić algorytm filtracji oraz stworzyć zestaw rozróżnialnych znaczników. Dodatkowo należało przedstawić możliwość praktycznego wykorzystania stworzonego systemu wizyjnego w warunkach pracy robota, czyli zaimplementować algorytm sterowania robotem współpracującym z systemem wizyjnym w taki sposób, by robot mógł wielokrotnie i skutecznie realizować zadaną trajektorię ruchu, bez istotnego spadku precyzji lokalizacji.

1.2. Układ pracy

Wykonaną pracę można podzielić na dwie niezależne części: teoretyczną oraz praktyczną. Część teoretyczna (Rozdziały 2, 3) opisuje obecnie dostępne metody lokalizacji robotów mobilnych oraz wprowadza w tematykę cyfrowego przetwarzania obrazu. Część praktyczna (Rozdziały 4, 5) opisuje sposób działania stworzonego systemu wizyjnego oraz przedstawia propozycję jego implementacji w rzeczywistym środowisku pracy robota. Dodatkowo należy podkreślić, że integralną część pracy stanowi stworzony i działający w praktyce system lokalizacji robota.

Praca została podzielona na sześć rozdziałów w których:

Rozdział 2 wprowadza w tematykę różnych metod lokalizacji robotów mobilnych

Rozdział 3 opisuje aspekty cyfrowego przetwarzania obrazu

Rozdział 4 przedstawia sposób działania stworzonego systemu cyfrowego przetwarzania obrazu

Rozdział 5 opisuje system sterowania robotem mobilnym łączący odczyty z systemu wizyjnego oraz sensorów odometrycznych

Rozdział 6 podsumowuje całość wykonanej pracy oraz proponuje dalsze usprawnienia stworzonego systemu.

2. Metody lokalizacji robotów mobilnych

Lokalizacja robota mobilnego w otoczeniu jest aktualnie jednym, z głównych obszarów badań w robotyce. Z perspektywy robota sprowadza się do odpowiedzi na proste z pozoru pytanie „Gdzie jestem i dokąd zmierzam?”. Jak się jednak okazuje, nie jest to zadanie trywialne, a do lokalizacji robotów obecnie stosowany jest szeroki wachlarz metod, wykorzystujących najrozmaitsze techniki. Niniejszy rozdział ma na celu sklasyfikowanie oraz przedstawienie zasad działania tych najbardziej popularnych.

Większość dostępnych metod lokalizacji robotów mobilnych można przyporządkować do jednej z trzech podstawowych grup:

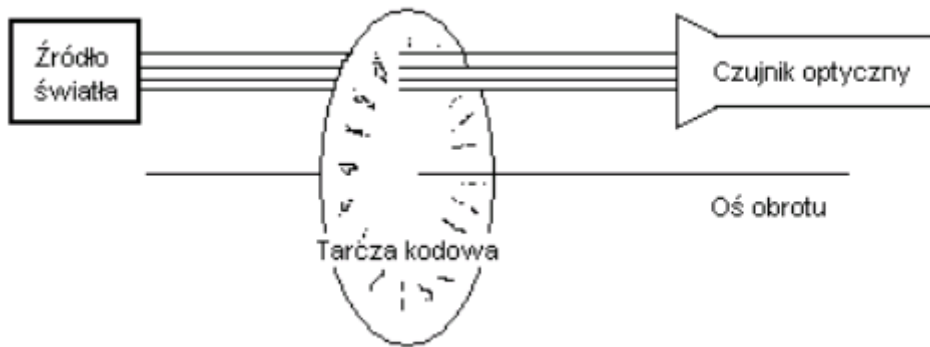
- **metody względne** – są to metody wykorzystujące wewnętrzne parametry robota takie jak: prędkość kół, przyspieszenie, kierunek jazdy, przechył
- **metody bezwzględne** – metody wykorzystujące parametry z otoczenia zewnętrznego robota takie jak: odległość od przeszkody, natężenie światła, temperatura otoczenia
- **metody jednoczesnej lokalizacji i mapowania (SLAM)** – są to metody względnie najnowsze i aktualnie rozwijane, wykorzystujące do lokalizacji wiele różnych czujników, łączące znane do tej pory metody względne i bezwzględne

2.1. Względne metody lokalizacji

Nazwa względne metody lokalizacji robota pochodzi od zasady działania, w której pozycja robota mierzona jest względem poprzedniej pozycji, jaką robot zajmował. Cechą wszystkich metod względnych jest to, iż mierzą jedynie wewnętrzne parametry poruszania się robota takie jak: prędkość kół, przyspieszenie, kierunek jazdy, przechył. Estymowane w ten sposób położenie nie odnosi się w żaden sposób do istniejących w terenie rzeczywistych obiektów. Wadą takiego podejścia jest to, iż błąd lokalizacji kumuluje się przez cały czas i jest niemożliwy do skorygowania. Obecnie w robotyce mobilnej wśród metod względnych najbardziej rozpowszechnione są odometria oraz metody inercyjne.

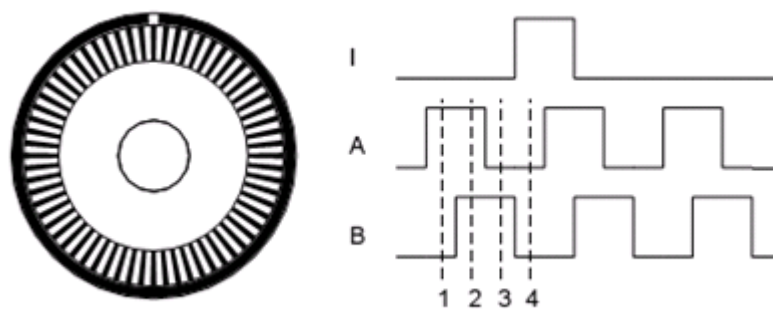
Odometria

Sposób działania metody odometrycznej polega na pomiarze kąta obrotu kół napędowych lub wałów silników. Na tej podstawie następnie wyliczana jest rzeczywiście przejechana odległość oraz skręt, jaki robot wykonał podczas drogi. W przypadku wykorzystywanych w laboratorium robotów z serii Pioneer 3DX jako sensor odometryczny zastosowano enkodery inkrementacyjne optyczne. Enkoder jest to tarcza tachometryczna (kodowa) ze specjalnymi otworami na osi koła, która wskutek obrotu przecina wiązkę światła skierowaną na fotodetektor przechwytyjący i zliczający impulsy świetlne (*Rysunek 2-1*). Kąt obrotu koła jest proporcjonalny do liczby wytworzonych i przechwyconych przez fotodetektor impulsów światła.



Rysunek 2-1. Zasada działania czujnika odometrycznego[19]

Aby możliwe było określenie kierunku obrotu tarczy w układzie elektronicznym, dodaje się dodatkowy kanał generujący impulsy przesunięte o 90 stopni w fazie względem pierwszego kanału. Jak widać na rysunku (**Rysunek 2-2**) kierunek obrotu koła można określić badając czy sygnał z kanału A wyprzedza w przesunięciu fazowym sygnał z kanału B, czy jest odwrotnie.



Rysunek 2-2. Zasada działania enkodera inkrementacyjnego z możliwością odróżnienia kierunku obrotu [1]

Zaletami metody odometrycznej są relatywnie niski koszt i łatwość implementacji. Odometria bywa również bardzo zawodna i podatna na błędy, a najprostszym przykładem jej zawodności jest poślizg któregoś z kół. W takim przypadku koło ślizgając się wykonuje większą liczbę obrotów nieproporcjonalną do przejechanej odległości. Może to spowodować zarówno błąd w określeniu pozycji robota jak i jego orientacji. Ponadto metoda ta jest całkowicie nieodporna na awarię zasilania, a w takiej sytuacji pozycja robota jest tracona bez możliwości ponownego odtworzenia.

Błędy w metodzie odometrycznej możemy sklasyfikować jako błędy systematyczne oraz błędy niesystematyczne. Błędy systematyczne to takie, które wynikają z niedokładności konstrukcji systemu. Ich cechą charakterystyczną jest to, iż są błędami stosunkowo łatwymi do przewidzenia i możliwymi do wyeliminowania wskutek właściwej kalibracji robota. Przykładowo rozpatrując sytuację, w której jedno z kół ma nieznacznie inną średnicę niż drugie (np. ze względu na niższe ciśnienie w oponie), robot wykonując taką samą liczbę obrotów obydwoma kołami będzie skręcał, podczas gdy w „normalnych” warunkach

powinien poruszać się do przodu. Możliwe wówczas staje się wyeliminowanie takiego błędu uwzględniając różnicę średnic kół. W takim przypadku mniejsze koło wykonywałoby więcej obrotów proporcjonalnie do różnicy średnic rozmiarów obydwu kół. Błędy niesystematyczne spowodowane są interakcją robota ze środowiskiem zewnętrznym. Zazwyczaj powodują mniejszy błąd w stosunku do błędów systematycznych, natomiast przypadkowość ich występowania sprawia, że są błędami niezwykle trudnymi do przewidzenia i skorygowania. Z tego powodu pozycję robota wyznaczoną przez odometrię zwykle koryguje się wykorzystując inne metody bezwzględne.

Przyczyny błędów systematycznych:

- niejednakowa średnica kół
- błąd pomiaru średnicy koła
- błąd pomiaru rozstawu kół
- skrzywienie kół
- ograniczona rozdzielczość enkodera
- ograniczona częstotliwość próbkowania

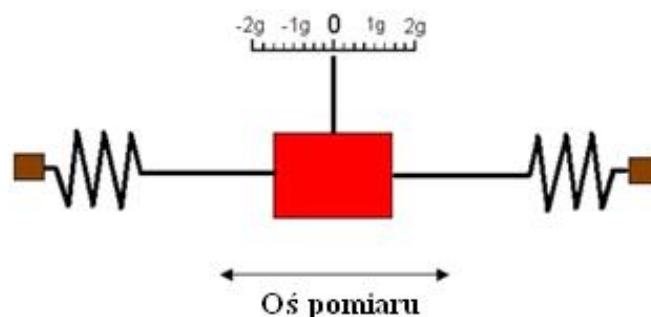
Przyczyny błędów niesystematycznych:

- przemieszczanie po nierównym podłożu
- przemieszczanie po nieprzewidzianych obiektach na podłożu
- poślizg kół spowodowany:
 - ✓ śliskim podłożem
 - ✓ gwałtownym przyspieszeniem
 - ✓ poślizgiem podczas szybkiego skręcania
 - ✓ działaniem sił zewnętrznych

Nawigacja inercyjna

Nawigacja inercyjna jest alternatywną metodą dla odometrii. Istota działania metody polega na pomiarze przyspieszenia lub momentu obrotowego wytwarzanego podczas wykonywania skrętów. W skład typowego systemu nawigacji inercyjnej wchodzi zestaw akcelerometrów i żyroskopów, w którym akcelerometry są odpowiedzialne za pomiar przesunięcia robota wzdłuż określonej osi, natomiast żyroskopy mierzą jego rotację[16].

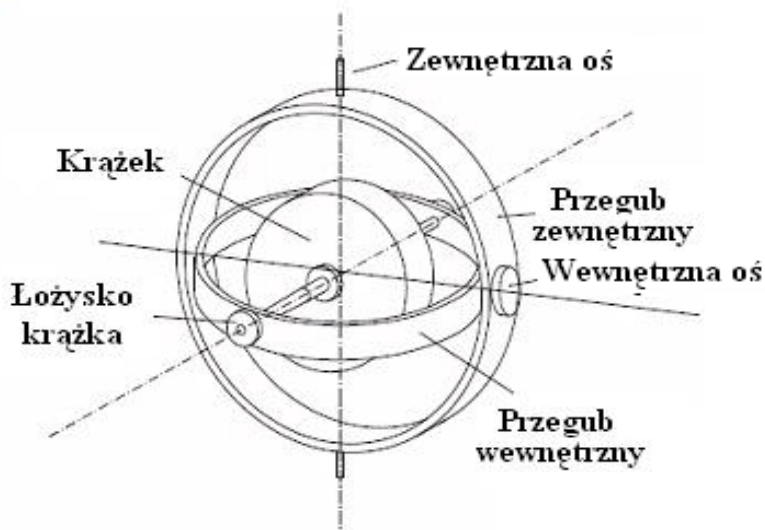
Idea akcelerometru polega na wykorzystaniu bezwładnej masy, oddziałującej na tłumiki z siłą proporcjonalną do przyspieszenia (**Rysunek 2-3**). W systemie nawigacji inercyjnej, akcelerometry wykorzystywane są do pomiaru przesunięcia wzdłuż określonej osi, które jest otrzymywane w wyniku operacji podwójnego całkowania przyspieszenia po czasie. Dla określenia położenia robota na płaszczyźnie, konieczne jest zastosowanie dwóch akcelerometrów, a w przypadku przestrzeni trójwymiarowej - trzech.



Rysunek 2-3. Zasada działania akcelometru z drgającą masą

Obecnie na rynku oprócz analogowych akcelometrów z drgającą masą, dostępne są również cyfrowe akcelerometry wykorzystujące materiały piezoelektryczne, generujące ładunek elektryczny proporcjonalny do przyspieszenia. Inna konstrukcja cyfrowego akcelometru polega na wykorzystaniu pojemności elektrycznej pomiędzy materiałami. W tym przypadku w wyniku działania przyspieszenia następuje zmiana odległości pomiędzy materiałami stanowiącymi okładki kondensatora, powodująca zmianę pojemności elektrycznej, która jest proporcjonalna do odległości pomiędzy okładkami[17].

Do wspomagania działania akcelometrów w celu określenia orientacji robota, używane są żyroskopy. Mechaniczny żyroskop (**Rysunek 2-4**) ma postać krążka, który raz wprowadzony w ruch obrotowy zachowuje pierwotne położenie osi obrotu, z niewielkimi ruchami, precesyjnymi które są uwzględniane w określaniu kierunku lub są eliminowane przez tłumienie [9].



Rysunek 2-4. Mechaniczny żyroskop[1]

Próba obrotu, szybko wirującego się koła w żyroskopie powoduje wytworzenie momentu obrotowego, proporcjonalnego do szybkości obrotu koła zgodnie ze wzorem:

$$\tau = I * \omega * \Omega$$

Gdzie:

T – wytworzony moment obrotowy

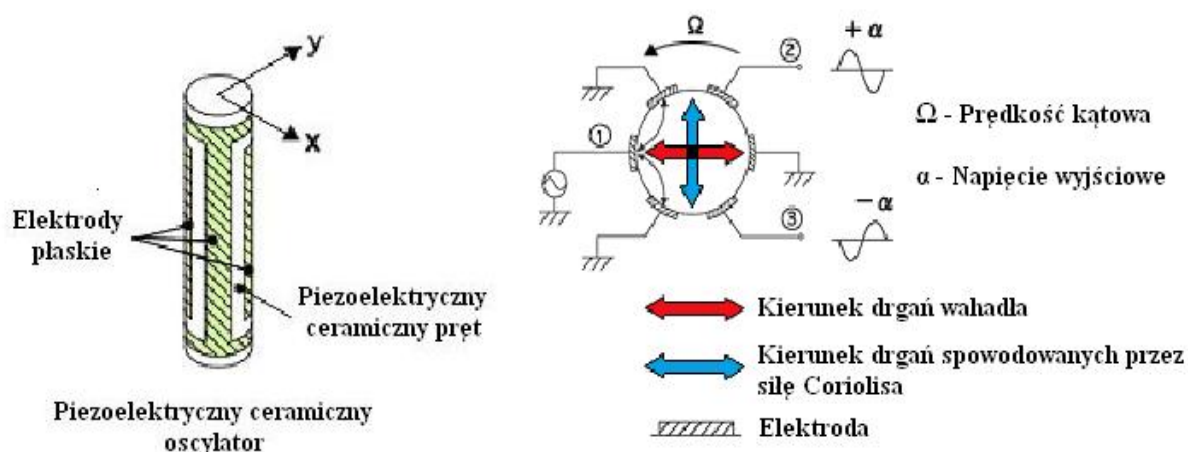
I – moment bezwładności koła

ω - prędkość obrotu koła

Ω – szybkość ruchu precesyjnego koła

Kąt obrotu jest proporcjonalny do całki po czasie z mierzonego momentu obrotowego.

W robotyce mobilnej oprócz mechanicznych żyroskopów, bardzo popularne są również żyroskopy cyfrowe. W tym przypadku do pręta zbudowanego z piezoelektrycznego materiału ceramicznego, za pośrednictwem elektrod doprowadzone jest napięcie w taki sposób, że pręt zaczyna drgać i pracować jak wahadło Foucalt'a. Następnie siła Coriolisa powstająca w wyniku obrotu wahadła powoduje wytworzenie napięcia elektrycznego na odpowiednio umieszczonych elektrodach wyjściowych, proporcjonalnego do prędkości obrotu. [11]



Rysunek 2-5. Budowa żyroskopu cyfrowego[11]

Istotną zaletą metod inercyjnych jest możliwość otrzymywania wyników pomiarów z bardzo niewielkim opóźnieniem. Dodatkowo używane sensory są zależne jedynie od siebie, nie emitują żadnych zakłóceń do otoczenia oraz są niezawodne, potrafią być również bardzo dokładne. Stosując metody inercyjne należy mieć na uwadze fakt, iż pomiary pozycji robota z ich udziałem wykorzystują operację całkowania, która zwielokrotnia błąd, jaki wynikałby z dokładności sensora. Główną wadą nawigacji inercyjnej jest jej wysoki koszt, który sprawia, że metoda rzadko kiedy znajduje zastosowanie w robotyce mobilnej na rzecz odometrii.

2.2. Bezwzględne metody lokalizacji

Bezwzględne metody nawigacji robotów oprócz sensorów obecnych na pokładzie robota wykorzystują obiekty znajdujące się w jego otoczeniu. Istotną zaletą takiego podejścia jest to, że błędy metody nie kumulują się w czasie, a błąd w określeniu położenia robota jest jedynie błędem wynikającym z dokładności konstrukcji użytego sensora. Z tego powodu metody bezwzględne często używane są do korekcji pomiaru metod względnych.

Dalmierze

Dalmierze obecnie są bardzo popularnymi urządzeniami lokalizującymi, wykorzystywanymi w robotyce mobilnej, gdzie wyznaczają odległość robota od przeszkód występujących w jego otoczeniu. Zasada działania dalmierzy jest bardzo prosta i polega na wysyłaniu fali fizycznej określonego rodzaju (zależnie od rodzaju dalmierza może to być fala elektromagnetyczna lub ultradźwiękowa), która odbija się od przeszkody i wraca z powrotem po określonym czasie. Na podstawie pomiaru czasu pomiędzy wysłaniem a powrotem fali, określana jest odległość pomiędzy robotem a przeszkodą. Określa ją wzór:

$$d = v * t$$

Gdzie:

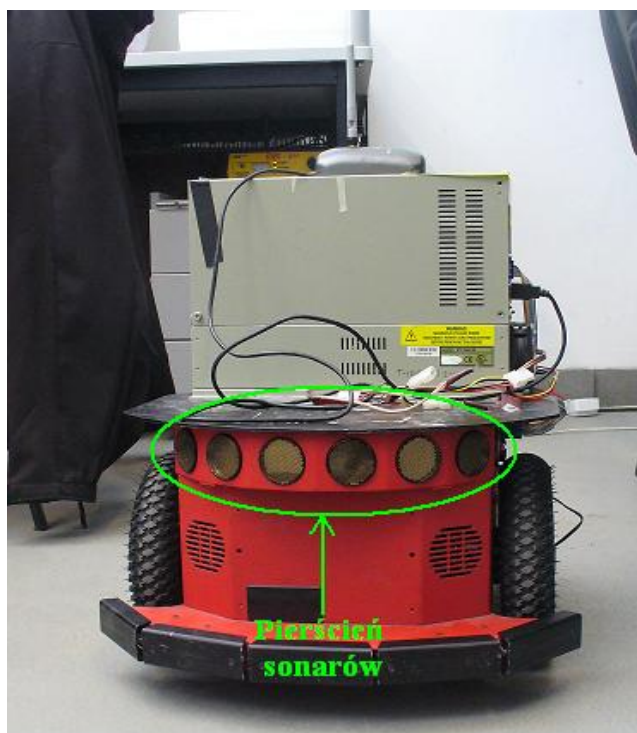
d – podwójny dystans dalmierza od przeszkody

v - szybkość propagacji fali w danym środowisku

t - czas mierzony od wysłania fali do jej powrotu

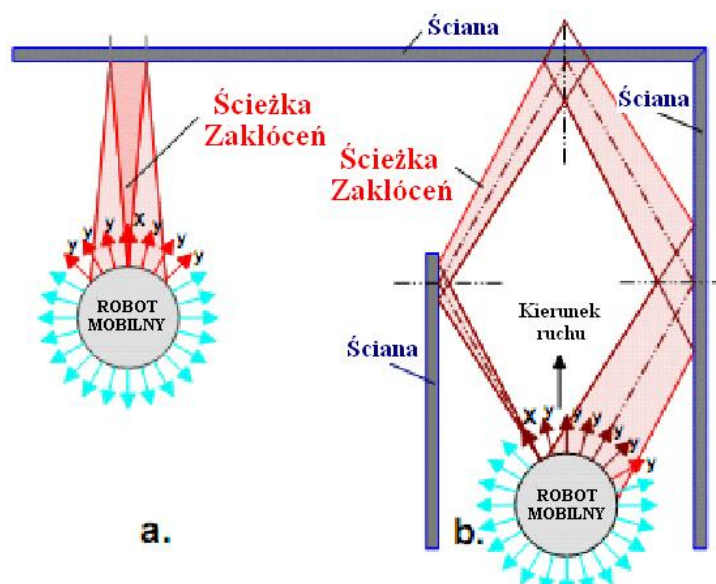
Spośród dostępnych rodzajów dalmierzy, obecnie najpowszechniejsze są dalmierze sonarowe oraz dalmierze laserowe, tzw. lidary.

Sonar do pomiaru dystansu od przeszkody używa fali ultradźwiękowej. Czujnik ten jest obecnie najbardziej popularnym urządzeniem stosowanym jako dalmierz w lokalizacji robota, głównie z powodu niewielkiego kosztu. Sensory sonarowe wykorzystywane są między innymi w robotach Pioneer z serii 3DX, gdzie zastosowano ich dwa zespoły (tzw. pierścienie) skierowane pod różnymi kątami na obudowie robota (**Rysunek 2-6**). Dzięki zastosowaniu pierścieni sonarowych robot jest w stanie określić odległość od przeszkód w terenie w różnych kierunkach.



Rysunek 2-6. Robot z serii Pioneer 3DX z Sonarami

Główną wadą metody sonarowej jest możliwość wielokrotnego odbicia fali sonaru od kilku różnych przeszkód lub przechwycenie fali wysłanej z jednego sonaru przez inny, co dodatkowo może spowodować błąd pomiarowy (**Rysunek 2-7**). Stosując tę metodę, należy mieć również na uwadze jej wysoki szum pomiarowy i możliwość zakłócania przez nią działania innych sonarów znajdujących się w pobliżu. Działanie sonaru ponadto może być zakłócone przez inne urządzenia znajdujące się w pobliżu emitujące fale dźwiękowe o częstotliwościach zbliżonych do częstotliwości sonaru. Do takich urządzeń zalicza się między innymi silniki i pompy [1].



Rysunek 2-7. Błąd pomiarowy sonaru w skutek wielokrotnego odbicia[1]

Idea metody laserowej jest zbliżona do działania sonaru – polega na pomiarze czasu pomiędzy wysłaniem fali, a jej powrotem w wyniku odbicia od przeszkody. Kwestią odróżniającą dwie metody jest to, iż zamiast fali dźwiękowej sonaru używana jest laserowa wiązka elektromagnetyczna. Główną zaletą wykorzystania lasera jest mniejsza podatność na zakłócenia, główną wadą natomiast jest wysoka cena, która powoduje, że dalmierz laserowy nie jest tak popularny jak sonarowy. Innym istotnym ograniczeniem zastosowań dalmierza laserowego jest to, że światło lasera źle odbija się od pewnych powierzchni np. materiałów przezroczystych.

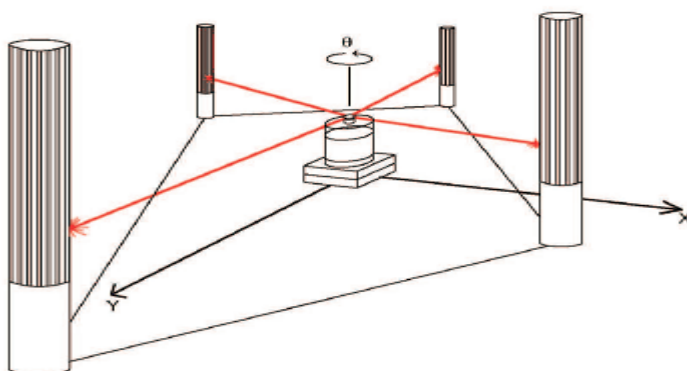
Laser namiarowy

Laser namiarowy jest rozwiązaniem wykorzystującym obrotową głowicę z laserem (**Rysunek 2-8**) oraz zestaw znaczników o znanych współrzędnych w pomieszczeniu.



Rysunek 2-8. Obrotowa głowica wraz z laserem namiarowym

Głowica lasera obracając się wokół, rejestruje położenie kątowe pomiędzy poszczególnymi znacznikami i na tej podstawie wyznaczana jest pozycja oraz orientacja robota. Tego rodzaju system rozwinięty przez firmę Transitions Research Corporation (**Rysunek 2-9**) jest w stanie lokalizować robota w rejonie płaszczyzny kwadratu o boku 24,4 metra. W procesie kalibracji systemu robot początkowo potrzebuje wszystkich czterech znaczników, podczas gdy podczas normalnej pracy wystarczą jedynie dwa do określenia pełnej pozycji robota. [1]



Rysunek 2-9. Laser namiarowy z systemem czterech znaczników

Metoda nawigacji za pomocą znaczników

Nawigacja przy pomocy znaczników jest metodą w skład której wchodzi: kamera wizyjna umieszczona na robocie, pewna liczba znaczników oraz baza danych zawierająca informację o ich położeniu w otoczeniu. Robot przemieszczając się w terenie, znajduje określony znacznik w polu widzenia kamery i dzięki zastosowaniu numerycznego algorytmu przetwarzania obrazu jest w stanie zlokalizować się wobec niego. Ponieważ położenie znaczników w terenie zawarte jest w bazie danych, możliwa staje się lokalizacja bezwzględna w otoczeniu. Cały proces lokalizacji wykorzystuje skomplikowane algorytmy przetwarzania obrazu: filtrację, segmentację i analizę obrazu. Z racji faktu, iż poniższa praca wykorzystuje nawigację za pomocą znaczników, szczegółowy opis działania takiego algorytmu przedstawiona jest w rozdziale 5-tym pracy.

Znaczniki mogą stanowić dowolne przedmioty o znanych dla robota wymiarach geometrycznych, które można sklasyfikować na dwie zasadnicze grupy:

- **znaczniki naturalne** – czyli obiekty, które nie wymagają żadnej ingerencji w otoczenie robota i występują naturalnie w środowisku jego działania np. drzwi, okna, ściany
- **znaczniki sztuczne** – korzystające z rozmaitych sztucznych symboli geometrycznych, które celowo zostały wprowadzone do otoczenia w celu lokalizacji robota

Dokładność metody potrafi być również zależna od szeregu czynników [1]:

- parametry otoczenia takie jak oświetlenie, widoczność, mogą znacząco wpływać na rozpoznawanie znaczników. Im gorsza widoczność, tym trudniejsze dla algorytmu rozpoznanie znacznika
- dokładność pomiaru zależy od odległości i kąta między robotem i znacznikami. Im mniej miejsca na uchwyconym obrazie zajmuje znacznik, tym trudniejsze dla algorytmu jest właściwa identyfikacja oraz dokładne określenie położenia robota wobec znacznika.
- parametry wewnętrzne kamery takie jak rozdzielczość i jakość wykonania kamery istotnie wpływają na dokładność procesu lokalizacji
- właściwy rodzaj wstępnej filtracji jest kluczowy i powinien zostać dobrany dokładnie do otoczenia w jakim działa robot: jasność, kontrast, korekcja gamma, filtry cyfrowe

Z uwagi na skomplikowane metody przetwarzania obrazu, aby możliwe było działanie metody w czasie rzeczywistym konieczne staje się wykorzystanie szybkiej maszyny numerycznej, potrafiącej przetwarzać na bieżąco dużą ilość danych. Konieczne również staje się utrzymywanie i aktualizowanie bazy danych o znacznikach.

Globalny System Pozycjonujący (GPS)

Obecnie GPS jest jednym z najbardziej rozpowszechnionych systemów nawigacji, używanym w środkach transportu drogowego, wodnego i powietrznego. Metoda GPS do określenia pozycji korzysta z pomocy trzydziestu jeden satelitów (stan na 05.09.2009 r. [2]) krążących wokół Ziemi oraz odbiornika zamontowanego w obiekcie lokalizowanym.

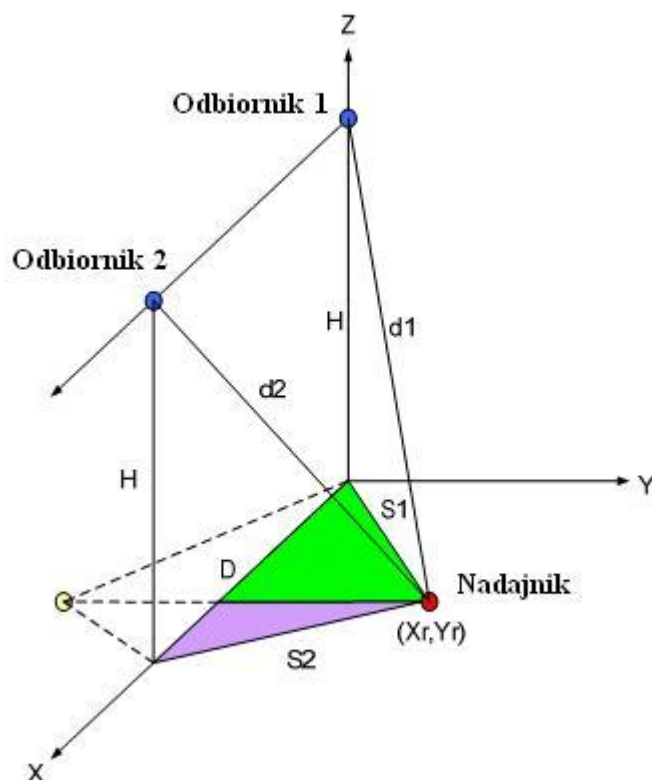
Idea metody polega na przesyłaniu przez satelity do odbiornika, sygnału radiowego zawierającego czas oraz swoją dokładną pozycję na orbicie. Na podstawie pomiaru czasu, znając prędkość fali, wyliczana jest odległość robota od poszczególnego satelity. Aby zlokalizować obiekt w przestrzeni trójwymiarowej oraz precyzyjnie zsynchronizować czas, potrzeba aby w polu zasięgu odbiornika znajdowały się cztery różne satelity. Dokładna pozycja i czas są następnie określane poprzez rozwiązanie układu równań. W praktyce do zlokalizowania obiektu na płaszczyźnie wystarczą dwa satelity.

Cechy nawigacji GPS:

- dokładność pomiaru rzędu kilku metrów
- podatność na zakłócenia elektromagnetyczne
- niemożliwość działania w pomieszczeniach zamkniętych np. w tunelach czy budynkach, z uwagi na fakt, iż sygnał generowany przez nadajniki GPS jest zbyt słaby
- brak możliwości określenia orientacji robota, a jedynie kierunek przemieszczania

Z uwagi na fakt, iż metoda korzystająca z globalnego systemu pozycyjnego, ma dokładność pomiaru rzędu kilku metrów, jej stosowanie w małym środowisku laboratoryjnym

jest niepraktyczne. Bazując na metodzie globalnego systemu pozycjonującego, możliwe stało się skonstruowanie wewnętrznego systemu GPS [10]. W tym przypadku nadajniki GPS umieszczone na satelitach krążących wokół Ziemi, zostają zastąpione przez odbiorniki umieszczone w znanej lokalizacji w pomieszczeniu. Cały system korzysta w dwóch odbiorników oraz nadajnika zamontowanego na robocie. Zasada działania jest analogiczna do tradycyjnego zewnętrznego systemu GPS, jednak zamiast fali radiowej system wykorzystuje falę ultradźwiękową. Na podstawie pomiaru czasu pomiędzy wysłaniem, a odebraniem sygnału, określana jest odległość robota od odbiornika. Mając dwa odbiorniki o znanych współrzędnych można określić położenie robota na płaszczyźnie.



Rysunek 2-10. Zasada działania GPS w pomieszczeniu[10]

Dokładna pozycja robota (X_r, Y_r) jest otrzymywana w wyniku rozwiązania poniższego układu równań:

$$d^2 = H^2 + S^2$$

$$S1^2 = Xr^2 + Yr^2$$

$$S2^2 = (D - Xr)^2 + Yr^2$$

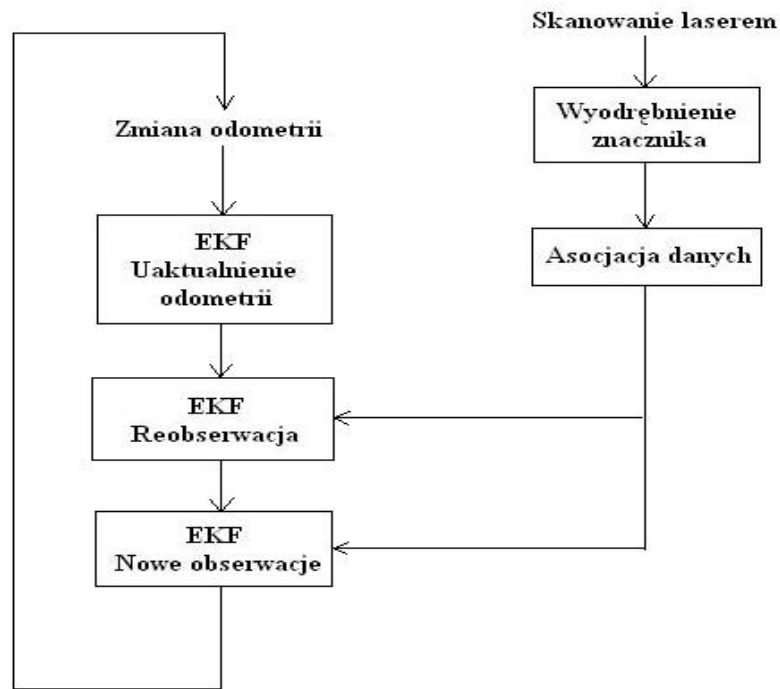
Istotną cechą wewnętrznego systemu GPS jest znacznie większa dokładność lokalizacji w stosunku do standardowej odmiany. W przypadku testów przeprowadzonych w środowisku laboratoryjnym o powierzchni prostokątnej o bokach 3,5m i 2,2m, otrzymane błędy mieściły się w przedziale $\pm 2\text{cm}$ [10]. Ponadto system jest względnie niedrogi, co czyni go dobrym rozwiązaniem dla lokalizowania robotów mobilnych w pomieszczeniach zamkniętych. Do wad systemu można zaliczyć to, że wymaga ingerencji w środowisko, w którym należy zamontować odbiorniki GPS.

2.3. Metody jednoczesnej lokalizacji i mapowania (SLAM)

Metody jednoczesnej lokalizacji i mapowania SLAM (*ang. Simultaneous Localization and Mapping*) są relatywnie najnowszymi metodami używanymi do lokalizacji robotów, w stosunku do metod opisanych w poprzednich paragrafach rozdziału. SLAM stanowi pewnego rodzaju ideę i może zostać zaimplementowany w robocie na wiele różnych sposobów. Najróżniejsze warianty metody wykorzystują rozmaite sensory oraz algorytmy matematyczne [3], zależnie od aktualnego położenia robota czy warunków panujących wokół. Wspólną cechą wszystkich metod z rodziny SLAM jest tworzenie mapy w nieznanym dla robota wcześniej otoczeniu, połączone z lokalizacją robota. W poniższym paragrafie przedstawione są przykłady implementacji kilku wariantów metody. Należy podkreślić, iż nie są to wszystkie znane konfiguracje, a możliwości są ograniczone jedynie przez wyobraźnię inżynierów tworzących system.

SLAM z wykorzystaniem lasera

Metoda opisana w pracy [3], wykorzystuje odometryczny czujnik przesunięcia oraz laserowy czujnik odległości. Jako aparat matematyczny metody, użyty jest algorytm rozszerzonej filtracji Kalmana (EKF). Stosowany czujnik laserowy posiada możliwość obrotu w pewnym zakresie kątów. Robot przemieszczając się wyznacza swoją pozycję na podstawie odometrii, a równolegle obszar skanowany jest czujnikiem laserowym, za pomocą którego budowana jest mapa otoczenia złożona ze ścian, przeszkód, krawędzi, wśród której zaznaczane są pewne charakterystyczne punkty – znaczniki. Na podstawie znaczników robot jest w stanie zweryfikować swoją pozycję, wyznaczaną przez odometrię. Znaczniki zwykle stanowią krawędzie lub miejsca, gdzie występują przedmioty o specyficznym kształcie, dające bardzo charakterystyczny odczyt dla laserowego czujnika. Schemat działania całego systemu przedstawia rysunek (**Rysunek 2-11**). Warto podkreślić, iż jego algorytm sterujący stanowi rozszerzona filtracja Kalmana, której zadaniem jest powiązanie danych otrzymanych przez laser z danymi otrzymanymi przez sensory odometryczne i na tej podstawie budowanie mapy oraz estymowanie aktualnego położenia robota.



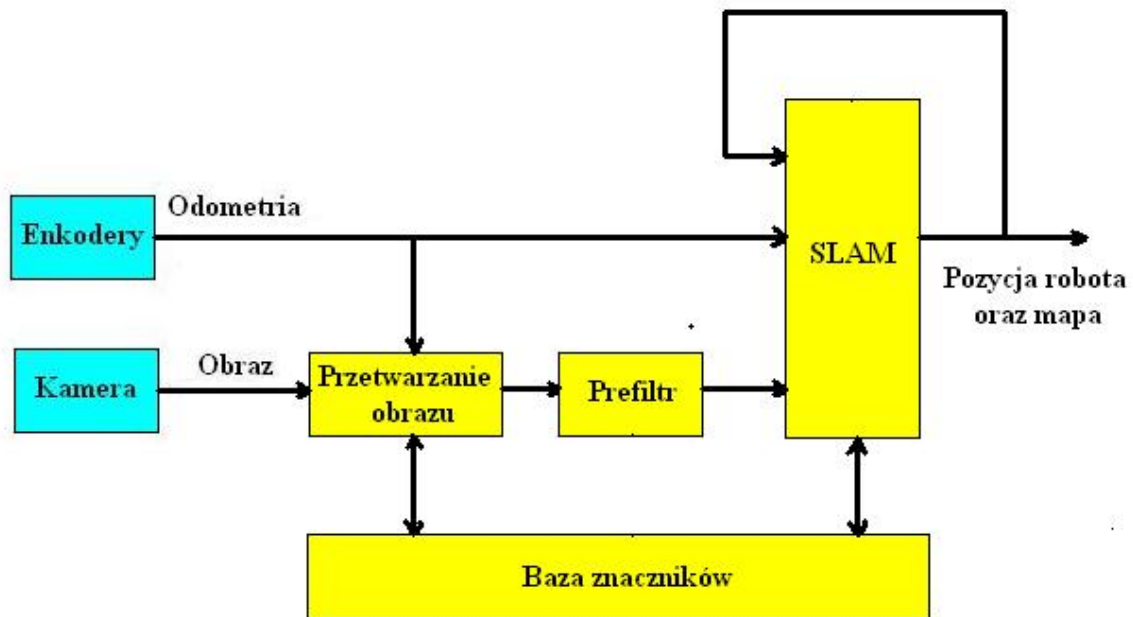
Rysunek 2-11. Schemat działania metody SLAM z wykorzystaniem lasera

vSLAM

Kolejna metoda z rodziny SLAM opisana jest w artykule [4]. Nazwa vSLAM pochodzi od wizji (ang. vision), ponieważ korzysta z kamery sprzężonej z czujnikiem odometrycznym. Metoda działa analogicznie do metody SLAM z wykorzystaniem lasera: podobnie buduje mapę otoczenia i wyszukuje znaczniki, jednak zamiast czujnika laserowego korygującego pozycję i wyszukującego charakterystyczne punkty (znaczniki), używana jest kamera wizyjna. Należy podkreślić, iż w przeciwieństwie do metody lokalizacji przy pomocy znaczników, która jest tematem niniejszej pracy, metoda vSLAM nie posiada pierwotnej bazy znaczników. Wykorzystanie kamery ma szereg zalet w stosunku do laserowego czujnika. Przede wszystkim w przypadku lasera, detekcja znacznika jest ekstremalnie trudna co szczególnie uwidacznia się w przypadku rozległych pomieszczeń, gdzie wszystkie krawędzie będące miejscami charakterystycznymi, używanymi do tworzenia znaczników są do siebie bardzo podobne. Dodatkowo zapisanie wszystkich cech i analizowanie ich w czasie rzeczywistym wymaga bardzo dużej złożoności obliczeniowej. Zastosowanie kamery znacznie ułatwia rozpoznawanie zarówno rozpoznawanie jak i tworzenie nowych znaczników w otoczeniu, nie obciążając w takim stopniu procesora. Wadą zastosowania kamery jest to, iż mapa otoczenia budowana jest jedynie na podstawie odometrii i brak jest na niej obrazu ścian oraz innych obiektów ze środowiska zewnętrznego.

Podstawową rolę w metodzie vSLAM odgrywają znaczniki, które są wykrywane i umieszczane na mapie tworzonej przez robota podczas przemieszczania się. Kamera pełni rolę korygowania mało dokładnego pomiaru odometrycznego, analogicznie jak laser w

metodzie SLAM z wykorzystaniem lasera. Sposób działania metody przedstawia schemat (*Rysunek 2-12*).

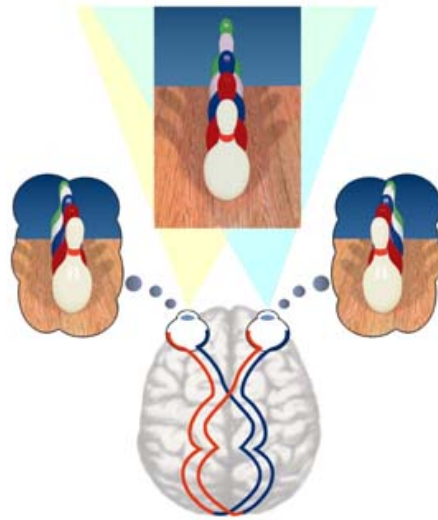


Rysunek 2-12. Schemat działania metody vSLAM

Obraz przechwycony z kamery jest najpierw przetwarzany i porównywany z istniejącą bazą znaczników. Jeśli na obrazie został rozpoznany znacznik istniejący w bazie, wtedy algorytm sterujący stara się skorygować położenie robota na podstawie obrazu znacznika. Jeśli znacznik nie został rozpoznany, następuje próba stworzenia nowego znacznika na podstawie obrazu otoczenia. W przypadku powodzenia tejże operacji, położenie robota zostaje określone na podstawie nowo stworzonego znacznika. Następnie zadaniem prefiltrowania jest skorygowanie czy takowa (obecna) pozycja określona na podstawie nowo stworzonego znacznika spełnia kryteria dokładności. Jeśli tak baza znaczników zostaje uaktualniona, w przeciwnym wypadku znacznik jest odrzucany. Główną część sterującą systemu stanowi moduł SLAM. Jego zadaniem jest przetwarzanie danych wejściowych z sensorów, lokalizacja robota oraz tworzenie mapy otoczenia. Jako aparat matematyczny modułu używany jest rozszerzony filtr Kalmana.

σ SLAM

Metoda σ SLAM [5], jest metodą opartą na systemie stereowizyjnym, zatem korzysta z dwóch kamer skierowanych w jednym kierunku. Zasada systemu stereowizyjnego jest analogiczna do działania pary ludzkich oczu (*Rysunek 2-13*).



Rysunek 2-13. Idea działania stereowizji

Dzięki takiemu rozwiązaniu, możliwa staje się interpretacja widzianych obiektów w przestrzeni trójwymiarowej, a nie w dwuwymiarowej jak ma to miejsce w przypadku pojedynczej kamery. Interpretacja obrazu z dwóch kamer odbywa się za pomocą algorytmu SIFT (*ang. Scale-invariant feature transform*). Metoda jest całkowicie niezależna od mechanicznej odometrii, a jakiegokolwiek przesunięcie czy obrót robota jest możliwy do zinterpretowania jedynie za pomocą systemu stereowizyjnego – wizyjna odometria. Jako aparat matematyczny metody, w tym przypadku została użyta metoda RBPF (*ang. Rao-Blackwellised Particle Filter*), która sprawdziła się znacznie lepiej dla tego przypadku niż EKF. Podobnie jak w przypadku poprzednich metod z rodziny SLAM, robot poruszając się konstruuje mapę otoczenia, wykrywa charakterystyczne punkty, na podstawie których jest w stanie zweryfikować swoją pozycję oraz szacuje swoje przemieszczenie.

3. Elementy przetwarzania obrazów

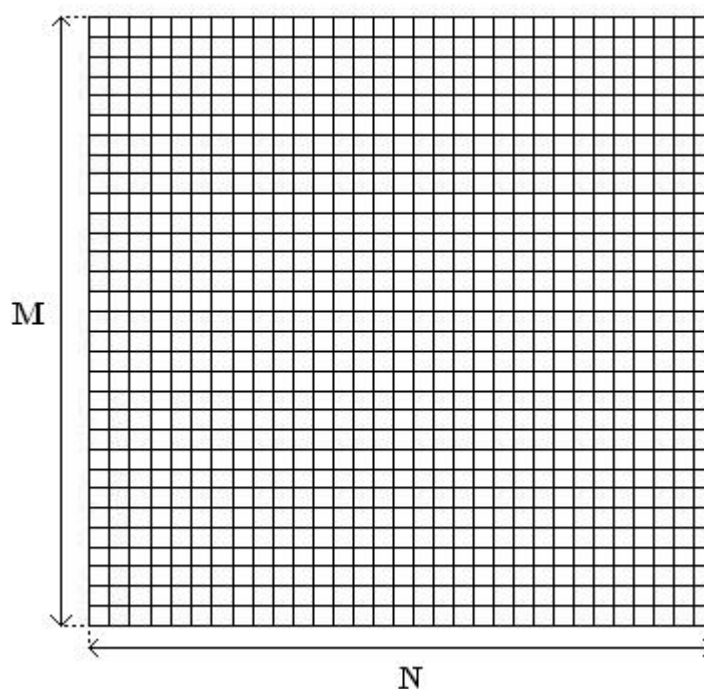
Jedną z podstawowych umiejętności obecnych w codziennym funkcjonowaniu człowieka stanowi analiza bodźców odbieranych przez narządy zmysłów. Powszechnie uważa się, że wzrok jest najistotniejszym zmysłem człowieka, dostarczającym większości informacji o otaczającym go świecie. Z tego właśnie powodu zastąpienie wzroku przez zautomatyzowane systemy wizyjne jest aktualnie jednym z najpilniejszych i najbardziej ambitnych zadań automatyki. Komputerowe przetwarzanie obrazu przeważnie bywa zaskakująco trudnym procesem angażującym bardzo złożone algorytmy numeryczne. Po części wynika to z faktu, że człowiek od momentu narodzin wykonuje czynności patrzenia i postrzegania w sposób naturalny, zupełnie nie zdając sobie sprawy z ich olbrzymiej złożoności. Szacuje się, że mózg człowieka wykonuje przetwarzanie widzianego obrazu co najmniej z szybkością 100 Mb/s [6], co zdecydowanie przerasta możliwości dzisiejszych komputerów. Okiełznanie tak szybkiego strumienia danych jest więc niesłychanie trudne. Z drugiej strony bogactwo informacji niesione przez obraz jest ogromne i właśnie dlatego warto zajmować się automatyzacją procesów wizyjnych.

3.1. Cyfrowa reprezentacja obrazów

W celu efektywnego wykorzystania obrazu jako źródła informacji należy przetworzyć go na postać cyfrową, a następnie przeprowadzić szczegółowy proces analizy. Ponieważ szybkość przetwarzania obrazu przez mózg człowieka jest niedościgniona dla możliwości dzisiejszych komputerów, konieczne staje się zastosowanie szeregu uproszczeń obrazu cyfrowego, takich jak [6]:

- ograniczenie zdolności rozpoznawania szczegółów
- ograniczenie ilości możliwych do rozróżnienia stanów elementu obrazu (kolorów)
- analizowanie obrazu płaskiego zamiast przestrzennego
- analizowanie obrazu statycznego zamiast dynamicznego

Redukcja zdolności rozpoznawania szczegółów na obrazie w postaci cyfrowej, polega na dyskretyzacji obrazu na piksele stosując siatkę punktów. Najczęściej stosowaną obecnie siatką pikseli w technice cyfrowej jest siatka prostokątna (*Rysunek 3-1*). Reprezentowany przez nią obraz jest sprowadzony do prostokątnej płaszczyzny o rozmiarach $M \times N$ określającej liczbę próbkowanych pikseli w pionie i w poziomie. Rozmiar siatki w technice cyfrowej określony jest jako rozdzielczość obrazu. Praktycznie im większa rozdzielczość obrazu, tym analizowany obraz posiada więcej szczegółów. Z drugiej strony większa rozdzielczość obrazu wymaga również większej złożoności obliczeniowej do jego analizy. Dobór odpowiedniej liczby pikseli jest więc zawsze kompromisem pomiędzy szczegółowością, a szybkością analizy obrazu.



Rysunek 3-1. Prostokątna siatka obrazu cyfrowego o rozdzielczości $M \times N$

Bardzo istotnym uproszczeniem stosowanym w cyfrowym przetwarzaniu obrazu, jest ograniczenie liczby kolorów. Podobnie jak w przypadku rozdzielczości obrazu, liczba kolorów jest kompromisem pomiędzy szybkością przetwarzania i ilością zajętej pamięci, a szczegółowością reprezentacji obrazu. Obecnie najpopularniejsze przestrzenie kolorów obrazów stanowią reprezentacje: binarna, monochromatyczna, kolorowa w przestrzeni RGB oraz kolorowa w przestrzeni HSV.

Obraz binarny

W przypadku reprezentacji binarnej każdy piksel na obrazie może przyjmować jedną z dwóch wartości, zwykle przedstawionych jako biel i czerń. Takie rozwiązanie jest idealne do przedstawienia konturów przedmiotów, dlatego obraz binarny jest powszechnie stosowany podczas operacji segmentacji obrazu w cyfrowym przetwarzaniu obrazów (*Rysunek 3-2*). Na co dzień z przestrzenią binarną barw można spotkać w wyświetlaczach LCD prostych urządzeń elektronicznych, takich jak kalkulatory czy starsze modele telefonów komórkowych.



Rysunek 3-2. Przykład binarnego obrazu dłoni

Obraz monochromatyczny

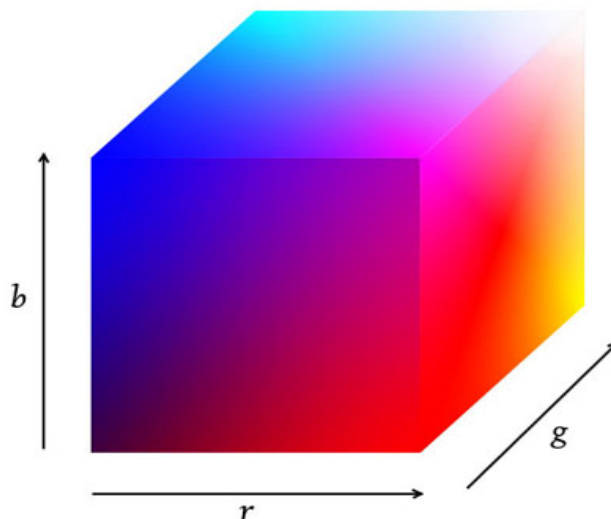
Obraz monochromatyczny najczęściej występuje w postaci obrazu szarościowego (**Rysunek 3-3**) i w przeciwieństwie do obrazu binarnego posiada cały szereg skwantowanych wartości, jakie mogą przyjąć poszczególne piksele. Jego najpopularniejsza ośmiobitowa reprezentacja posiada 256 skwantowanych odcieni szarości. Wartości graniczne to 0 – obraz zupełnie czarny, wartość 255 – obraz biały. Pozostałe wartości są pośrednimi odcieniami. Liczba poziomów jasności jest zawsze potęgą liczby dwa, ponieważ jest reprezentowana przez określoną liczbę bitów przyporządkowanych przez kartę graficzną.



Rysunek 3-3. Przykład obrazu w szarościowego

Kolorowy obraz RGB

Kolorowy obraz w przestrzeni RGB to po części rozszerzenie koncepcji zastosowanej w przypadku obrazu monochromatycznego. Jego istotą jest przypisanie każdemu pikselowi na obrazie trzech składowych o różnych poziomach jasności (zamiast jednej jak ma to miejsce na obrazie monochromatycznym): czerwonej (*ang. red*), zielonej (*ang. green*) oraz niebieskiej (*ang. blue*). Przestrzeń kolorów RGB jest przestrzenią euklidesową (**Rysunek 3-4**), a przypisanie w niej wartości od 0 do 255 każdej składowej, pozwala na uzyskanie ponad 16,5 miliona różnych kolorów. W dzisiejszych czasach, obraz RGB jest najbardziej rozpowszechnioną ze znanych reprezentacji kolorów używaną w większości dostępnych urządzeń cyfrowych takich jak: telewizory, monitory czy karty graficzne komputerów. Geneza przestrzeni RGB związana jest z naturą powstawania obrazu w lampach kineskopowych (CRT). Pobudzony do świecenia fosfor (luminofor) pokrywający wewnętrzną powłokę CRT może emitować światło odpowiadające jednej z barw: czerwonej, zielonej lub niebieskiej. Regulując natężenie strumienia elektronów bombardujących ekran, zmienia się intensywność świecenia odpowiedniego fosforu [12].

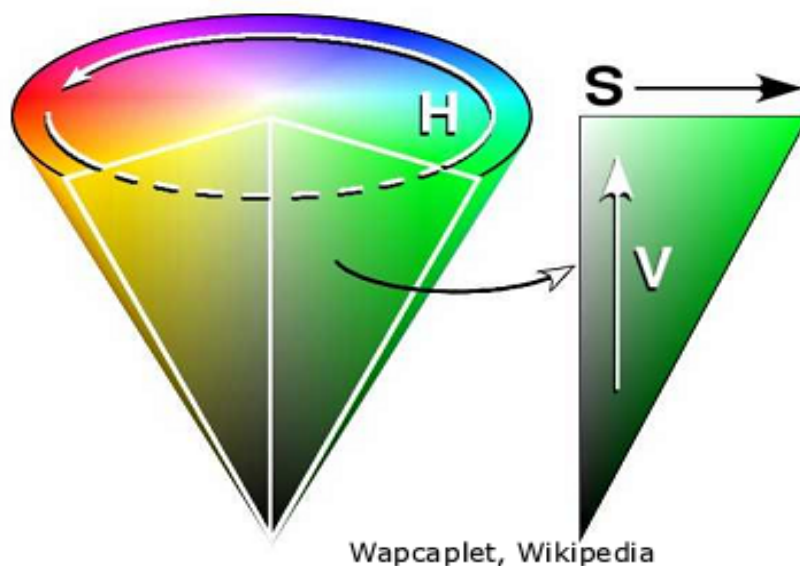


Rysunek 3-4. Przestrzeń barw RGB

Kolorowy obraz HSV

Przestrzeń barw HSV jest przestrzenią biegunową odzwierciedlającą sposób, w jaki artyści malarze mieszają farby na paletce [12]. Jej cechą charakterystyczną jest geometria w kształcie odwróconego stożka. Na obrazie kolorowym reprezentowanym w przestrzeni HSV, każdy piksel (podobnie jak w przestrzeni RGB) posiada trzy skwantowane składowe. W przypadku HSV są to odcień koloru (*ang. hue*) liczony w stopniach, nasycenie (*ang. saturation*) oraz jasność barwy (*ang. value*). Niewątpliwą zaletą kształtu przestrzeni HSV jest łatwość w określaniu obszaru, jaki zajmują poszczególne barwy. Miejsce występowania kolorów jest w niej bardzo regularne. Barwy zielona, czerwona i niebieska, zajmują dokładnie obszar odpowiadający po jednej trzeciej ($1/3$) pola powierzchni stożka (**Rysunek 3-5**). W przypadku przestrzeni kolorów RGB, zdefiniowanie dokładnego obszaru w jakim występują poszczególne barwy nie jest zadaniem trywialnym i z tego powodu przestrzeń HSV bywa

bardzo często stosowana w algorytmach komputerowego przetwarzania obrazów do klasyfikowania koloru poszczególnych pikseli.



Rysunek 3-5. Przestrzeń barw HSV

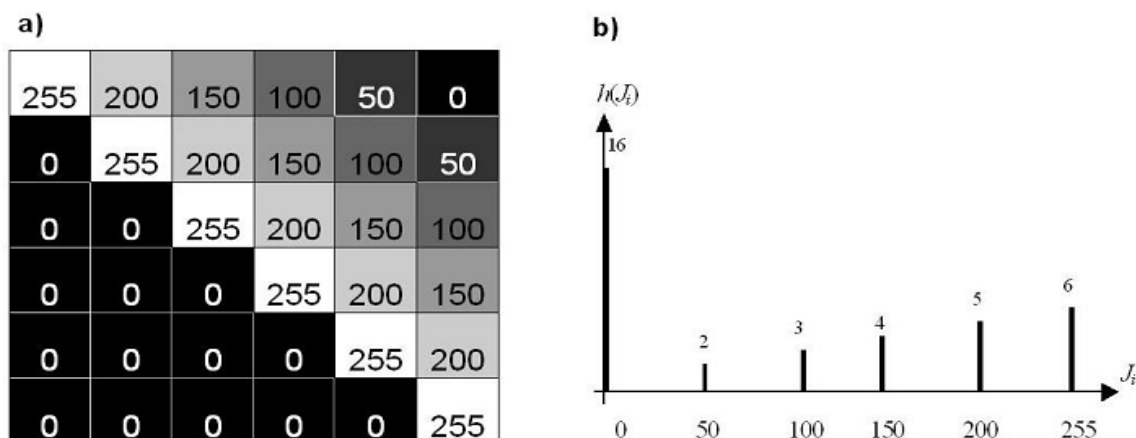
Analizowanie obrazu płaskiego stanowi kolejne ograniczenie w cyfrowym przetwarzaniu obrazów. Z ludzkiego punktu widzenia, człowiek posiadając parę oczu jest w stanie zinterpretować obraz w przestrzeni trójwymiarowej. Podobna sytuacja ma miejsce w przypadku zastosowania kamer wizyjnych i aby uzyskać obraz trójwymiarowy przedmiotu, konieczne jest zastosowanie przynajmniej dwóch kamer. Inną alternatywną metodą jest uzyskanie dwóch klatek obrazu z pojedynczej kamery pod różnymi kątami, co praktycznie ogranicza złożoność systemu wizyjnego.

Przetwarzanie obrazu wysokiej jakości obrazu w czasie rzeczywistym jest w praktyce niemożliwe z uwagi na ograniczone możliwości numeryczne obecnie dostępnych urządzeń. Aby uzyskać dynamiczny, płynny obraz należałoby go analizować z prędkością wynoszącą ok. 25 klatek na sekundę [6], co jest zdecydowanie za dużą liczbą dla współczesnej technologii przetwarzania obrazów. Wobec tego dynamiczna analiza jest ograniczana do kilku klatek na sekundę. Należy mieć również na uwadze, że liczba możliwych klatek do przetworzenia w ciągu sekundy jest uzależniona od rozdzielczości oraz rodzaju zastosowanej reprezentacji kolorów na obrazie.

3.2. Histogram obrazu cyfrowego

Dla obrazów cyfrowych popularną operacją jest wyznaczenie histogramu, czyli rozkładu występowania poszczególnych wartości pikseli. Histogram pokazuje, jak liczne są w obrazie punkty o różnych wartościach jasności (*Rysunek 3-6*). Przykładowo, jeśli rozpatrywanym obrazem jest obraz szarościowy zakodowany przy pomocy ośmiu bitów, to histogram będzie posiadał 256 różnych wartości od 0 do 255 na osi odciętych. Wartością odpowiadającą danemu punktowi, będzie natomiast liczba pikseli o danej jasności na obrazie. W przypadku kolorowego obrazu RGB, popularną operacją jest tworzenie histogramu dla każdej barwy składowej. Histogram obrazu cyfrowego znajduje praktyczne zastosowanie w

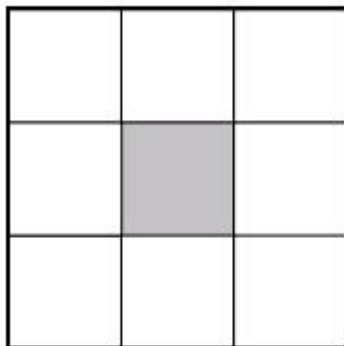
przetwarzaniu obrazów. Za jego pomocą możliwe staje się zinterpretowanie wielu cech uchwyconego obrazu cyfrowego takich jak jasność czy kontrast.



Rysunek 3-6. Przykładowy obraz (a) oraz jego histogram (b)

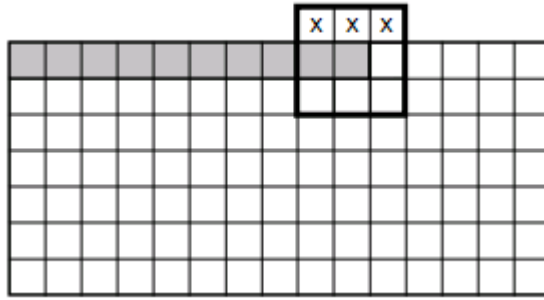
3.3. Filtracja obrazu cyfrowego

Filtry cyfrowe stanowią jedno z bardziej złożonych narzędzi do analizy obrazu. Ich cechą charakterystyczną jest to, iż do określenia wartości pojedynczego piksela, używanych jest wiele różnych pikseli stanowiących otoczenie. Z uwagi na tę zależność, cyfrowe filtry przetwarzania obrazów noszą miano filtrów kontekstowych. Otoczenie obrazu jest definiowane przez maskę filtra, która najczęściej stanowi kwadratowe okno o określonych rozmiarach. Na rysunku (**Rysunek 3-7**) przedstawione jest okno o rozmiarze 3x3, w którym szare pole oznacza analizowany piksel, podczas gdy białe pola stanowią jego sąsiedztwo.



Rysunek 3-7. Przykład maski ośmiospójnej o rozmiarze 3x3 [6]

Ograniczeniem działania filtrów kontekstowych jest brak możliwości wykonywania operacji dla pikseli znajdujących się w zewnętrznym rejonie obrazu (**Rysunek 3-8**). Pomimo tego ograniczenia, filtracja cyfrowa jest bardzo skutecznym narzędziem przetwarzania obrazów, potrafiącym w istotny sposób zmienić zawartość obrazu, w tym geometrię widocznych na obrazie obiektów.



Rysunek 3-8. Przykład maski ośmiospójnej o rozmiarze 3x3[6]

Typowymi zastosowaniami filtrów cyfrowych są:

- usuwanie z obrazu niepożądanego szumu
- wzmocnienie na obrazie pewnych elementów zgodnie z posiadanym wzorcem np. wyostrenie krawędzi
- usunięcie określonych wad z obrazu
- poprawa obrazu o złej jakości technicznej, np. poprawa kontrastu, rozjaśnienie obrazu
- rekonstrukcja obrazu, który uległ częściowemu zniszczeniu

Pod względem wykonywanej operacji matematycznej, filtry cyfrowe można sklasyfikować na:

- filtry liniowe – wykonujące liniową funkcję dla wybranych pikseli obrazu wejściowego
- filtry nieliniowe – wykonujące nieliniową funkcję wybranych pikseli obrazu wejściowego

Filtry liniowe

Aby filtr był zaliczany do filtrów liniowych, jego funkcja realizująca musi spełniać dwa warunki liniowości:

- warunek addytywności

$$\bullet \quad \varphi(f + g) = \varphi(f) + \varphi(g)$$

- warunek jednorodności

$$\bullet \quad \varphi(\lambda f) = \lambda \varphi(f), \lambda \in R$$

Gdzie:

φ – funkcja realizująca filtr

f, g – obrazy podlegające filtracji

W opisie działania filtrów liniowych najwygodniej jest posłużyć się pojęciem konwulcji dyskretnej[6]. Konwulcję dla dwuwymiarowego, dyskretnego obrazu można zapisać w następujący sposób:

$$L'(m, n) = (w \times L)(m, n) = \sum_{i, j \in K} L(m - i, n - j)w(i, j)$$

W tym przypadku cyfrowy obraz o określonych rozmiarach $L(m,n)$ poddawany jest operacji filtracji. Współczynniki $w(i,j)$ są zdefiniowane dla określonego rodzaju filtru i wraz z odpowiednimi elementami obrazu $L(m-i, n-j)$ znajdującymi się w oknie K , służą do obliczenia wartości $L'(m,n)$ w danym punkcie w obrazie wynikowym. Współczynniki $w(i,j)$, są zwykle dobierane jako liczby całkowite z uwagi na fakt, że operacje stałoprzecinkowe wykonywane są przez algorytm dużo szybciej.

Następnie po obliczeniu nowych wartości pikseli konieczne staje się przeprowadzenie operacji normalizacji. Dla filtrów, w których wszystkie współczynniki są większe, bądź równe zero ($w_{i,j} \geq 0$), możliwe jest zastosowanie prostej techniki normalizacji określonej wzorem:

$$L'(m,n) = \frac{1}{\sum_{i,j \in K} w(i,j)} \sum_{i,j \in K} L(m-i, n-j) w(i,j)$$

W przypadku, kiedy filtr wykorzystuje zarówno parametry dodatnie jak i ujemne, normalizację można przeprowadzić według wzoru:

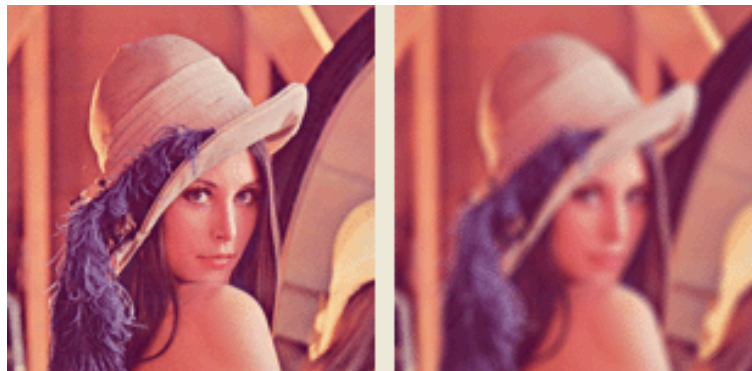
$$L''(m,n) = \frac{L'(m,n) + \min L'(m,n)}{\max L'(m,n) - \min L'(m,n)} * 2^B$$

Obszar otoczenia K oraz wartości elementów wchodzących w jego skład są dowolnie dobierane przez producentów filtrów konwulcyjnych. Ogólnie im większe otoczenie K , tym bardziej radykalne działanie filtru.

Filtry liniowe można podzielić na filtry dolnoprzepustowe oraz filtry górnoprzepustowe. Typowym filtrem dolnoprzepustowym jest filtr uśredniający, który usuwa drobne zakłócenia z obrazu. Efektem ubocznym stosowania filtru uśredniającego jest znaczące rozmycie obrazu (**Rysunek 3-10**).

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Rysunek 3-9. Maska filtru uśredniającego



Rysunek 3-10. Efekt działania filtru uśredniającego

Aby zmniejszyć negatywne skutki rozmywania obrazu przez filtry uśredniające, stosuje się filtry działające w sposób „ważony”. W takim przypadku wartość centralnego punktu maski przyjmuje większą wartość w stosunku do pozostałych pikseli w masce (**Rysunek 3-11**), tak by w bardziej znaczącym stopniu wpływał na wartość piksela na obrazie wynikowym.

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Rysunek 3-11. Maska filtru uśredniającego z większym wzmocnieniem dla punktu centralnego

Zadaniem filtrów górnoprzepustowych, zwanych inaczej gradientowymi jest wydobywanie składników odpowiedzialnych za szybkie zmiany jasności czyli: konturów, krawędzi, drobnych elementów itp. Przeważnie przyjmuje się, że filtry górnoprzepustowe uwypuklają krawędzie obiektów na obrazie. Charakterystyczną cechą filtrów górnoprzepustowych jest wykorzystywanie zarówno dodatnich jak i ujemnych współczynników w masce. Przykładem filtru górnoprzepustowego jest filtr Sobela. Zależnie od kąta pod jakim wykrywane są krawędzie, istnieje kilka jego wariantów (**Rysunek 3-12**, **Rysunek 3-13**, **Rysunek 3-14**), a efekt ich działania można zaobserwować na rysunkach (**Rysunek 3-15**, **Rysunek 3-16**).

$$\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

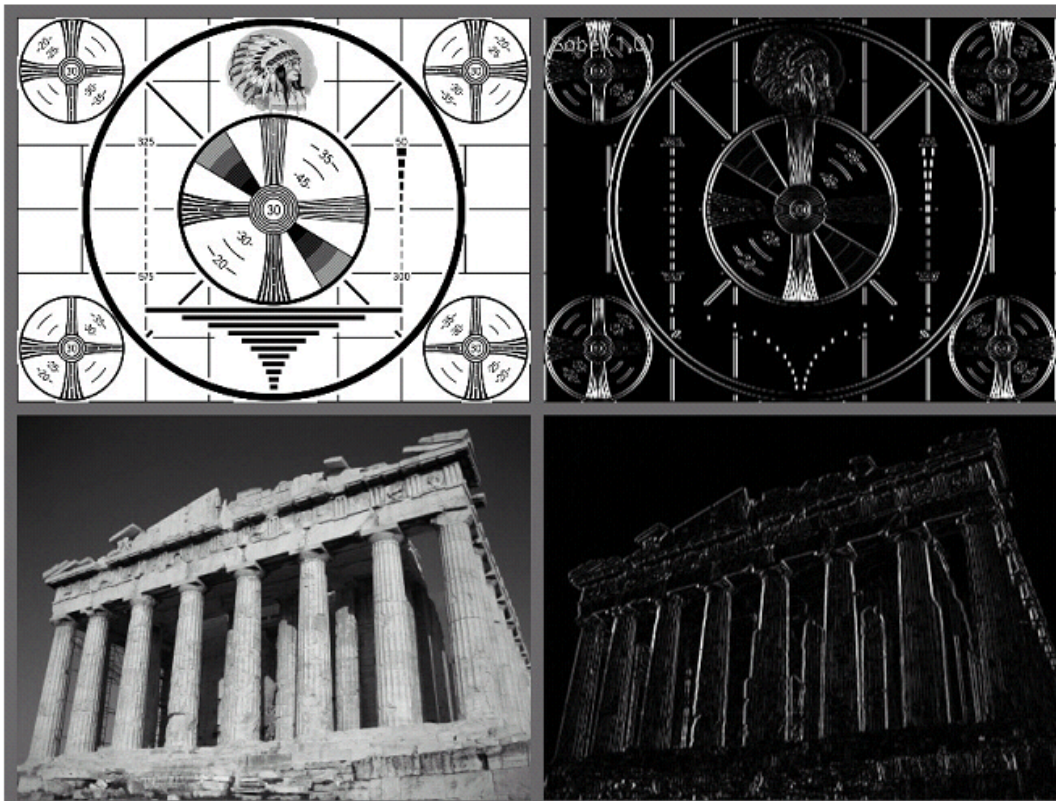
Rysunek 3-12. Maska filtru Sobela dla nachylenia 0° do poziomu

$$\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$$

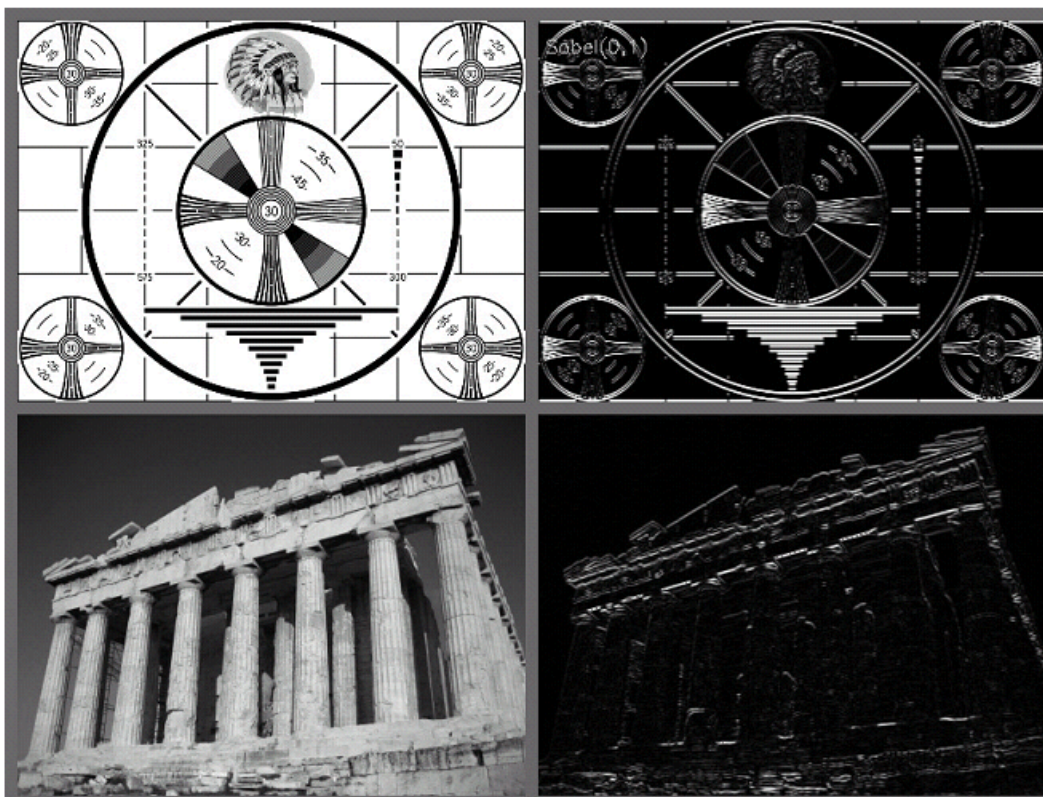
Rysunek 3-13. Maska filtru Sobela dla nachylenia 90° do poziomu

$$\begin{bmatrix} 0 & 1 & 2 \\ -1 & 0 & 1 \\ -2 & -1 & 0 \end{bmatrix}$$

Rysunek 3-14. Maska filtru Sobela dla nachylenia 45° do poziomu



Rysunek 3-15. Przykład zastosowania filtracji Sobela dla nachylenia 0° do poziomemu[7]



Rysunek 3-16. Przykład zastosowania filtracji Sobela dla nachylenia 90° do poziomemu[7]

Filtry nieliniowe

Filtry liniowe, pomimo swoich zalet i mnogości wykonywanych operacji nie nadają się do wszystkich zastosowań. Pełen wachlarz filtrów cyfrowych uzupełniają filtry nieliniowe. Przykładami typowych filtrów nieliniowych są filtry wykorzystujące operacje logiczne, filtr medianowy oraz kombinowane filtry wykrywające krawędzie.

Filtry wykorzystujące operacje logiczne najczęściej stosowane są podczas przetwarzania obrazów binarnych i korzystają z operatorów logicznych takich jak: zaprzeczenie (NOT) (*Rysunek 3-17*), iloczyn logiczny (AND) (*Rysunek 3-18*), suma logiczna (OR), różnica logiczna (SUB), suma rozłączna (XOR) oraz równoważność logiczna (NXOR) (*Rysunek 3-19*). Dla większości operacji logicznych wykorzystywane są dwa obrazy wejściowe tworzące jeden obraz wynikowy.



Rysunek 3-17. Obraz binarny Lena po zastosowaniu jego negacji[6]



Rysunek 3-18. Obraz binarny Lena po zastosowaniu logicznego sumowania z obrazem z udziałem trzech kółek[6]



Rysunek 3-19. Skutki działania operacji SUB, XOR, NXOR[6]

Zastosowanie filtru medianowego jest bardzo skuteczną metodą usuwania szumów. W przeciwieństwie do filtrów liniowych nie powoduje on rozmazywania się obrazu. Istota jego działania polega na zastosowaniu mediany, którą stanowi środkowa wartość w uporządkowanym ciągu wartości jasności pikseli z całego rozważanego otoczenia przetwarzanego piksela. Przykładowo dla maski o wartościach:

$$\begin{bmatrix} 15 & 104 & 37 \\ 19 & 255 & 55 \\ 234 & 89 & 167 \end{bmatrix}$$

Rysunek 3-20. Przykładowa maska poddana filtracji medianowej

Punkty są segregowane w szereg [15,19,37,55,**89**,104,167,234,255], a następnie wybierana jest z nich wartość środkowa: 89 i taką wartość będzie miał piksel na obrazie wynikowym. W przypadku filtru medianowego wielkość sąsiedztwa decyduje o intensywności filtracji. Obraz po zastosowaniu filtracji medianowej nie wymaga dodatkowych operacji skalowania, co jest pewną zaletą, gdyż nie obciąża maszyny numerycznej. Wadą tego rodzaju filtrów jest, że w przypadku szumu niesymetrycznego zmieniają średnią intensywność obrazu. Efekt działania filtracji medianowej przedstawia (**Rysunek 3-21**).



Rysunek 3-21. Przykład zastosowania filtru medianowego do usuwania szumu impulsowego

Kombinacyjne filtry krawędziowe są po części rozszerzeniem koncepcji filtrów górnoprzepustowych z rodziny filtrów liniowych. Efekty działania filtru Sobela dla krawędzi poziomych i pionowych można prześledzić na rysunkach (**Rysunek 3-15**, **Rysunek 3-16**). Analizując obydwa obrazy można łatwo zauważyć, że krawędzie poziome nie są wykrywane przez filtr pionowy i odwrotnie. W celu wyznaczenia krawędzi zarówno poziomych jak i pionowych, na obrazie zwykle stosuje się kombinowane połączenie dwóch obrazów wynikowych, w których jeden został odfiltrowany przez pionowy, a drugi przez poziomy filtr Sobela. Jest to przykład działania nieliniowego filtru kombinacyjnego wykrywającego krawędzie. Do połączenia dwóch obrazów w jeden stosuje się powszechnie formułę euklidesową:

$$L'(m, n) = \sqrt{(L_1(m, n))^2 + (L_2(m, n))^2}$$

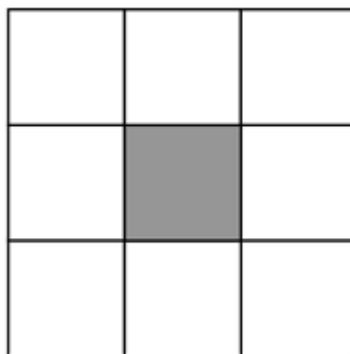
lub formułę modułową:

$$L'(m, n) = ||L_1(m, n)| + |L_2(m, n)||$$

Zaletą stosowania formuły modułowej jest mniejsza złożoność obliczeniowa, przy lekko pogorszonej jakości obrazu wynikowego.

3.4. Operacje morfologiczne

Morfologia matematyczna opiera się na dwóch nieliniowych operatorach lokalnych obliczających ekstrema: minimum (inf) oraz maksimum (sup). W przypadku binarnym odpowiednikiem operatora sup jest suma zbiorów, natomiast operatorowi inf odpowiada iloczyn zbiorów. Operatory te stosowane są dla każdego punktu obrazu wraz z jego sąsiedztwem. Kształt sąsiedztwa jest określany przy pomocy tzw. elementu strukturującego, który jest odpowiednikiem maski dla operacji filtracji. Najczęściej stosowanym elementem strukturującym jest element strukturujący jednostkowy (**Rysunek 3-22**). Kształtem swoim obejmuje on jednopunktowe najbliższe sąsiedztwo punktu. Teoretycznie ma on kształt koła jednostkowego o środku w danym punkcie, a w praktyce ściśle zależy od przyjętej siatki punktów.



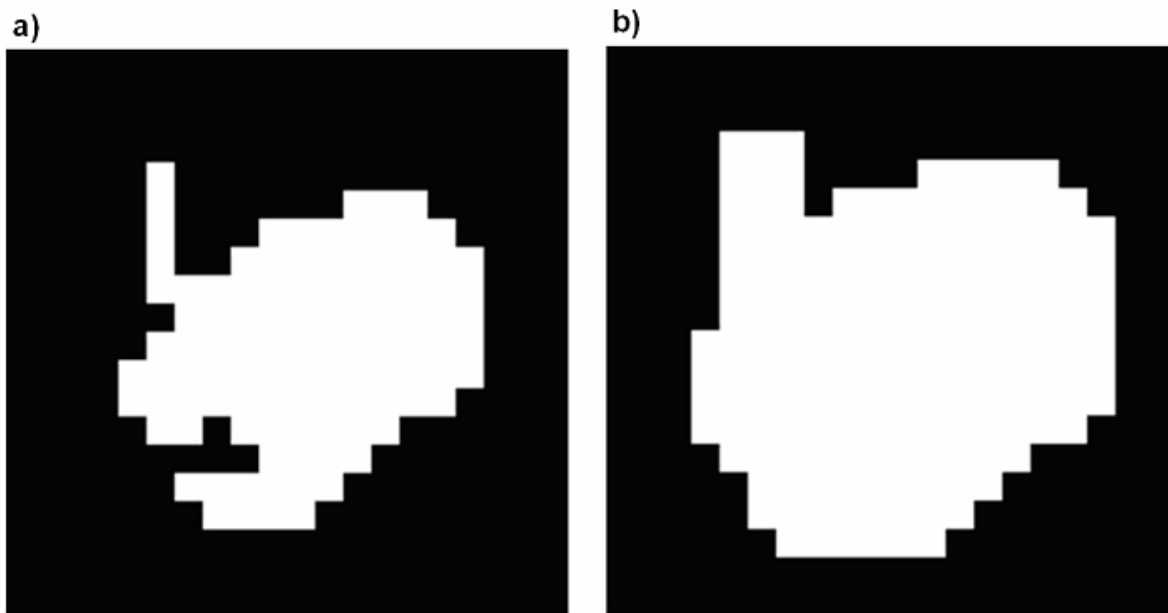
Rysunek 3-22. Element strukturujący uwzględniający 8 sąsiadów na siatce kwadratowej

Dylacja

Dylacja obrazu A przez element strukturujący B jest zdefiniowana jako suma przesunięć zbioru A o wszystkie elementy b , przy czym $b \in B$.

$$A \oplus B = \bigcup_{b \in B} A_b$$

W praktycznej implementacji dylacji element strukturujący jest przesuwany po kolejnych pikselach obrazu. Ze wszystkich pikseli jakie obejmuje sąsiedztwo, wybierany jest piksel o największej wartości i wstawiany w miejsce przetwarzanego piksela. Dla obrazów binarnych, operacja dylacji powoduje rozszerzenie jaśniejszych obszarów kosztem ciemniejszych (**Rysunek 3-23**).



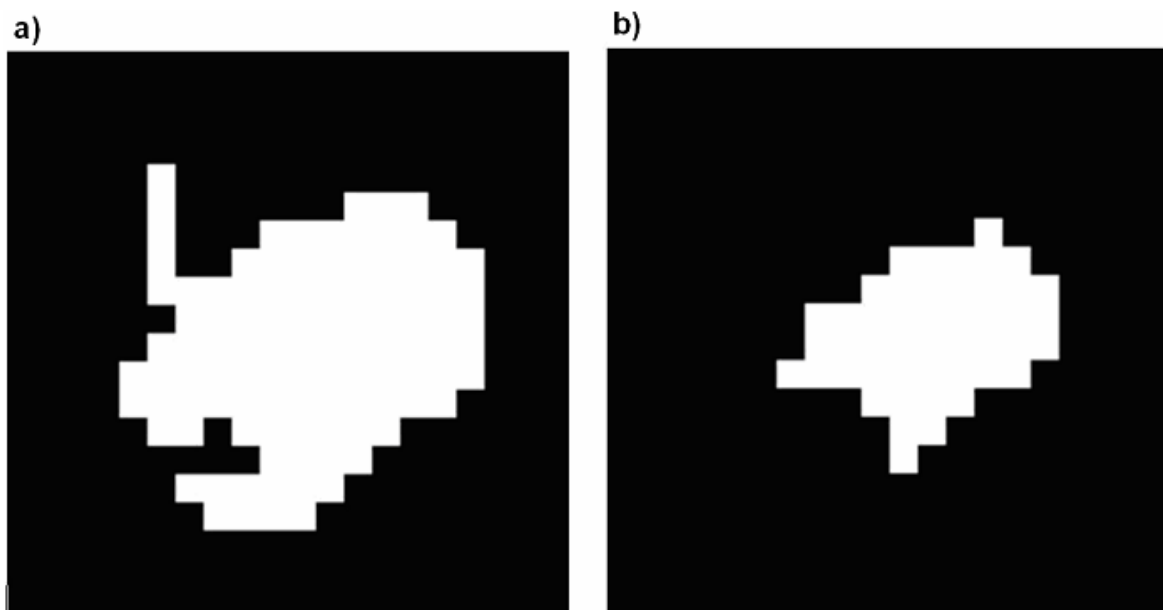
Rysunek 3-23. Przykład zastosowania operatora dylacji: a) obraz oryginalny, b) obraz po przeprowadzonej dylacji [8]

Erozja

Erozja obrazu A przez element strukturujący B jest zdefiniowana jako iloczyn przesunięć zbioru A o wszystkie elementy b , przy czym $b \in B$.

$$A \ominus B = \bigcap_{b \in B} A_{-b}$$

W praktycznej implementacji dylacji, element strukturujący jest przesuwany po kolejnych pikselach obrazu. Ze wszystkich pikseli jakie obejmuje sąsiedztwo wybierany jest piksel o najmniejszej wartości i wstawiany w miejsce przetwarzanego piksela. Dla obrazów binarnych, operacja erozji powoduje rozszerzenie ciemniejszych obszarów, kosztem jaśniejszych (**Rysunek 3-24**).



Rysunek 3-24. Przykład zastosowania operatora erozji[8]

Otwarcie i zamknięcie

Otwarcie i zamknięcie są dwoma podstawowymi filtrami morfologicznymi. Są one złożeniem erozji i dylacji definiowanymi następująco:

otwarcie:

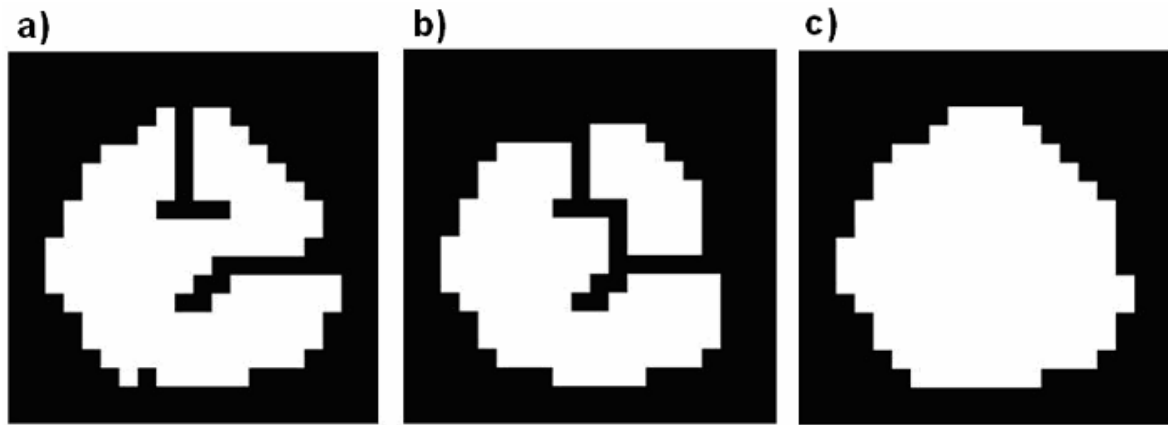
$$A \circ B = [A \ominus B] \oplus B$$

zamknięcie:

$$A \bullet B = [A \oplus B] \ominus B$$

Otwarcie jest operacją polegającą na wykonaniu erozji, a następnie po niej dylacji. Otwarcie usuwa drobne obiekty i drobne szczegóły, nie zmieniając zasadniczej wielkości figury. Może rozłączyć niektóre obiekty z przewężeniami.

Zamknięcie jest operacją polegającą na wykonaniu dylacji, a następnie po niej erozji. Zamknięcie wypełnia wąskie wcięcia i zatoki oraz drobne otwory wewnątrz obiektu, nie zmieniając wielkości zasadniczej części, może połączyć leżące blisko siebie obiekty (*Rysunek 3-25*).



Rysunek 3-25. Przebieg operacji otwarcia i zamknięcia prostego obrazu, obraz oryginalny (a), obraz po otwarciu (b), obraz po zamknięciu elementem 5x5 (c). [8]

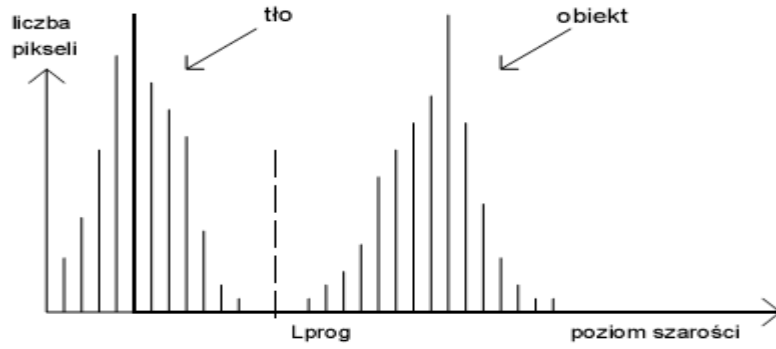
3.5. Segmentacja

Segmentacja obrazu jest techniką polegającą na podziale obrazu na logiczne części zwane obszarami. Pojęcie obszaru jest używane do określenia pewnej jednorodnej pod pewnym względem grupy punktów i mogą ją stanowić np. piksele o jednakowym kolorze, poziomie jasności, fakturze czy należące do jednego obiektu[6]. Bardzo często segmentacja polega na wyodrębnieniu pewnych obiektów od tła. Obecnie w technice przetwarzania obrazów znanych jest kilka różnych rodzajów segmentacji takich jak: segmentacja przez podział obszaru, segmentacja metodą wykrywania krawędzi czy segmentacja przez rozrost obszaru.

Segmentacja przez podział obszaru

Segmentacja przez podział obszaru polega na stopniowym podziale dużych obszarów na mniejsze zwykle przy zastosowaniu kilku iteracji. Idea metody polega na porównywaniu wartości jasności każdego elementu z pewną wartością progową T . Następnie, każdy element obrazu (piksel- $f(x,y)$) jest przydzielany do jednej z dwóch kategorii: o wartości większej bądź równej wartości progowej T $\{f(x,y) \geq T\}$ lub wartości mniejszej od wartości progowej $\{f(x,y) < T\}$. Po operacji progowania otrzymuje się tak zwany obraz binarny, składający się z samych zer i jedynek ($\{f(x,y) \geq T\} = 0$ i $\{f(x,y) < T\} = 1$). Kwestią dyskusyjną w tej metodzie jest dobór odpowiedniej wartości progu T .

Najpopularniejszą metodą wyboru wartości progowej jest analiza histogramu obrazu, która bazuje na założeniu, iż szukany obiekt ma wyraźnie inną jasność od tła. W takim przypadku na histogramie powinny pojawić się dwa wyraźne wierzchołki (**Rysunek 3-26**), spośród których jeden odpowiada za obiekt, a drugi za tło. Próg następnie jest dobierany jako wartość pośrednia pomiędzy tymi obszarami.



Rysunek 3-26. Dobór progu za pomocą histogramu[6]

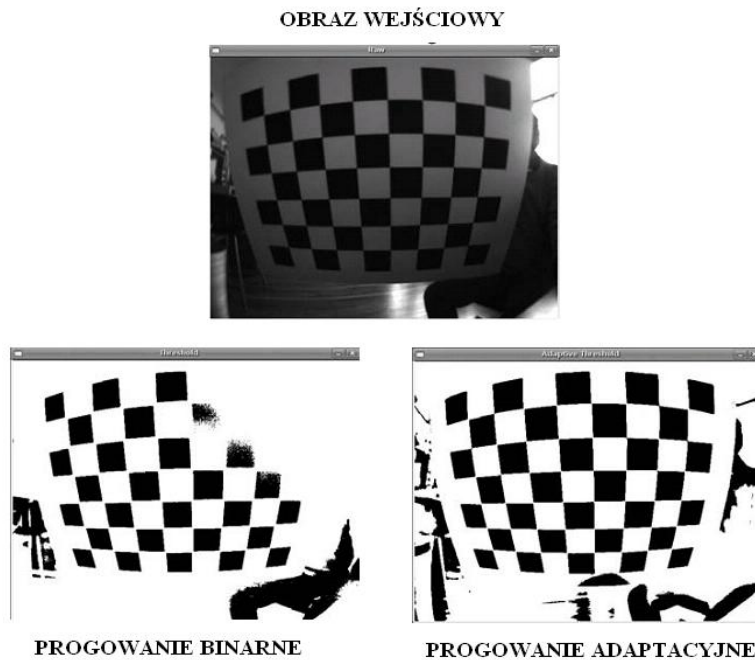
Istotną wadą tej metody jest to, iż obraz histogramu bywa wielomodalny, a wierzchołki które należy odseparować są trudne do wyróżnienia. Wobec tego metoda doboru progu za pomocą histogramu jest ograniczona do scen o dość jednorodnych powierzchniach występujących na kontrastowym tle[6].

Innym sposobem doboru progu jest tzw. progowanie adaptacyjne umożliwiające różny dobór progu dla każdego piksela na obrazie, w zależności od sąsiedztwa, w jakim piksel się znajduje. Parametrami wejściowymi funkcji jest minimalna wariancja $v_{\min}$ oraz rozmiar sąsiedztwa $p \times p$. Dla każdego piksela i, j o sąsiedztwie $p \times p$, -liczona jest wartość średnia m_{ij} oraz wariancja v_{ij} :

$$m_{i,j} = \frac{1}{2p} * \sum_{s=-p}^p \sum_{t=-p}^p f_{i+s,j+t}$$

$$v_{i,j} = \frac{1}{2p} \sum_{s=-p}^p \sum_{t=-p}^p |f_{i+s,j+t} - m_{i,j}|$$

Lokalny próg dla piksela i, j liczony jest według wzoru $t_{ij} = m_{ij} + v_{ij}$, gdy $v_{ij} > v_{\min}$. Jeśli $v_{ij} < v_{\min}$, wtedy wartość progu t_{ij} jest wartością progu jaką przyjmował poprzedni przetwarzany piksel t_{ij-1} . Praktycznym problemem dającym się rozwiązać przy pomocy progowania adaptacyjnego jest sytuacja, w której szukany obiekt jest pod wpływem intensywnego refleksu światła, co czyni jego barwę nieregularną z punktu widzenia kamery. W takim przypadku dobór pojedynczego progu otrzymanego za pomocą histogramu może okazać się niewystarczający, co jest widoczne na rysunku (**Rysunek 3-27**).



Rysunek 3-27. Porównanie działania progowania binarnego z progowaniem adaptacyjnym, w przypadku obiektu będącego pod intensywnym refleksem światła[7]

Segmentacja metodą wykrywania krawędzi

Segmentacja metodą wykrywania krawędzi polega na wykorzystaniu filtru gradientowego górnoprzepustowego. W wyniku jego działania na obrazie pozostają jedynie kontury przedmiotów. Gradient obrazu jest następnie poddawany operacji progowania, a elementy, które podczas tych operacji nie zostały zidentyfikowane jako krawędzie, są łączone dla uformowania krzywych zamkniętych. Metody wykrywania krawędzi mają nieco inne zastosowanie, niż metody wykorzystujące progowanie i praktycznie nadają się jedynie do obrazów z dużym kontrastem i niskim poziomem zakłóceń o wysokich częstotliwościach. W przeciwieństwie do progowania binarnego, dobrze sprawdzają się w przypadku obrazów z zakłóceniami o małych częstotliwościach i z powodzeniem można je stosować do wykrywania obiektów z intensywnym refleksem światła.

Segmentacja przez rozrost obszaru

Metody segmentacji przez rozrost obszaru, w przeciwieństwie do metod przez podział obszaru, nie zajmują się różnicami między obszarami, a wychwytywaniem ich podobieństw. Najprostsza wersja takiego algorytmu polega na analizie sąsiedztwa poszczególnych pikseli. Jeśli piksele sąsiednie mają zbliżoną jasność, wtedy są łączone w jeden obszar. W ten sposób po kilku iteracjach takiego algorytmu tworzone są coraz większe obszary punktów odpowiadających podobnym poziomom jasności. Sam proces segmentacji trwa tak długo aż wszystkie obszary są jednolite. Problematiczną kwestią w tym algorytmie jest dobór pikseli, od których algorytm zaczyna działanie. Mogą być one wybierane w sposób losowy w najprostszych wersjach algorytmu, a w przypadku bardziej złożonych implementacji, możliwy jest początkowy podział obrazu na kilka obszarów i testowanie ich jednolitości. Główną zaletą metod segmentacji przez rozrost obszaru jest mała wrażliwość na zakłócenia,

wadą natomiast jest to, że słabo sprawdzają się w przypadku skomplikowanych scen mając tendencję do tworzenia zbyt dużej liczby małych obszarów.

3.6. Momenty geometryczne bezwładności

Metody momentowe służą do opisu właściwości figur. Wynika to z faktu, że figury geometryczne o różnych kształtach wykazują podobne cechy, które mogą być określone przez momenty. Przykładowo za pomocą momentów może być odróżniony kształt kwadratu od trójkąta czy koła. Momenty bezwładności pierwszego rzędu określają położenie środka ciężkości figury, natomiast momenty drugiego rzędu są miarą bezwładności figury. Wyznaczenie poszczególnych momentów bezwładności sprowadza się do prostego sumowania po wszystkich punktach figury $f(x,y)$. Dla momentów pierwszego rzędu $p=1, q=0$ lub $p=0, q=1$. Przypadku momentów drugiego rzędu $p=2, q=0$ lub $p=0, q=2$.

$$m_{p,q} = \sum_x \sum_y x^p y^q f(x,y)$$

Współrzędne środków ciężkości można obliczyć ze wzorów:

$$\bar{x} = \frac{m_{10}}{m_{00}}, \bar{y} = \frac{m_{01}}{m_{00}},$$

Definiuje się również momenty centralne, niezależne od współrzędnych obiektu:

$$\mu_{p,q} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q f(x,y)$$

Momenty centralne można znormalizować, dzieląc je przez odpowiednią potęgę μ . Otrzymuje się niezmiennie względem przesunięcia i skalowalne momenty centralne znormalizowane wyrażające się wzorem:

$$\eta_{p,q} = \frac{\mu_{p,q}}{\mu_{00}^{\frac{p+q}{2}}}, \mu = \frac{p+q}{2} + 1$$

Za pomocą momentów znormalizowanych można wyznaczyć niezmienniki momentowe:

$$h_1 = \eta_{20} + \eta_{02}$$

$$h_2 = (\eta_{20} + \eta_{02})^2 + 4\eta_{11}^2$$

$$h_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2$$

$$h_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2$$

$$h_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{12} + \eta_{03})^2] \\ + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]$$

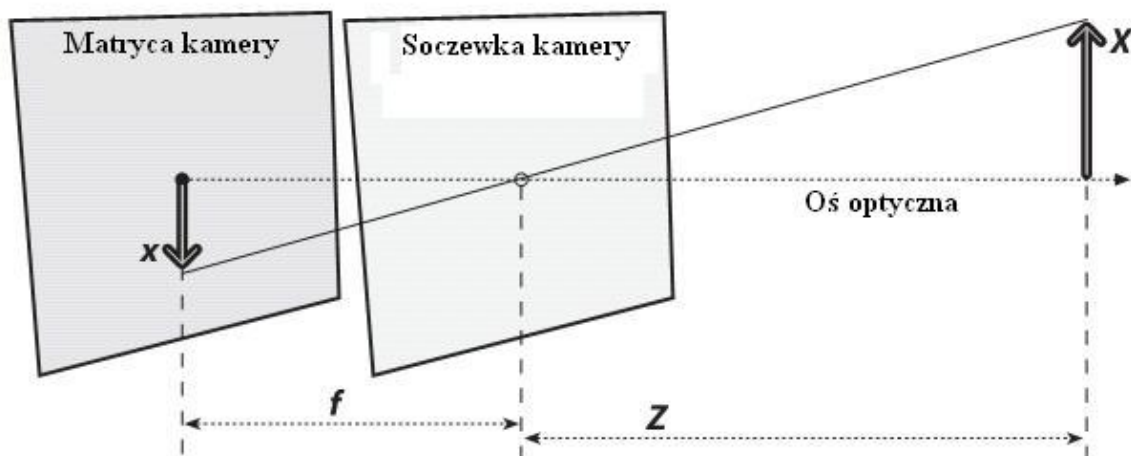
Niezmienniki momentowe mają olbrzymie znaczenie w dzisiejszej technice rozpoznawania obrazów. Można je wyznaczyć dla każdej figury geometrycznej. Umożliwiają one rozpoznawanie obiektów o różnych kształtach, pod różnymi kątami widzenia kamery. Jest to szczególnie ważna właściwość w przypadku rozważanego problemu lokalizacji robota mobilnego, gdzie rozpoznawany znacznik może znajdować w najrozmaitszych pozycjach wobec kamery. Filtracja za pomocą niezmienników momentowych jest w stanie bezbłędnie zlokalizować znacznik na obrazie. Niezmienniki momentowe ponadto powszechnie używane są do rozpoznawania liter i cyfr, więc znajdują zastosowanie w oprogramowaniu przekształcającym ręcznie pisany tekst do postaci cyfrowej[8].

3.7. Kamera cyfrowa

Zasada działania kamery cyfrowej jest analogiczna do zasady działania ludzkiego oka i polega na rejestracji natężenia padającego światła. Światło wypromieniowywane z zewnętrznego źródła (żarówka, Słońce) padając na obiekt będący w polu widzenia kamery częściowo jest przez niego absorbowane i częściowo ulega odbiciu. Wskutek wychwytywania przez kamerę odbitego promieniowania od innych obiektów tworzony jest cyfrowy obraz.

Podstawy konstrukcji i działania kamery cyfrowej

Najprościej zasadę działania kamery ilustruje rysunek (**Rysunek 3-28**), gdzie f jest ogniskową kamery, Z jest odległością kamery od obiektu, X jest rzeczywistą wielkością obiektu, a x jest wielkością uchwyconego obiektu na matrycy.



Rysunek 3-28. Zasada działania kamery cyfrowej[7]

Korzystając z metody podobieństwa trójkątów, można wyprowadzić równanie na wielkość obiektu tworzonego na matrycy:

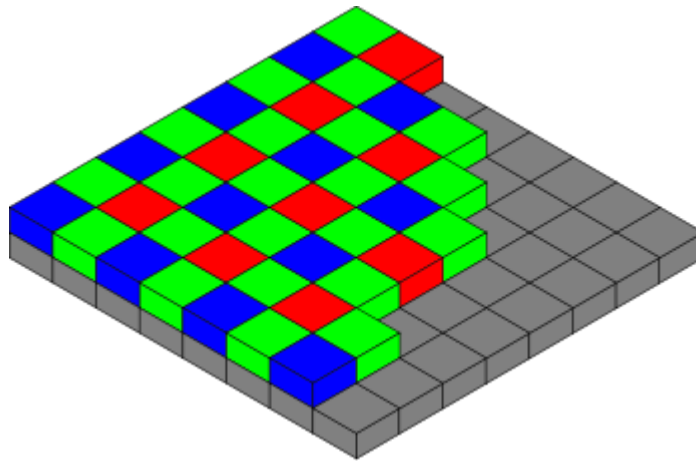
$$-x = f \frac{X}{Z}$$

Podstawowymi elementami, z których zbudowana jest każda kamera cyfrowa są: soczewka skupiająca, światłoczuła matryca oraz filtr mozaikowy. Konieczność stosowania soczewki wynika z faktu, że otwór wychwytyjący promieniowanie świetlne w kamerze jest

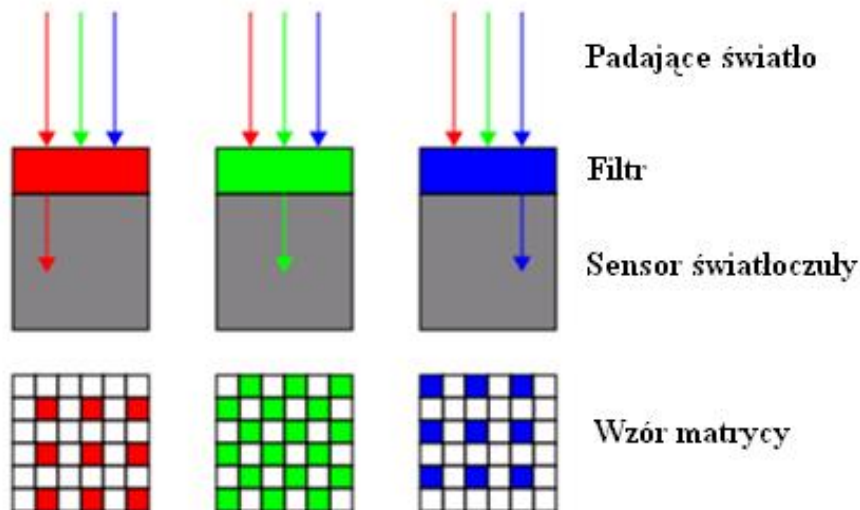
relatywnie mały. Soczewka jest, więc niezbędna dla uzyskania wysokiej czułości. Matryca kamery zbudowana jest z pewnej ilości światłoczułych elementów umieszczonych w pionie i w poziomie, analogicznie do pikseli na obrazie cyfrowym. Promień światła padający na określony element matrycy powoduje powstanie nim ładunku elektrycznego, proporcjonalnego do natężenia światła. Wartość powstałego ładunku jest następnie przetwarzana do postaci cyfrowej, poprzez przetworniki analogowo-cyfrowe. Obecnie w technice cyfrowego przetwarzania obrazów upowszechnione są dwie podstawowe technologie światłoczułych elementów matrycy:

- CCD – matryca jest zbudowana z światłoczułych kondensatorów
- CMOS – matryca jest zbudowana z fotodiod i tranzystorów

Ponieważ światłoczuła matryca reaguje na cały zakres długości fal promieniowania świetlnego, konieczne staje się użycie filtra mozaikowego w celu rozróżniania barwy. Najpopularniejszym stosowanym filtrem mozaikowym jest filtr Bayera. Istota jego działania polega na pokryciu matrycy światłoczułych elementów, mozaiką złożoną z elementów filtrujących selektywnie zakresy promieniowania świetlnego (*Rysunek 3-29, Rysunek 3-30*). Odpowiada to barwom w przestrzeni RGB: czerwonej, zielonej, niebieskiej. Na pojedynczy piksel na cyfrowym obrazie wynikowym, przypadają zatem cztery światłoczułe elementy położone w swoim sąsiedztwie, z których dwa pokryte są filtrem zielonym, a po jednym niebieskim i czerwonym. Przewaga zielonych elementów w filtrze mozaikowym wynika z właściwości oka ludzkiego, które jest najbardziej wrażliwe na barwę zieloną. Wartość dla dwóch elementów mozaiki zielonej przypadających na jeden piksel wynikowy jest następnie uśredniana i wszystkie trzy składowe przetwarzane są do postaci cyfrowej za pomocą przetworników analogowo-cyfrowych.



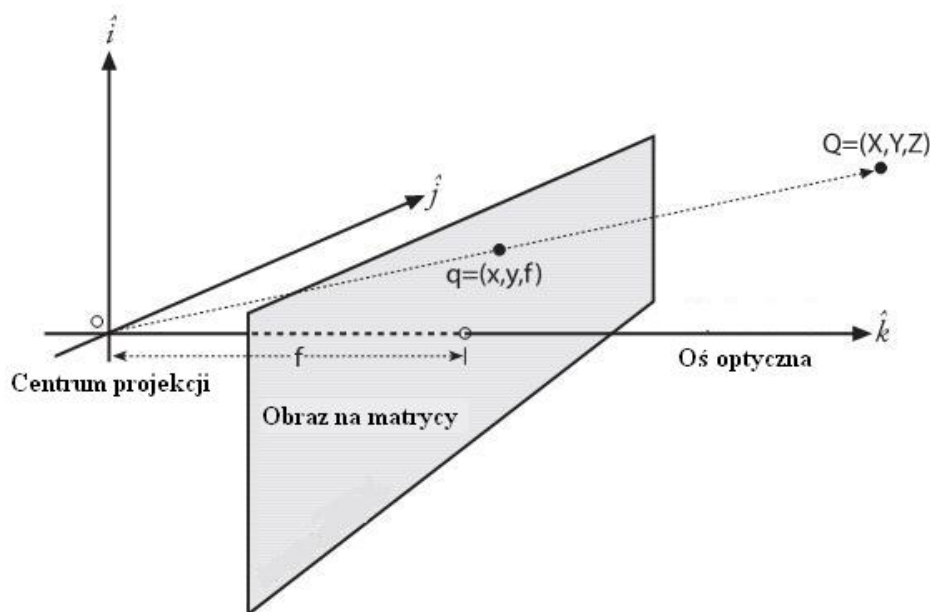
Rysunek 3-29. Światłoczuła matryca pokryta filtrem Bayera[18]



Rysunek 3-30. Filtracja barw RGB przez filtr Bayera[18]

Model i zniekształcenia obrazu kamery cyfrowej wykorzystywane przez środowisko OpenCV

Opisując zniekształcenia kamery, środowisko OpenCV posługuje się modelem kamery przedstawionym na rysunku (**Rysunek 3-31**). Nie jest to rzeczywisty model, ponieważ cyfrowy obraz na matrycy jest rozważany z prawej strony punktu projekcji (odwrotnie niż w rzeczywistości). Taki zabieg ma na celu uproszczenie obliczeń matematycznych.



Rysunek 3-31. Punkt $Q(X,Y,Z)$ będący rzeczywistym obiektem jest rzutowany na obraz matrycy, poprzez centrum projekcji. Punktem wynikowym jest punkt $q(x,y,f)$ znajdujący się na cyfrowej matrycy kamery[7]

W praktyce obliczenie położenia punktu q znajdującego się na matrycy kamery na jest obarczone pewnym błędem. Błąd ten spowodowany jest głównie niedokładnościami w procesie produkcji kamery i zwykle uwidacznia się w najtańszych kamerach cyfrowych o słabej jakości wykonania. Jako przyczyny błędu lokalizacji punktu na obrazie matrycy można

wyróżnić niedokładność konstrukcji matrycy, niedokładność konstrukcji soczewki jak i brak precyzji w ich umiejscowieniu wewnątrz kamery.

Przykładem błędów konstrukcji oraz umiejscowienia matrycy jest odchylenie środka matrycy od osi optycznej. Może on występować zarówno w kierunkach: pionowym jak i poziomym i jest określony przez dwa współczynniki c_x i c_y . Ponadto z uwagi na fakt, iż piksele na matrycy zwykle nie mają idealnie kwadratowego kształtu wyróżnia się dwie wartości ogniskowych f_x i f_y . Uwzględniając współczynniki odchylenia oraz wartości ogniskowych, można wyprowadzić dwa wzory na położenie obrazu na matrycy:

$$x = f_x \left(\frac{X}{Z} \right) + c_x, \quad y = f_y \left(\frac{Y}{Z} \right) + c_y,$$

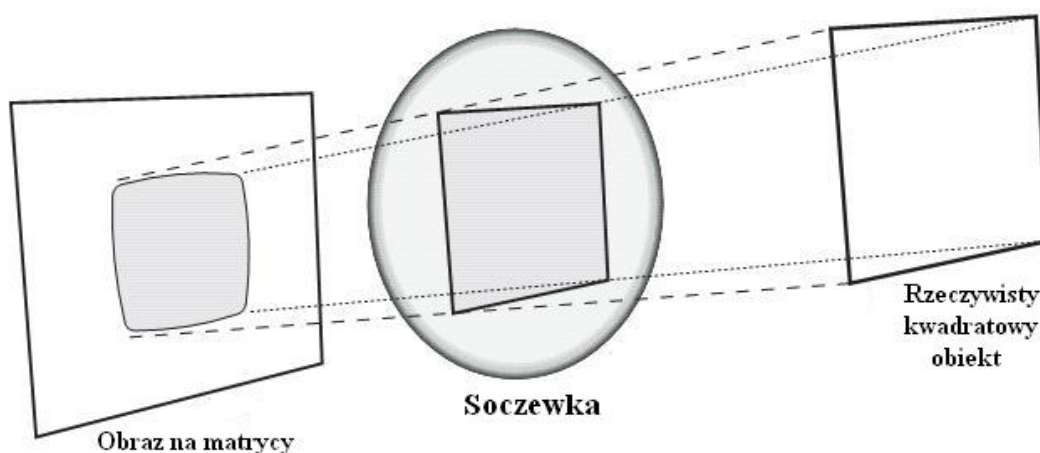
co pozwala na wyprowadzenie równania pomiędzy rzeczywistym punktem $Q(X, Y, Z)$, a odpowiadającym mu punktem obrazu na matrycy $q(x,y,w)$:

$$q = MQ, \quad \text{gdzie: } q = \begin{bmatrix} x \\ y \\ w \end{bmatrix}, \quad M = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}, \quad Q = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

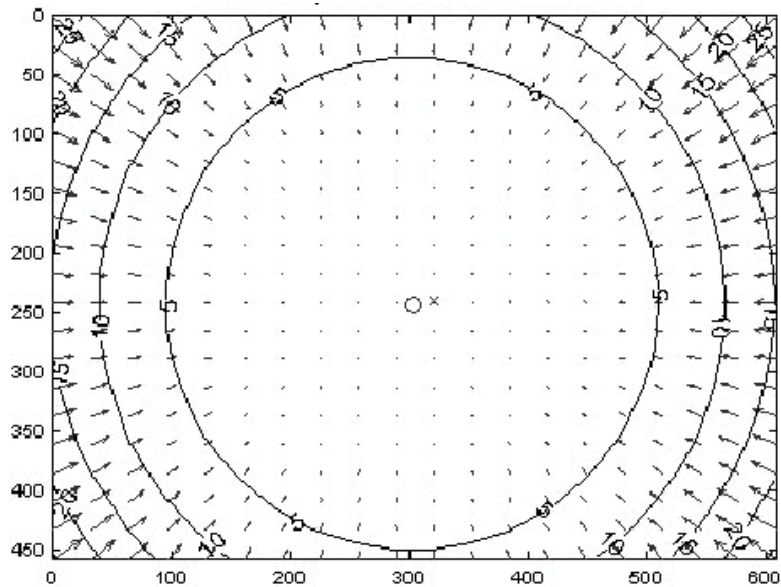
Błędy w pozycji obrazu na matrycy wynikające z konstrukcji soczewki, nazywane są zniekształceniami radialnymi. Spowodowane są one faktem, iż promienie padające na obszar brzegowy soczewki są bardziej zakrzywiane, niż promienie świetlne padające w jej środku (**Rysunek 3-32, Rysunek 3-33**). Z tego powodu w praktyce zniekształcenia radialne są minimalne dla punktów znajdujących się w centrum matrycy i rosną im dalej od niego. Matematycznym sposobem na skorygowanie zniekształceń radialnych jest ich rozwinięcie w szereg Taylora. Dla większości kamer do korekcji błędu wystarczają dwa parametry k_1 i k_2 , a tylko w sporadycznych przypadkach konieczne jest zastosowanie trzeciego współczynnika k_3 . Parametr r określa odległość analizowanego piksela od środka matrycy.

$$x_{skorygowane} = x(1 + k_1 * r^2 + k_2 * r^4 + k_3 * r^6)$$

$$y_{skorygowane} = y(1 + k_1 * r^2 + k_2 * r^4 + k_3 * r^6)$$

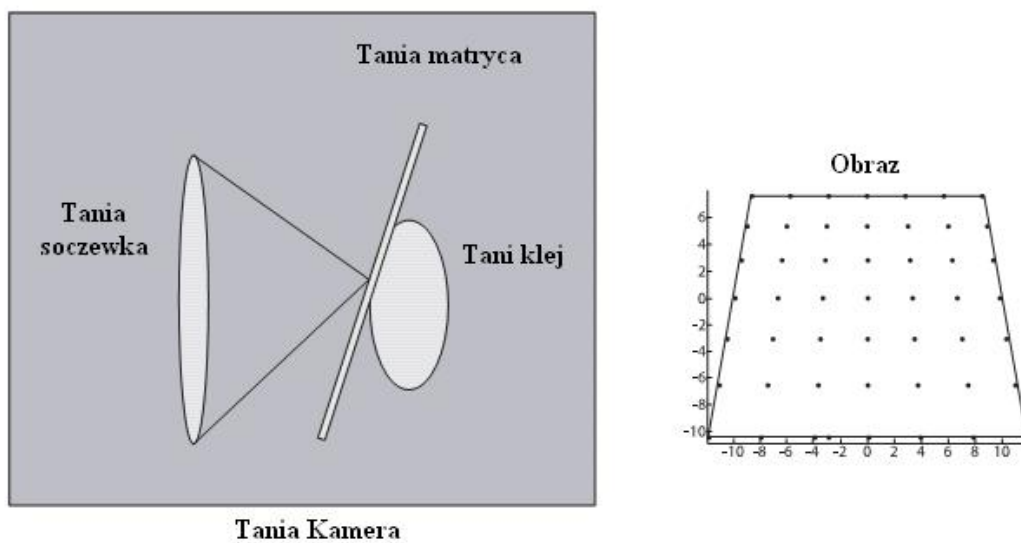


Rysunek 3-32. Przykład powstawania zniekształceń radialnych [7]

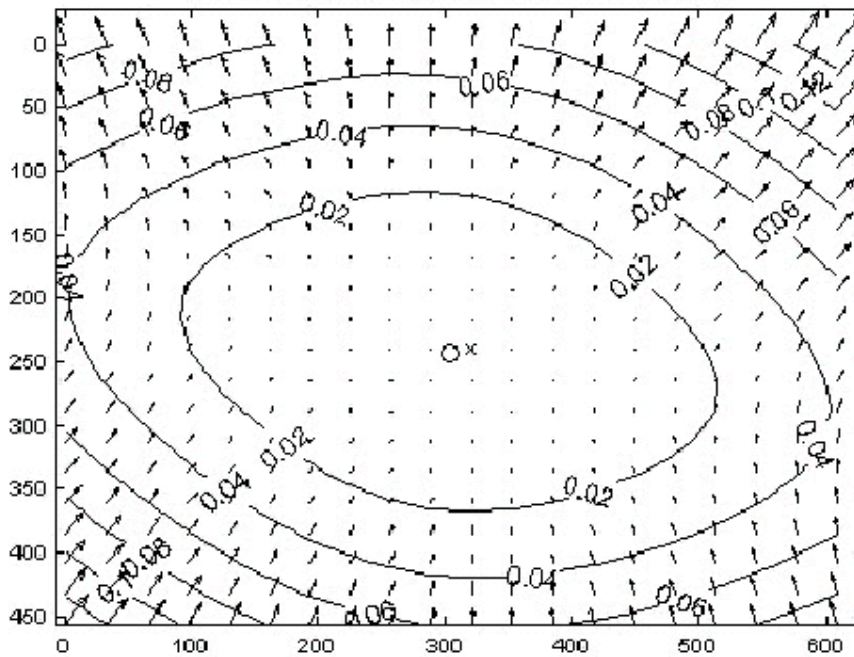


Rysunek 3-33. Rzeczywisty obraz zniekształceń radialnych, poszczególne koła obrazują obszary o jednakowym zniekształceniu radialnym [7]

Kolejnym typem zniekształceń mających wpływ na lokalizację obrazu na matrycy kamery są zniekształcenia tangencjalne. Są one spowodowane niedokładnością w umiejscowieniu matrycy względem soczewki, kiedy płaszczyzna soczewki nie jest idealnie równoległa do płaszczyzny matrycy (**Rysunek 3-34**). Rozkład zniekształceń tangencjalnych na matrycy przedstawia rysunek (**Rysunek 3-35**).



Rysunek 3-34. Przyczyny powstawania zniekształceń tangencjalnych[7]



Rysunek 3-35. Rzeczywisty model zniekształceń tangencjalnych

Zniekształcenia tangencjalne mogą zostać skorygowane używając równań, w których zostają wprowadzone dwa nowe parametry p_1 i p_2 , a r oznacza odległość od środka matrycy.

$$x_{\text{skorygowane}} = x + [2p_1y + p_2(r^2 + 2x^2)]$$

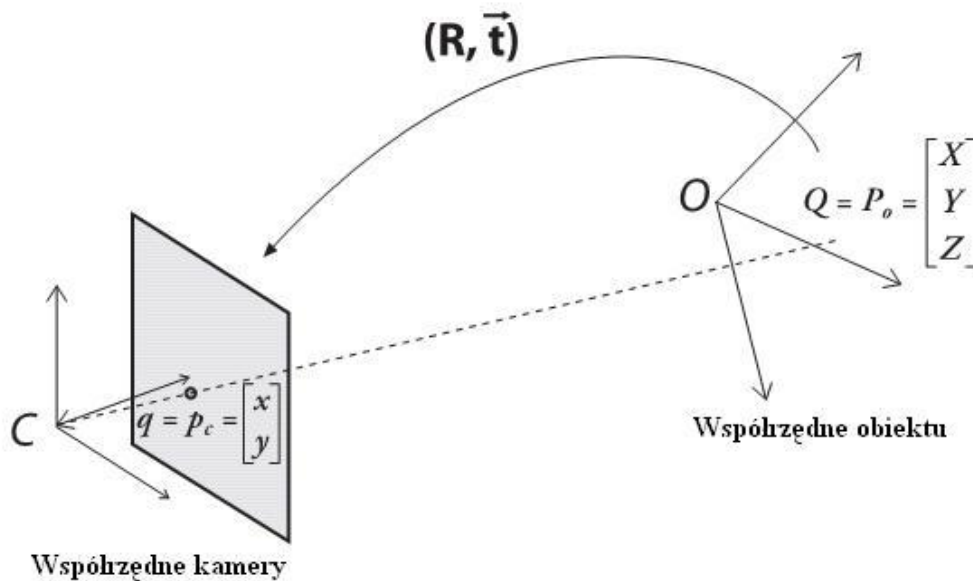
$$y_{\text{skorygowane}} = y + [p_1(r^2 + 2y^2) + 2p_2x]$$

Podsumowując, OpenCV używa siedmiu współczynników c_x , c_y , k_1 , k_2 , k_3 , p_1 , p_2 , które obrazują odchylenie środka matrycy od osi optycznej oraz zniekształcenia radialne i tangencjalne. Ponadto błędy w konstrukcji pikseli uwzględniane są przez dwie wartości ogniskowych f_x i f_y . Należy dodać, iż istnieje wiele innych rzeczywistych zniekształceń, ale ich efekt jest relatywnie niewielki, a uwzględnione typy błędów są najbardziej znaczące.

Kalibracja kamery

Kalibracja kamery jest empiryczną procedurą mającą na celu wyznaczenie parametrów kamery czyli: ogniskowej, środka matrycy kamery oraz parametrów zniekształceń tangencjalnych i radialnych. Wielkości te są niezbędne dla określenia dokładnej pozycji dowolnego obiektu względem kamery.

W rozpatrywanym problemie kalibracji kamery istnieją dwa trójwymiarowe układy, z których jeden obrazuje względne położenie kamery względem obiektu, a drugi położenie obiektu względem kamery. Dla dowolnego obiektu uchwyconego w kamerze możemy wyznaczyć jego przekształcenie z jednego układu współrzędnych do drugiego, posługując się pojęciami: rotacji oraz translacji.



Rysunek 3-36. Przekształcenie współrzędnych obiektu (Q) do współrzędnych obiektu na matrycy kamery (q) z pomocą wektorów rotacji (R) i translacji (t) [7]

Zadaniem translacji jest określenie położenia jednego punktu, względem drugiego w układzie współrzędnych, natomiast rotacja odpowiada za względne położenie kątowe tzw. orientację obydwu obiektów. Powszechnie rotacja i translacja są przedstawiane w postaci wektorowej, a przekształcenie z jednego układu współrzędnych odbywa się poprzez równanie:

$$P_c = R(P_o - T)$$

Gdzie:

P_c to położenie punktu obrazu na matrycy

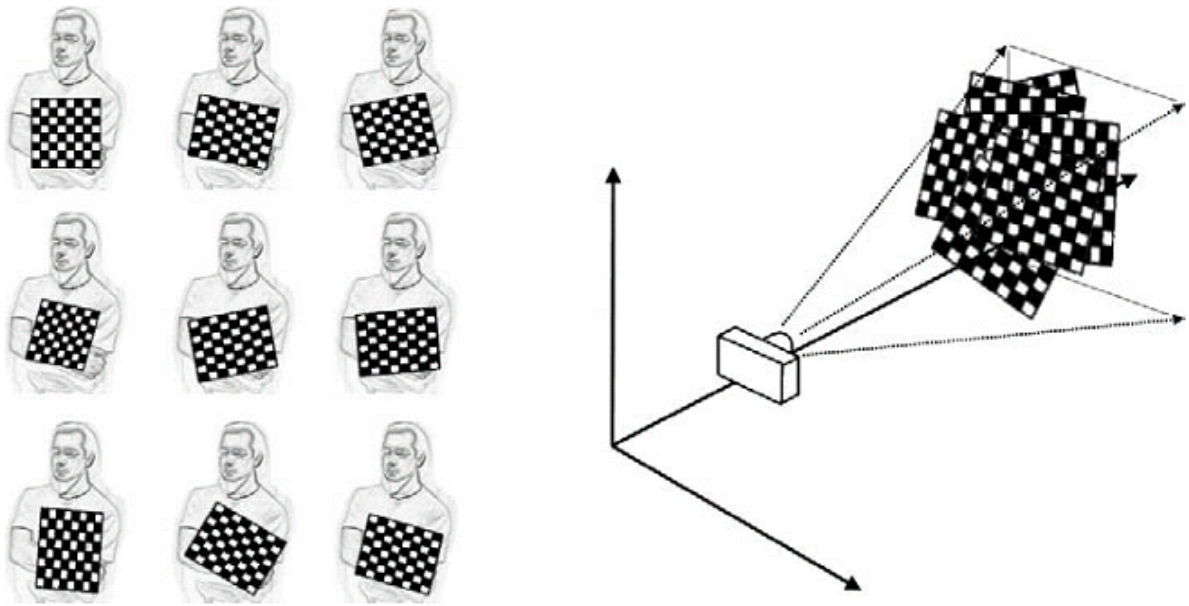
P_o to położenie punktu w rzeczywistym środowisku będącym w polu widzenia kamery

T to wektor translacji zawierający współrzędne punktu w przestrzeni trójwymiarowej

R to macierz rotacji zawierający położenie kątowe względem wszystkich trzech osi

W przestrzeni trójwymiarowej każdy punkt opisany jest przez 3 współrzędne (X, Y, Z) oraz 3 położenia kątowe względem każdej z osi ($\varphi_x, \varphi_y, \varphi_z$). Razem z czterema parametrami wewnętrznymi kamery f_x, f_y, c_x, c_y daje to łącznie 10 różnych parametrów, które algorytm musi estymować dla każdego widoku.

Proces kalibracji kamery można przeprowadzić używając dowolnego trójwymiarowego obiektu o znanych wymiarach geometrycznych, jednak najbardziej powszechnym (dostępnym w wielu różnych programach do kalibracji kamery) jest płaski wzór szachownicy. Szachownica jest fotografowana w różnych pozycjach względem kamery i na podstawie tych obrazów przeprowadzany jest dalszy proces kalibracji.



Rysunek 3-37. Kalibracja kamery [7]

Do algorytmu kalibracji używane są narożniki pól szachownicy, czyli elementy znajdujące się na obszarze granicznym czterech kwadratów: dwóch białych i dwóch czarnych.

Aby obliczyć położenie obiektu względem kamery, konieczne staje się użycie szeregu równań i zestawienie z nich układu równań wielu niewiadomych. Parametry ogniskowych oraz środka matrycy są stałe, dla wszystkich obrazów nieznane i również wyznaczenie ich wartości jest zadaniem kalibracji kamery. Poprawna kalibracja kamery wymaga spełnienia warunku:

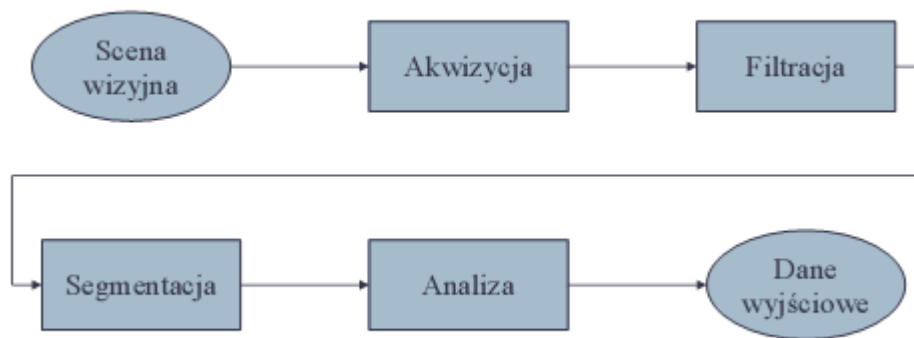
$$2NK \geq 6K$$

gdzie N to liczba narożników, a K to liczba zdjęć, gdzie minimalną liczbą zdjęć jest liczba dwa.

Większa liczba zdjęć i narożników, sprzyja większej dokładności estymacji szukanych parametrów. W praktyce do procesu kalibracji kamery została użyta szachownica złożona z 48 narożników, uchwycona na 20tu zdjęciach pod różnymi kątami i w różnej odległości od kalibrowanej kamery. Do przeprowadzenia kalibracji użyte zostało oprogramowanie Camera Calibration Toolbox dla środowiska Matlab [13].

4. Opis algorytmu przetwarzania obrazu użytego do lokalizacji robota mobilnego

Proces analizy obrazu w celu lokalizacji robotów przebiega według schematu typowego dla większości procesów obróbki obrazu (**Rysunek 4-1**).

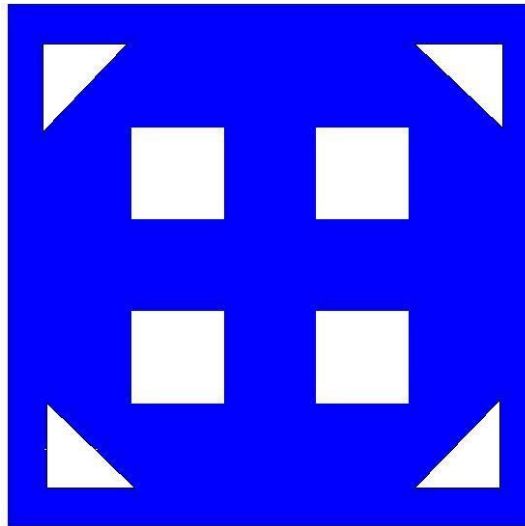


Rysunek 4-1. Ogólny schemat działania algorytmu wizyjnego[15]

Obszar znajdujący się w polu widzenia kamery (scena wizyjna) poddawany jest akwizycji obrazu i przekształcany do postaci cyfrowej. Następnie wykonywany jest proces wstępnej filtracji (kondycjonowania) mający na celu poprawienie jakości, by umożliwić lub ułatwić dalsze działanie algorytmu. Segmentacja wyodrębnia szukane obiekty od tła na obrazie. W przypadku omawianego systemu polega to na identyfikacji dokładnego miejsca, w którym znajduje się znacznik na obrazie dwuwymiarowym. Podczas procesu analizy, pozycja znalezionego znacznika zostaje wyznaczona w przestrzeni trójwymiarowej, co stanowi dane wyjściowe algorytmu.

4.1. Budowa znacznika

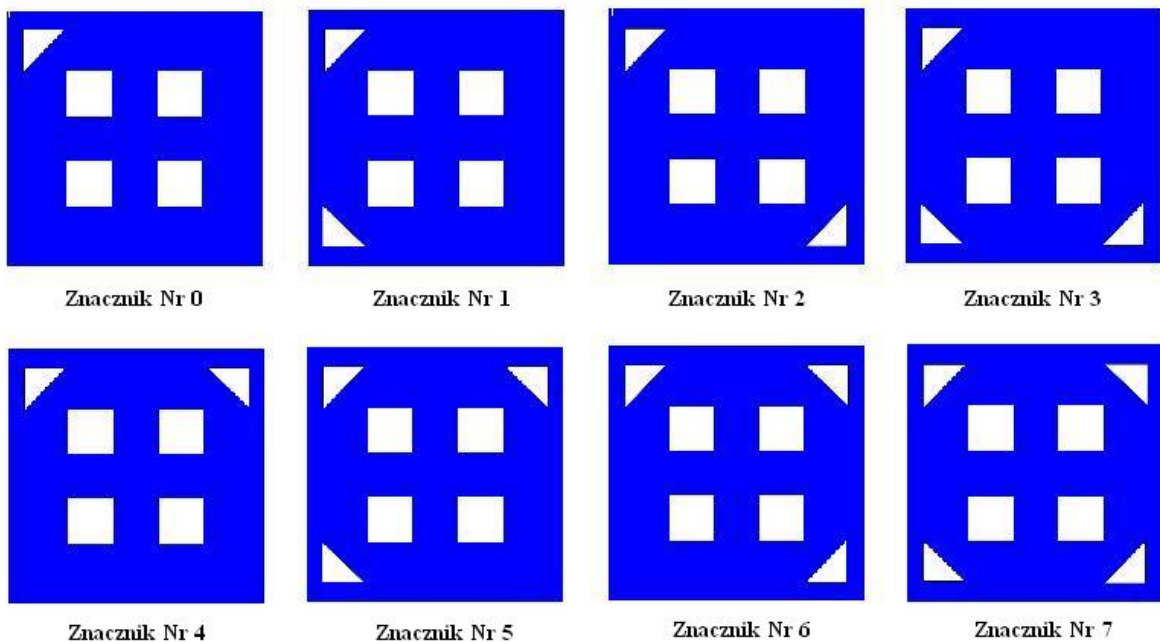
Znacznik użyty do lokalizacji, stworzony na potrzeby działania algorytmu jest przedstawiony na rysunku (**Rysunek 4-2**). Ma on kształt kwadratu o długości boku równej siedemnaście centymetrów. W jego środkowej części umieszczone są białe kwadratowe pola o długości boku równej trzy centymetry, które umożliwiają określenie pozycji względem kamery. Aby możliwe stało się zlokalizowanie robota w wielu miejscach w pomieszczeniu należało stworzyć zestaw odróżnialnych znaczników. Do tego celu służą białe równoramienne trójkąty widoczne w narożnych obszarach.



Rysunek 4-2. Znacznik użyty do lokalizacji

Rysunek (**Rysunek 4-3**) przedstawia cały zestaw ośmiu stworzonych na potrzeby działania systemu wizyjnego znaczników, różnią się one liczbą narożnych trójkątów. Licząc od lewego dolnego trójkąta ruchem przeciwnym do wskazówek zegara każdemu poszczególnemu trójkątowi przyporządkowane są kolejno wartości: 1,2,4, a numer znacznika jest określony poprzez ich sumowanie. Jeśli dany trójkąt występuje na znaczniku to jego umowna wartość jest uwzględniana przy wyliczaniu numeru. Jeśli nie występuje jego umowna wartość jest równa zero. Przykładowo dla znacznika numer 5 wartość wyliczana jest następująco:

$$\text{Numer znacznika } 5 = 1 * 1 + 2 * 0 + 4 * 1 = 5$$



Rysunek 4-3. Zestaw znaczników

Celem umieszczenia czwartego trójkąta w lewym górnym rogu jest weryfikacja poprawności działania identyfikacji i nie ma on przypisanej żadnej wartości binarnej. Jeśli ów trójkąt zostaje rozpoznany na obrazie wtedy algorytm przyjmuje, że numer przyporządkowany znacznikowi jest określony prawidłowo, w przeciwnym wypadku identyfikacja znacznika jest uznawana za niepoprawną.

4.2. Akwizycja obrazu

Jako urządzenie wykorzystane do akwizycji obrazu zastosowano kamerę internetową Creative Live Cam Optia AF (*Rysunek 4-4*) wyposażoną w matrycę CMOS. Niewątpliwa zaleta tego urządzenia to bardzo wysoka jak na kamery internetowe, rozdzielczość matrycy dwóch mega pikseli, dzięki czemu możliwe staje się uzyskiwanie obrazu o rozdzielczości 1600x1200 pikseli. Jest to kamera z rodziny kamer szerokokątnych (pole widzenia 63 stopnie). W przypadku takiego rodzaju kamery bardzo istotną rolę odgrywa algorytm segmentacji, ponieważ oprócz szukanego znacznika na obrazie pojawia się duża liczba innych, niepotrzebnych z punktu widzenia działania algorytmu, przedmiotów. Zasilanie kamery odbywa się poprzez port USB, co jest rozwiązaniem wygodnym i nie wymaga dodatkowej karty akwizycji obrazu. Kamera posiada również funkcję automatycznego wyostrażania obrazu.



Rysunek 4-4. Kamera Creative Live Cam Optia AF [20]

4.3. Kondycjonowanie obrazu

Kondycjonowanie obrazu odbywa się głównie poprzez interfejs kamery, za pomocą którego regulowane są wartości ekspozycji, kontrastu, jasności, koloru oraz korekcji gamma. Wartość tych ustawień regulowana była manualnie w zależności od otoczenia, w jakim pracuje kamera. Manualne dobieranie ustawień jest rozwiązaniem dobrym w przypadku miejsc o regularnym sztucznym oświetleniu. W przypadku pomieszczenia laboratorium, w którym przeprowadzone zostały testy wartość ustawień trzeba było zmieniać w zależności od pory dnia i aktualnego nasłonecznienia. Automatyzacja procesu doboru parametrów kamery

mogłaby przyczynić się do znacznego usprawnienia działania całego systemu. Dodatkowo w celu usuwania przypadkowego szumu na obrazie została zastosowana filtracja medianowa.

4.4. Segmentacja sceny wizyjnej

Proces segmentacji znacznika w rozpatrywanym algorytmie jest procesem dwustopniowym. Pierwszy etap segmentacji ma na celu wyodrębnienie znacznika od pozostałych elementów tworzących scenę wizyjną. Drugi etap segmentacji polega na wyodrębnieniu poszczególnych elementów znacznika.

Lokalizacja znacznika na obrazie rozpoczyna się od wykrycia wszystkich obszarów odpowiadających niebieskiej barwie. Proces ten korzysta z przestrzeni HSV i analizuje każdy piksel z osobna kwalifikując go do barwy niebieskiej lub też nie. Użycie przestrzeni barw HSV jest dość oczywiste w tym przypadku, ponieważ poszczególne barwy zajmują w niej bardzo regularne obszary. Barwy niebieska, zielona oraz czerwona zmieniają się co 120 stopni podstawy stożka, dzięki czemu można je łatwo zdefiniować dzieląc podstawę na tercje. Każdej tercji przyporządkowywane są odpowiednie zakresy nasycenia i jasności, których czyste barwy zielona, niebieska i czerwona mają względnie duże wartości. Rysunek (*Rysunek 4-5*) przedstawia obraz po wstępnym wykryciu kolorów. Oprócz obszaru znacznika zawiera on jeszcze dodatkowe obszary, co jest zjawiskiem niepożądanym z punktu widzenia dalszego przetwarzania obrazu. Mogą to być zakłócenia lub inne przedmioty obecne na obrazie w kolorze niebieskim. Pierwotna wersja algorytmu, wykorzystywała morfologiczne operacje, w celu filtracji zakłóceń. Wadą operacji morfologicznych w zastosowaniu do filtracji obszaru maski jest słaba filtracja obcych konturów o dużych wielkościach. W innym przypadku, gdy znacznik znajduje się względnie daleko od kamery (zajmuje niewielki obszar na obrazie), istnieje niebezpieczeństwo odfiltrowania obszaru samego znacznika.



Rysunek 4-5. Obraz oryginalny na lewo, obraz po operacji wykrycia maski na prawo

Bardzo dobrym rozwiązaniem wyszczególnionych problemów okazało się zastosowanie filtracji za pomocą konturów. Operacja wykrywania konturów na obrazie binarnym zaimplementowana jest w bibliotece OpenCV poprzez funkcję *cvFindContours*. Umożliwia ona wykrycie jedynie konturów zewnętrznych, co w konsekwencji oznacza możliwość odfiltrowania kwadratów oraz trójkątów z obszaru zajmowanego przez sam znacznik

(*Rysunek 4-6*). Ma to szczególne znaczenie dla procesu segmentacji znacznika za pomocą histogramu opisanego w dalszej części rozdziału.

Aby ze wszystkich znalezionych konturów na obrazie wyodrębnić jedynie te odpowiadające w rzeczywistości kształtowi kwadratowemu zostają zastosowane niezmienniki momentowe, których zadaniem jest identyfikacja danego kształtu z różnej perspektywy widzenia. Spośród znalezionych elementów na obrazie, wybierany jest następnie największy możliwy i interpretowany jako obszar zajmowany przez znacznik na obrazie (tzw. maska znacznika). Dzięki takiemu rozwiązaniu w przypadku znalezienia na obrazie więcej niż jednego znacznika, algorytm zajmuje się analizą tego znajdującego się najbliżej.



Rysunek 4-6. Na lewo maska obrazu, na prawo maska po wykonaniu operacji filtrowania z wykorzystaniem kształtu konturów

Segmentacja znacznika mająca na celu wyodrębnienie poszczególnych jego elementów stanowi kolejny etap działania programu. Do tego celu konieczne jest przeprowadzenie operacji progowania, na obszarze obrazu zajmowanym przez znacznik. Operacja progowania wykonywana jest na obrazie monochromatycznym, który jest otrzymywany poprzez konwersję obrazu kolorowego za pomocą oprogramowania OpenCV. W pierwotnej wersji algorytmu wartość progu ustawiona była jako stała, co czyniło algorytm mało elastycznym dla zmiennych warunków oświetleniowych, różnej odległości oraz kąta obrotu znacznika od kamery. Automatyzacja tego procesu była jednym z celów pracy magisterskiej. Aby go osiągnąć przeprowadzona została seria eksperymentów dla czterech różnych sposobów dobierania progu za pomocą: analizy wielkości maski, metody iteracyjnej, progowania adaptacyjnego, analizy kształtu histogramu. Ostatecznie optymalnym dla problemu lokalizacji robota okazał się dobór progu za pomocą histogramu.

Dobór progu za pomocą wielkości maski

Pomysł doboru progu za pomocą wielkości maski polegał na zsumowaniu wszystkich pikseli, jakie zawierała maska znacznika. Jak wiadomo im znacznik znajdowałby się bliżej kamery tym większa liczba pikseli znalazłaby się w masce. Za pomocą doświadczalnego doboru kilku progów w zależności od wielkości maski, można precyzyjnie odróżnić szukane elementy od tła. Podczas badań udało się za pomocą zdefiniowania trzech różnych progów, lokalizować znacznik w odległości do trzech metrów od kamery. Metoda posiada szereg wad

mocno ograniczających jej zastosowanie w rozwiązywanym problemie, z których największą okazała się duża wrażliwość na panujące warunki oświetleniowe. W przypadku jej stosowania dla pomieszczeń o różnej światłości niezbędna byłaby korekcja progu w zależności od padającego światła w pomieszczeniu. Dodatkowo odchylenie znacznika od osi optycznej kamery również powoduje zmianę wartości progu, co wiązałoby się ze zdefiniowaniem odpowiednich progów uwzględniając zarówno kąt obrotu jak i odległość znacznika względem kamery.

Iteracyjny dobór progu

Iteracyjny dobór progu polega na rozpoczęciu progowania od pewnej ustalonej wartości. W przypadku uzyskania niezadowolających rezultatów wartość progu jest zwiększana lub zmniejszana i progowanie jest wykonywane ponownie, aż do uzyskania odpowiednich efektów. Metoda okazała się być najskuteczniejszą spośród testowanych metod doboru progu. Można ją zaimplementować w taki sposób by zawsze była zbieżna, poprzez ustawienie odpowiednio małego kroku iteracji. Z drugiej strony wymaga znacznych nakładów obliczeniowych, a co za tym idzie jest działą wolniej w porównaniu do metody doboru progu za pomocą histogramu. Iteracyjny dobór progu nie zapewnia ponadto odpowiedniego doboru progu w przypadku intensywnego refleksu światła. Taką wadę posiadają wszystkie metody, używające pojedynczego progu dla całej powierzchni znacznika.

Progowanie adaptacyjne

Zaletą progowania adaptacyjnego jest to, iż może zapewnić dobór odpowiedniego progu w momencie padania intensywnego refleksu światła na znacznik. Z uwagi na sposób działania całości algorytmu skuteczności progowania adaptacyjnego pod tym kryterium nie udało się przetestować. Wynika to z faktu, iż intensywny refleks światła zakłóca wykrywanie poszczególnych kolorów na obrazie, a to z kolei uniemożliwia prawidłowe zlokalizowanie maski znacznika.

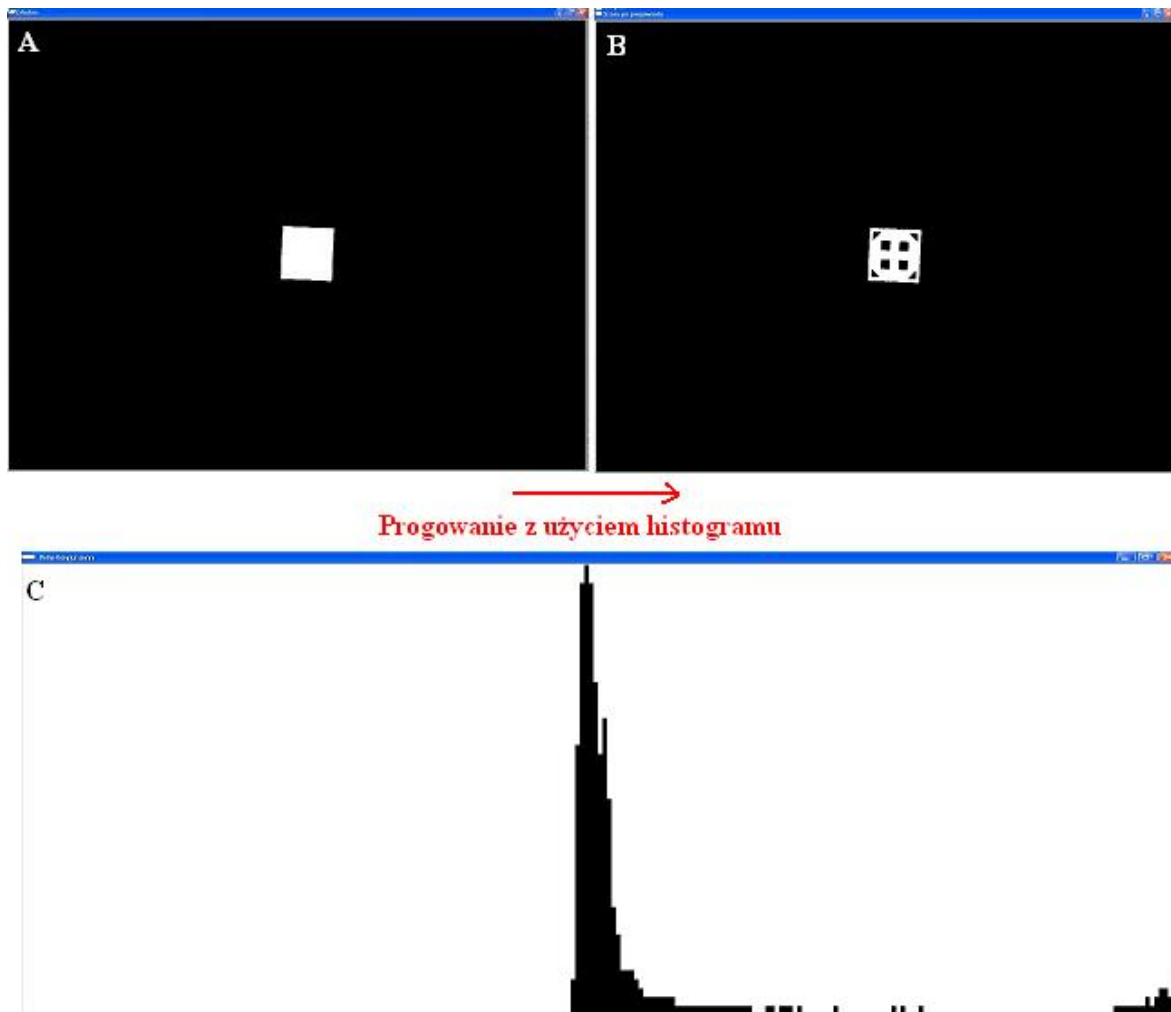
W przypadku regularnego oświetlenia, wartości parametrów progowania adaptacyjnego (wielkość otoczenia, minimalna wariancja) trzeba dobierać w zależności od odległości i kąta obrotu znacznika względem kamery, co zastosowanie progowania adaptacyjnego czyni mało praktycznym i trudnym w porównaniu do dającego podobne efekty, a znacznie prostszego w implementacji doboru progu za pomocą obrazu histogramu.

Dobór progu za pomocą histogramu

Istotą doboru progu za pomocą histogramu, jest obliczenie histogramu dla fragmentu obrazu zaznaczonego maską. Jeśli maska została precyzyjnie wykryta i nie zawiera, żadnych innych elementów oprócz rzeczywistego obszaru znacznika, to na histogramie powinny pojawić się dwa wyraźne poziomy jasności. Jednym z tych poziomów jest kolor tła (błękit), drugim kolor elementów znacznika (biel). Aby precyzyjnie oddzielić jedne elementy od drugich na obrazie wystarczy użyć progu odpowiadającego „dolinie” na histogramie. Skuteczność metody okazała się w praktyce niewiele gorsza od skuteczności metody iteracyjnej, przy znacznie większej wydajności obliczeniowej. Warunkiem koniecznym dla prawidłowego działania metody jest to, iż zarówno tło jak i obiekt szukany powinny mieć

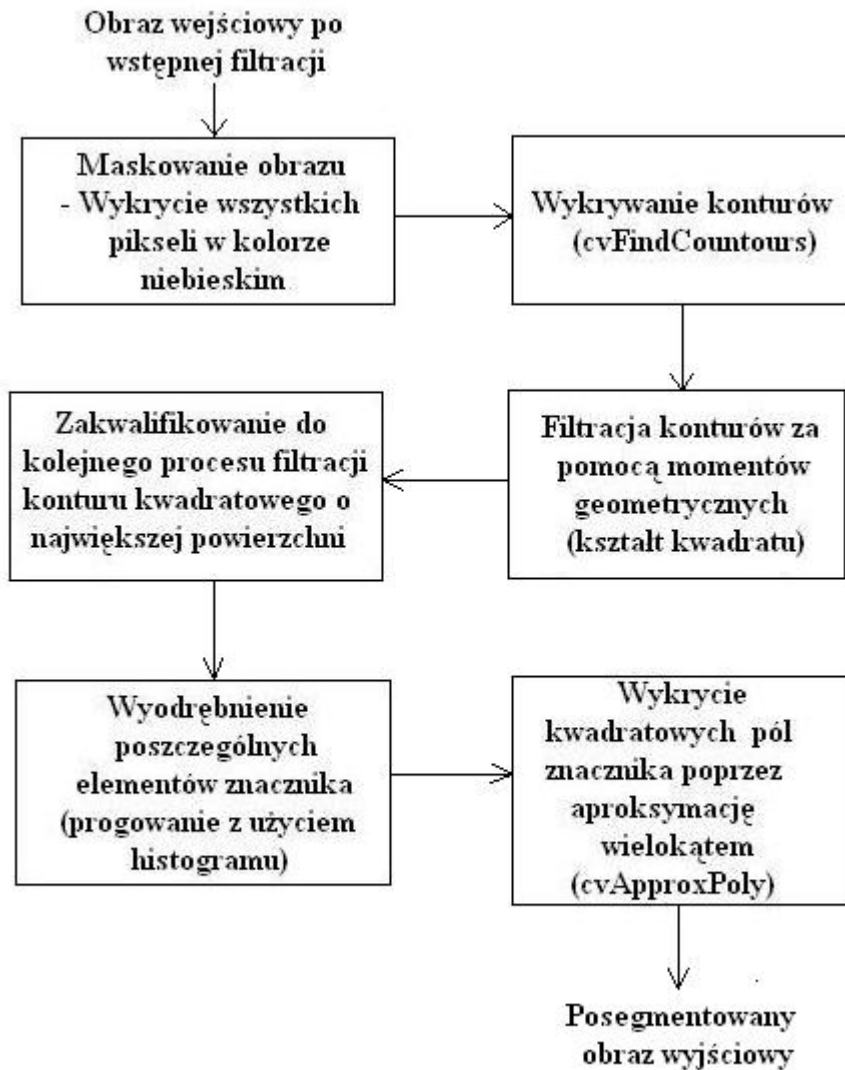
barwę jednorodną, znacznie różniącą się od siebie jasnością. W przeciwnym wypadku obraz histogramu jest „płaski” i niemożliwe staje się dobranie optymalnego progu na jego podstawie. Metoda w porównaniu do innych testowanych metod jest mało wrażliwa na panujące warunki oświetleniowe, kąt obrotu i odległość znacznika względem kamery.

W wyniku testów czterech metod progowania zdecydowanie najlepszą okazała się metoda doboru progu przy pomocy obrazu histogramu. Wskutek operacji progowania obraz posiada wyraźnie oddzielone od tła poszczególne obszary służące do lokalizacji oraz identyfikacji znacznika. Oddzielone kwadraty są na sam koniec aproksymowane wielokątem za pomocą funkcji `cvApproxPoly`, dzięki czemu obszar przez nie zajmowany jest możliwie precyzyjnie wyznaczony. Ma to bardzo duże znaczenie w dalszym procesie lokalizacji. Efekty procesu segmentacji znacznika za pomocą histogramu przedstawia rysunek (*Rysunek 4-7*).



Rysunek 4-7. Progowanie z użyciem histogramu A – obszar przed progowaniem, B – obszar po progowaniu, C – obraz rzeczywistego histogramu użytego do progowania

Podsumowując wykorzystany algorytm segmentacji jest skomplikowanym, wielostopniowym procesem numerycznym, angażującym bardzo dużą moc obliczeniową procesora. Stanowi on bardzo istotne ogniwo całego algorytmu lokalizującego kamerę względem znacznika, a jego działanie krok po kroku zostało przedstawione na schemacie (*Rysunek 4-8*).

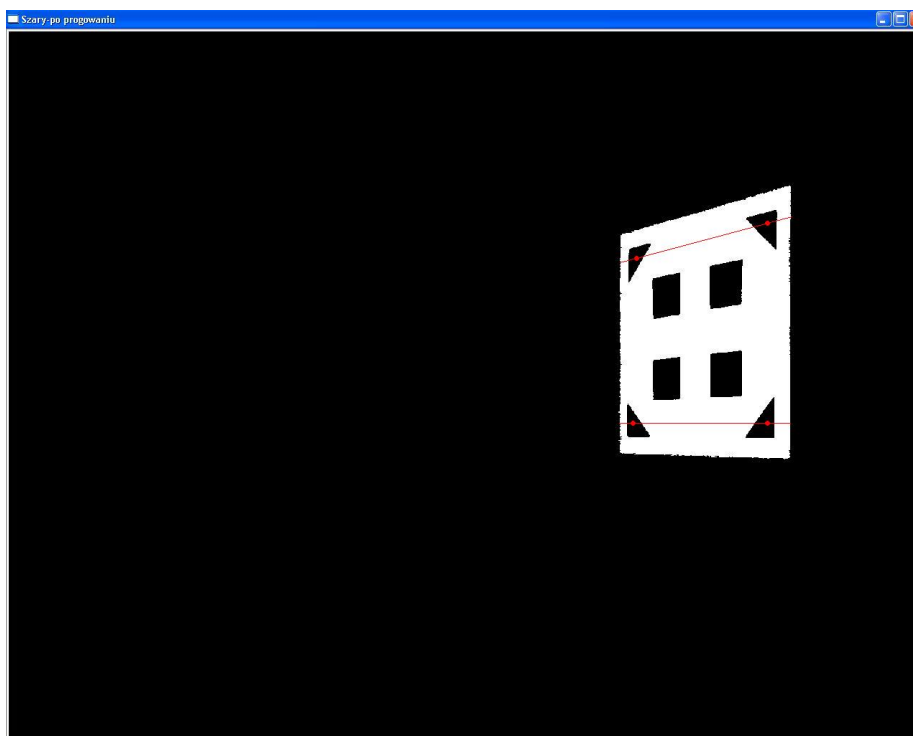


Rysunek 4-8. Schemat działania algorytmu segmentacji

4.5. Analiza obrazu

W procesie analizy obrazu można wyróżnić dwa etapy składowe: identyfikację numeru znacznika oraz lokalizację znacznika w przestrzeni.

Proces identyfikacji numeru znacznika rozpoczyna się poprzez wykrycie współrzędnych rogów maski. Ponieważ poszczególne trójkąty, będące elementami kodu binarnego występują zawsze w tych samych miejscach znacznika, skalując odpowiednio obraz wyznaczane są na obrazie obszary, na których występują elementy identyfikacyjne. Istotę skalowania obrazu przedstawia rysunek (*Rysunek 4-9*).

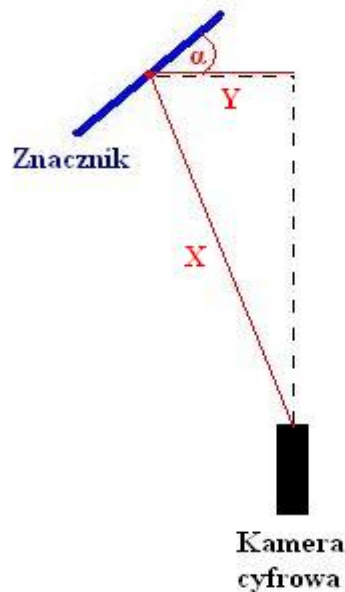


Rysunek 4-9. Znacznik po operacji progowania wraz z geometrycznie wyznaczonymi obszarami służącymi do identyfikacji

Prowadzone są dwie proste: jedna równoległe do górnej krawędzi, druga równoległe do dolnej krawędzi. Odległości prostych od poziomych krawędzi odpowiadają 0,133 wysokości całego znacznika na obrazie (ponieważ geometryczny środek trójkąta znajduje się 2 cm od poziomej i pionowej krawędzi na znaczniku o rozmiarach 15 cm, $2/15 = 0,133$). Następnie obecność identyfikacyjnych elementów jest sprawdzana w punktach odpowiadającym geometrycznym środkom trójkątów identyfikacyjnych (na obydwu wyznaczonych odcinkach punkty odpowiadające 0,133 długości znacznika odpowiednio od lewej i prawej krawędzi). Obszary te są weryfikowane na obecność trójkąta poprzez zastosowanie maski obrazu uwzględniającej sąsiedztwo dziewięciu pikseli. Jeśli maska zastosowana wykrywa wszystkie piksele w kolorze czarnym na obrazie binarnym, wtedy wartość binarna trójkąta jest uwzględniana przy wyliczaniu numeru znacznika, w przeciwnym wypadku wartość trójkąta wynosi 0. Uwzględnienie sąsiedztwa zamiast sprawdzania koloru pojedynczego piksela powoduje, że algorytm jest dużo bardziej niezawodny, gdyż eliminuje występowanie przypadkowych zakłóceń.

Proces lokalizacji znacznika w przestrzeni trójwymiarowej ma na celu precyzyjne wyznaczenie pozycji kamery względem znacznika. Lokalizacja znacznika odbywa się na podstawie środków czterech kwadratów, które wyznaczane są na podstawie liczenia momentów geometrycznych oraz piątego pomocniczego punktu, którego współrzędne stanowią geometryczną średnią z punktów stanowiących środki poszczególnych kwadratów (środek znacznika). Podczas testów sprawdzana była również możliwość udoskonalenia istniejącego algorytmu, szacując położenie na podstawie dziewięciu punktów, dodając do istniejących współrzędnych znacznika cztery rogi maski. Jak się okazało podczas testów zmniejszyło to zdecydowanie dokładność pomiarów, gdyż precyzja wyznaczenia współrzędnych rogów maski na obrazie uchwyconym przez kamerę jest zbyt mała. Estymacja

położenia przestrzennego znacznika opiera się na funkcji `cvFindExtrinsicCameraParams2` ze środowiska OpenCV, która jako parametry wejściowe używa parametrów wewnętrznych kamery, współczynników zniekształceń oraz współrzędnych środków kwadratów na obrazie, zwracając wektory rotacji i translacji dla znacznika. Położenie znacznika jest określone przez 6 współrzędnych określających rotację i translację znacznika w przestrzeni trójwymiarowej. Z punktu widzenia działania algorytmu lokalizującego robota na płaszczyźnie wystarczające staje się użycie jedynie dwóch parametrów translacji (X, Y) oraz jednego parametru rotacji (α) przedstawionych na rysunku (**Rysunek 4-10**).



Rysunek 4-10. Parametry pozycji przestrzennej wyznaczone przez algorytm lokalizacji kamery względem znacznika

4.6. Testy działania algorytmu

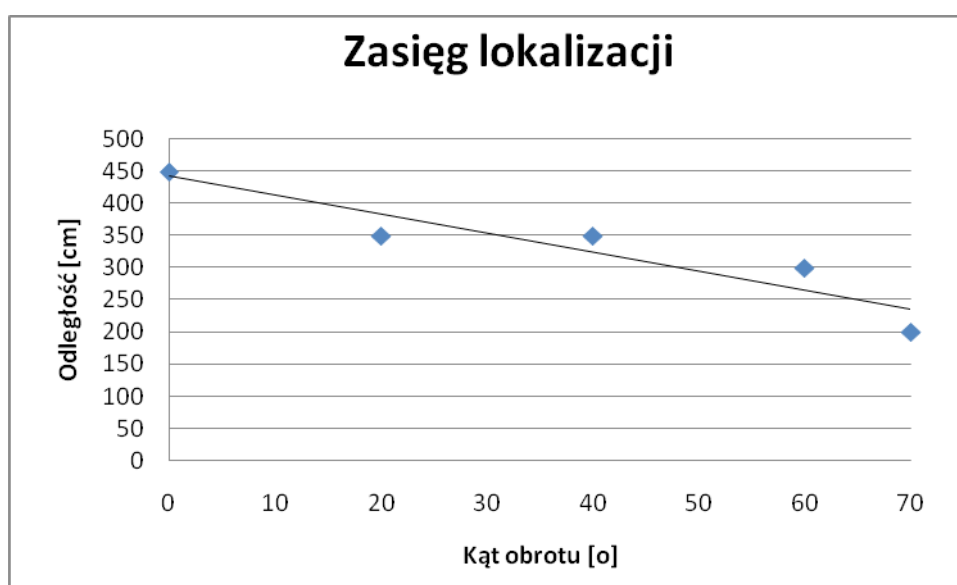
Wykonane testy algorytmu miały na celu ocenę skuteczności jego działania. Sposób przeprowadzenia testów, polegał na umieszczeniu znacznika w miejscu pod określonym kątem i oddalania od niego stopniowo kamery. Odległość znacznika od kamery wyznaczana była za pomocą taśmy mierniczej. Odpowiednie ustawienie katowej orientacji znacznika następowało poprzez użycie kątomierza, a następnie zweryfikowanie ustawienia poprzez pomiar kąta kamerą z odległości 50 cm. Pomiary przeprowadzone zostały w godzinach wieczornych w laboratorium, przy sztucznym regularnym świetle. Ich rezultaty mogą zmieniać się w stosunku do panujących warunków oświetleniowych, a także otoczenia znacznika. Wyniki testów zostały umieszczone w tabeli:

Tabela 4-1. Wyniki przeprowadzonych testów algorytmu wizyjnego

Odległość znacznika od kamery[cm]	Kąt obrotu znacznika [°]	Błąd wyznaczenia odległości	Błąd wyznaczenia kąta	Identyfikacja znacznika
50	0	< 1cm	+/-4 °	Dobra
100	0	< 1cm	+/-4 °	Dobra
150	0	+/-1cm	+/-4 °	Dobra
200	0	+/-1cm	+/-4 °	Dobra
250	0	+/-2cm	+/-4 °	Niewielkie błędy
300	0	+/-3cm	+/-4 °	Nieemożliwa
350	0	+/-3cm	+/-4 °	Nieemożliwa
400	0	+/-3,5 cm	+/-4 °	Nieemożliwa
450	0	+/-5 cm	+/-4 °	Nieemożliwa
50	20	< 1cm	< 1 °	Dobra
100	20	< 1cm	< 1 °	Dobra
150	20	+/-1cm	+/-1 °	Dobra
200	20	+/-1cm	+/-1 °	Niewielkie błędy
250	20	+/-2cm	+/-2 °	Nieemożliwa
300	20	+/-3cm	+/-2 °	Nieemożliwa
350	20	+/-3cm	+/-2 °	Nieemożliwa
50	40	< 1cm	+/-1 °	Dobra
100	40	< 1cm	+/-1 °	Dobra
150	40	+/-1cm	+/-1 °	Dobra
200	40	+/-1cm	+/-2 °	Niewielkie błędy
250	40	+/-2cm	+/-2 °	Nieemożliwa
300	40	+/-3cm	+/-2 °	Nieemożliwa
350	40	+/-3cm	+/-2 °	Nieemożliwa
50	60	< 1cm	+/-1 °	Dobra
100	60	< 1cm	+/-1 °	Dobra
150	60	+/-1cm	+/-1 °	Niewielkie błędy
200	60	+/-1cm	+/-1 °	Nieemożliwa
250	60	+/-2cm	+/-1 °	Nieemożliwa
300	60	+/-3cm	+/-1 °	Nieemożliwa
50	70	< 1cm	+/-1 °	Dobra
100	70	< 1cm	+/-1 °	Dobra
150	70	+/-1cm	+/-1 °	Niewielkie błędy
200	70	+/-1cm	+/-1 °	Nieemożliwa

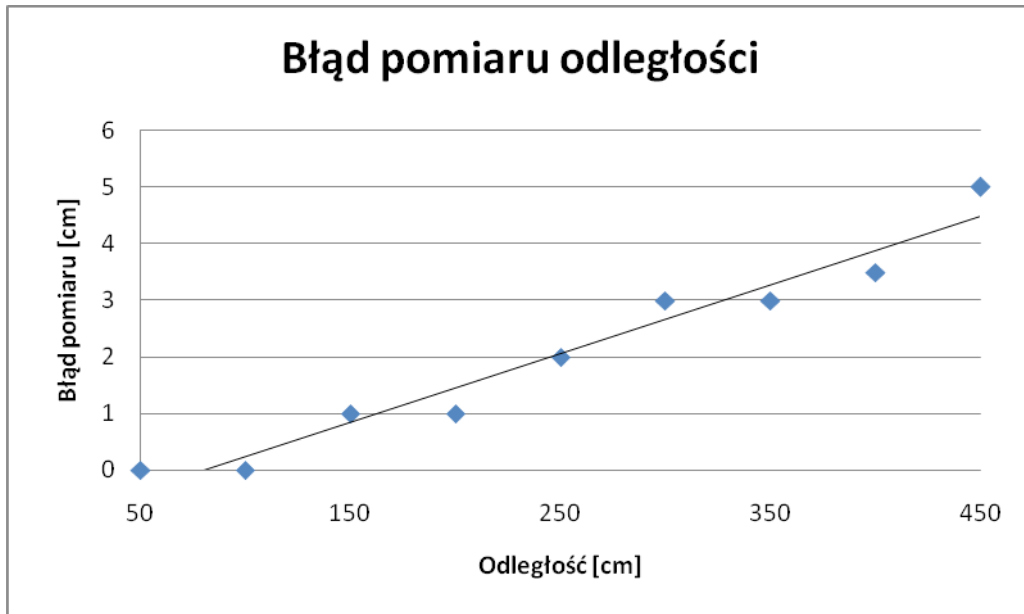
Jak wynika z przeprowadzonych testów dokładność pomiarów obarczona jest stosunkowo niewielkim błędem, wystarczającym do precyzyjnego zlokalizowania robota w otoczeniu. W granicznych przypadkach błędy nie przekraczały pięciu centymetrów odległości i czterech stopni orientacji kątowej. Testy potwierdziły tezę, że właściwa identyfikacja znacznika oraz lokalizacja jest bardziej zawodna dla znacznika znajdującego się w dużej odległości i pod dużym kątem wobec kamery. Wyjątkiem od tej reguły jest błąd szacowania kąta obrotu, który jest największy, gdy znacznik znajduje się dokładnie naprzeciwko kamery i maleje wraz ze wzrostem kąta obrotu. Zależność poszczególnych wielkości od siebie i ich trendy podsumowują wykresy (*Rysunek 4-11*, *Rysunek 4-12*, *Rysunek 4-13*, *Rysunek 4-14*).

Jak wynika z przeprowadzonych testów maksymalny zasięg lokalizacji wynosi 4,5 metra. Odległość ta zmienia się w zależności od kąta obrotu (*Rysunek 4-11*). Osiąga maksimum dla znacznika znajdującego się prostopadłe do osi optycznej kamery i maleje wraz ze wzrostem odchylenia

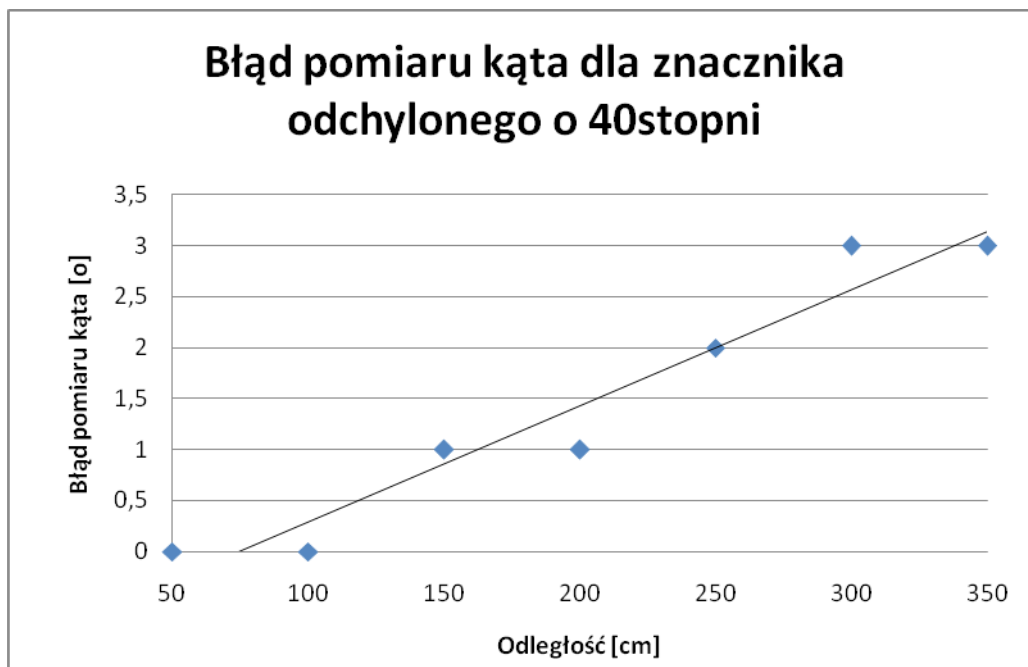


Rysunek 4-11. Zasięg lokalizacji dla poszczególnych kątów obrotu znacznika

Zarówno bezwzględny błąd pomiaru odległości jak i kąta rośnie wraz ze wzrostem dystansu kamery od znacznika (*Rysunek 4-12*, *Rysunek 4-13*). Wielkość bezwzględnego błędu pomiaru odległości jest niezależna od kąta obrotu znacznika. Zerowy błąd pomiarowy na wykresach oznacza w rzeczywistości odpowiednio błąd poniżej 1cm odległości i 1 ° orientacji kątowej. Należy podkreślić ponadto, iż względny błąd pomiaru odległości nie przekraczał 1,2% co stanowi bardzo dobry rezultat.

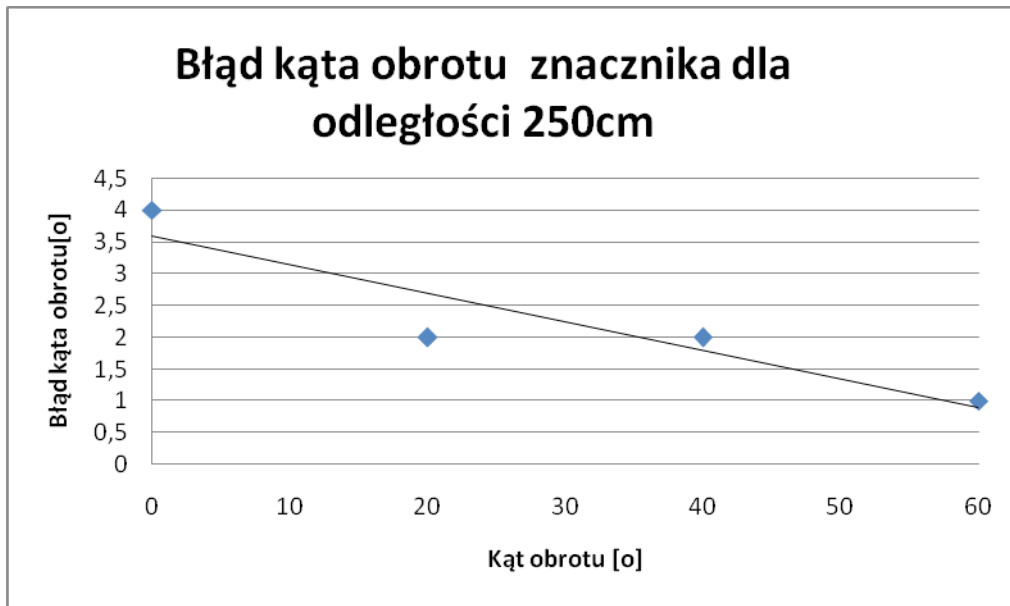


Rysunek 4-12. Zależność błędu bezwzględnego pomiaru odległości znacznika od odległości pomiędzy znacznikiem i kamerą



Rysunek 4-13. Zależność błędu bezwzględnego pomiaru kąta orientacji znacznika odchylonego o 40 stopni względem osi optycznej kamery, od jego odległości od kamery

Na podstawie eksperymentalnych wyników można również stwierdzić, iż błąd kąta maleje wraz ze wzrostem kąta obrotu znacznika (*Rysunek 4-14*).



Rysunek 4-14. Zależność błędu bezwzględnego pomiaru kąta orientacji znacznika znajdującego się w odległości 250cm względem kamery, od kąta obrotu znacznika w stosunku do osi optycznej kamery

Algorytm znacznie lepiej radzi sobie z lokalizowaniem pozycji znacznika, niż jego identyfikacją. Wynika to z faktu, iż w przypadku dużej odległości znacznik zajmuje stosunkowo niewielką część obrazu i na jego poszczególne części przypada po kilka pikseli obrazu. Stąd częste kłopoty z właściwą interpretacją znacznika. W takich przypadkach wielkość maski często bywa wykrywana nieprecyzyjnie i potrafi „zlewać się” z trójkątami identyfikacyjnymi (*Rysunek 4-15*).



Rysunek 4-15. Obraz znacznika na obrazie uchwyconego z odległości 4ech metrów

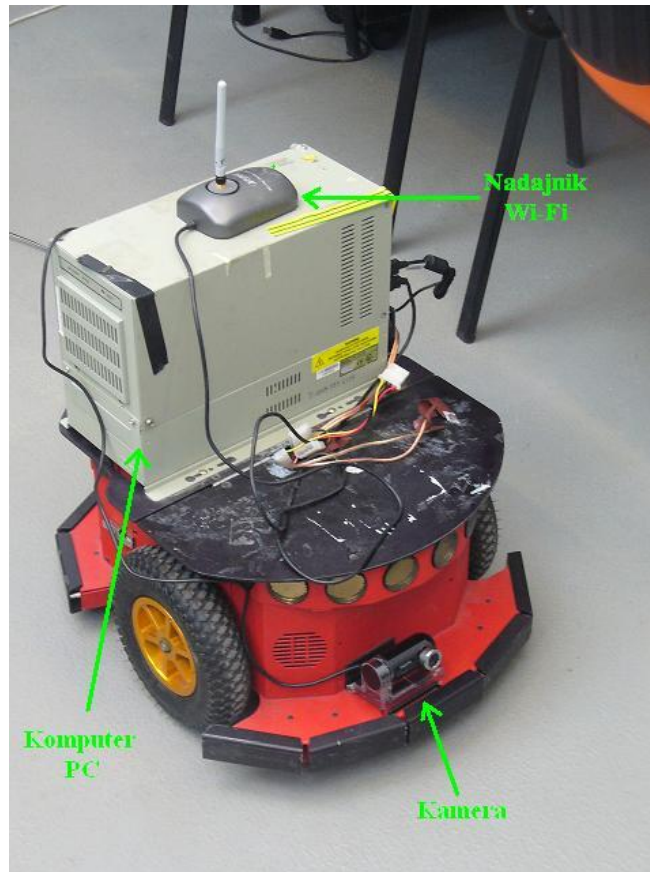
Na podstawie przeprowadzonych badań można stwierdzić, iż procesem istotnie ograniczającym działanie algorytmu jest segmentacja. Podczas testów systemu dla dużych odległości kamery od znacznika (nie uwzględniono w tabeli) zdarzało się, że lokalizacja przeprowadzona była poprawnie (z niewielkim błędem), jednak prawidłowe wykrycie znacznika na obrazie następowało jednorazowo na kilkanaście klatek obrazu. Innym aspektem zawodności algorytmu jest to, iż nie nadaje się on do pomieszczeń, w których występuje dużo kwadratowych elementów w kolorze niebieskim (odpowiadających kształtowi i kolorowi znacznika). W takim przypadku znalezienie obszaru występowania znacznika może okazać się zadaniem bardzo trudnym i algorytm nie jest przeznaczony dla tego typu otoczenia.

5. Zastosowanie systemu wizyjnego w środowisku robota mobilnego

W celu praktycznego użycia stworzonego algorytmu wizyjnego do lokalizacji robota kamera zostaje umieszczona na robocie mobilnym Pioneer 3DX, a znaczniki zostają rozmieszczone w określonych miejscach w otoczeniu. Poniższy rozdział ma na celu szczegółowy opis zbudowanego systemu lokalizacji robota mobilnego, którego istota polega na zlokalizowaniu robota znajdującego się w nieznanym punkcie pomieszczenia, a następnie umożliwienie jego przemieszczania po określonej trasie.

5.1. Opis robota mobilnego Pioneer 3DX

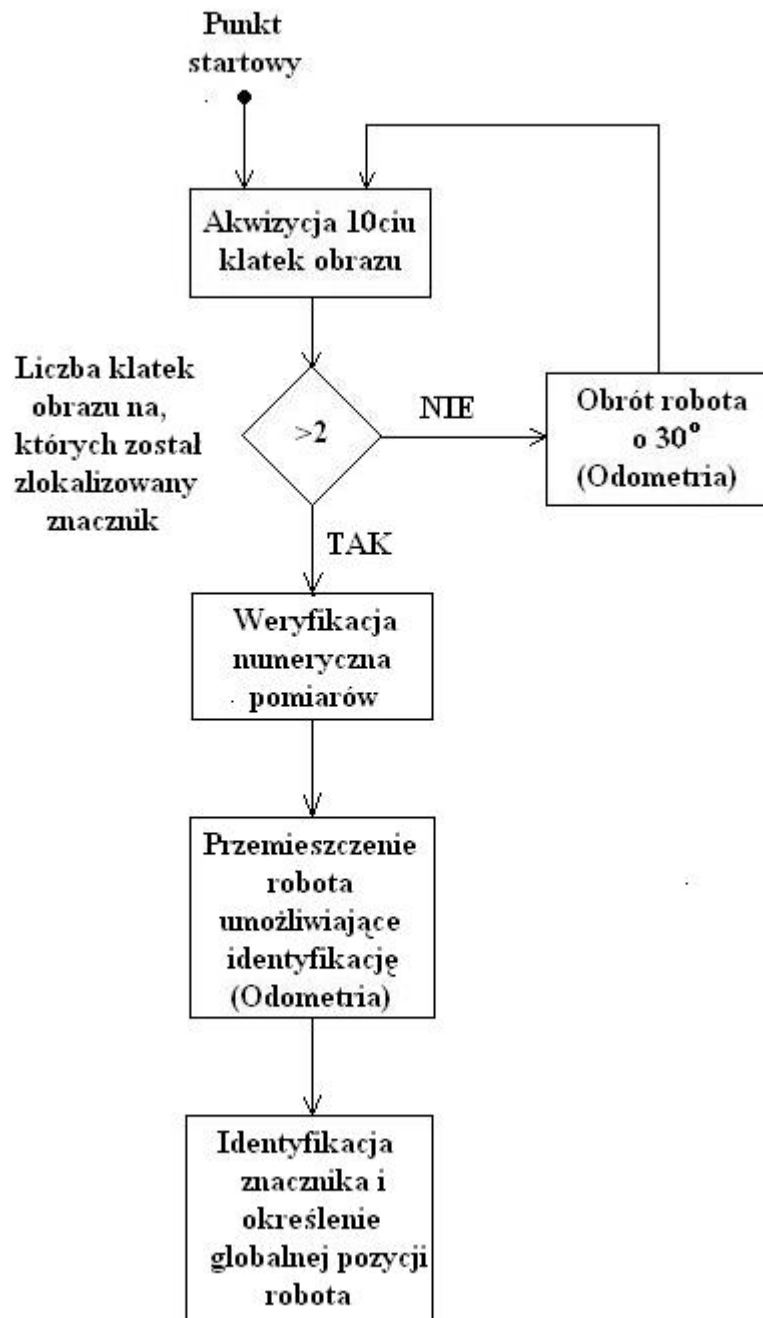
Pioneer 3DX jest robotem dostępnym na rynku do celów naukowo-badawczych i edukacyjnych. Używana do testów wersja robota (*Rysunek 5-1*) wyposażona została w 2 silniki napędowe wyposażone w enkodery, pakiet baterii do zasilania robota, dwa pierścienie po osiem czujników sonarowych każdy, czujniki kontaktowe umieszczone na zderzakach oraz mikrokontroler umożliwiający sterowanie robotem [14]. Do sterowania robotem użyty został ponadto komputer klasy PC znajdujący się na obudowie robota, który komunikuje się bezpośrednio z mikrokontrolerem poprzez szeregowy interfejs RS-232. Jako urządzenie peryferyjne (poprzez interfejs USB) w komputerze zostały dołączone bezprzewodowa karta sieciowa oraz kamera Creative Live Cam Optia. Środowiskiem sterującym robota jest ARIA (ang. Advanced Robotics Interface for Applications), używające języka C++, obsługiwane przy pomocy Microsoft Visual Studio. Aby ułatwić komunikację pomiędzy użytkownikiem, a robotem wykorzystywany jest ponadto jeden z komputerów znajdujących się w laboratorium, który za pomocą sieci bezprzewodowej i oprogramowania zdalnego pulpitu dostępnego w systemie operacyjnym Windows umożliwia połączenie się z robotem.



Rysunek 5-1. Robot Pioneer 3DX

5.2. Opis algorytmu sterowania robotem mobilnym w pomieszczeniu laboratorium

System sterujący robotem na potrzeby lokalizacji robota mobilnego używa do swojego działania kamery wraz z algorytmem przetwarzania obrazu oraz sensorów odometrycznych. Początkowo robot znajdując się w nieznanym punkcie pomieszczenia, wykorzystuje kamerę do określenia swojej wstępnej pozycji. Za ten etap odpowiada pierwsza część algorytmu lokalizującego, której proces działania przedstawiony jest na schemacie blokowym (*Rysunek 5-2*):

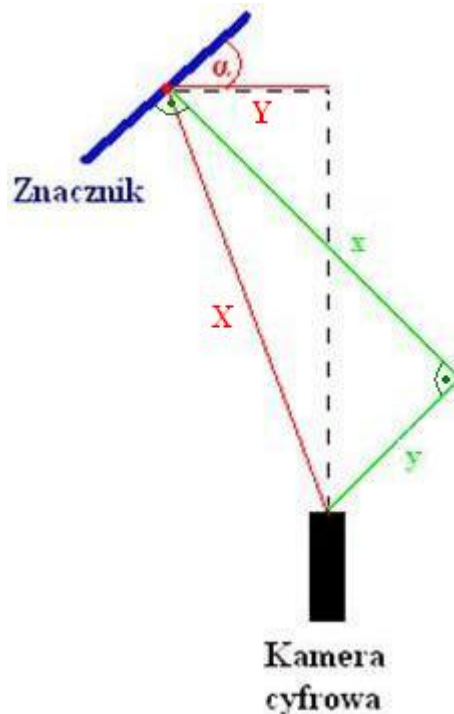


Rysunek 5-2. Schemat blokowy procesu określającego położenie robota mobilnego w otoczeniu

Algorytm w momencie rozpoczęcia działania wychwytuje 10 klatek obrazu i jeśli przynajmniej na trzech z nich znacznik nie zostanie zlokalizowany, wtedy robot obraca się o 30 stopni z użyciem sensorów odometrycznych. Następnie próbuje znaleźć znacznik na obrazie stojąc pod inną pozycją kątową. W przypadku zlokalizowania znacznika na przynajmniej trzech klatkach obrazu, następuje weryfikacja pomiarów. Jej celem jest odrzucenie pomiarów obarczonych dużym błędem. W tym celu uwzględniane są współrzędne (X , Y , α) dla każdej klatki na której znacznik został zlokalizowany i następuje ich porządkowanie w rosnącym ciągu od najmniejszej do największej wartości. Następnie spośród trzech ciągów wartości wybierane są wartości środkowe (mediany) X , Y , α . Takie

rozwiązanie zwykle sprawdza się dużo lepiej niż uśrednianie wartości wszystkich pomiarów, ponieważ w przypadku dużego błędu jednej współrzędnej i małej liczby przetwarzanych obrazów, pomiar byłby obciążony znacznym błędem. Z drugiej strony w przypadku wszystkich prawidłowych pomiarów mediana nie odbiega w znaczący sposób od wartości uśrednionej. W praktyce można udoskonalić system o bardziej skomplikowany algorytm odrzucania błędnych pomiarów, jednak zastosowane, proste rozwiązanie sprawdza się znakomicie, co wykazały przeprowadzone eksperymenty. Wszelkie próby ingerencji mogłyby jedynie nieznacznie poprawić jego dokładność.

Następnym krokiem algorytmu jest przemieszczenie robota z zastosowaniem odometrii, tak by w efekcie znajdował się naprzeciwko znacznika w odległości 70 cm, co stanowi optymalny dystans dla procesu identyfikacji. Do realizacji tego celu otrzymane wyniki zwrócone przez algorytm przetwarzania obrazu transformowane są do wygodniejszego z perspektywy pracy z robotem układu współrzędnych (*Rysunek 5-3*).



Rysunek 5-3. Układy współrzędnych używane w lokalizacji robota względem znacznika
gdzie:

$$x = X * \cos \left(\alpha + \arcsin \left(\frac{Y}{X} \right) \right)$$

$$y = X * \sin \left(\alpha + \arcsin \left(\frac{Y}{X} \right) \right)$$

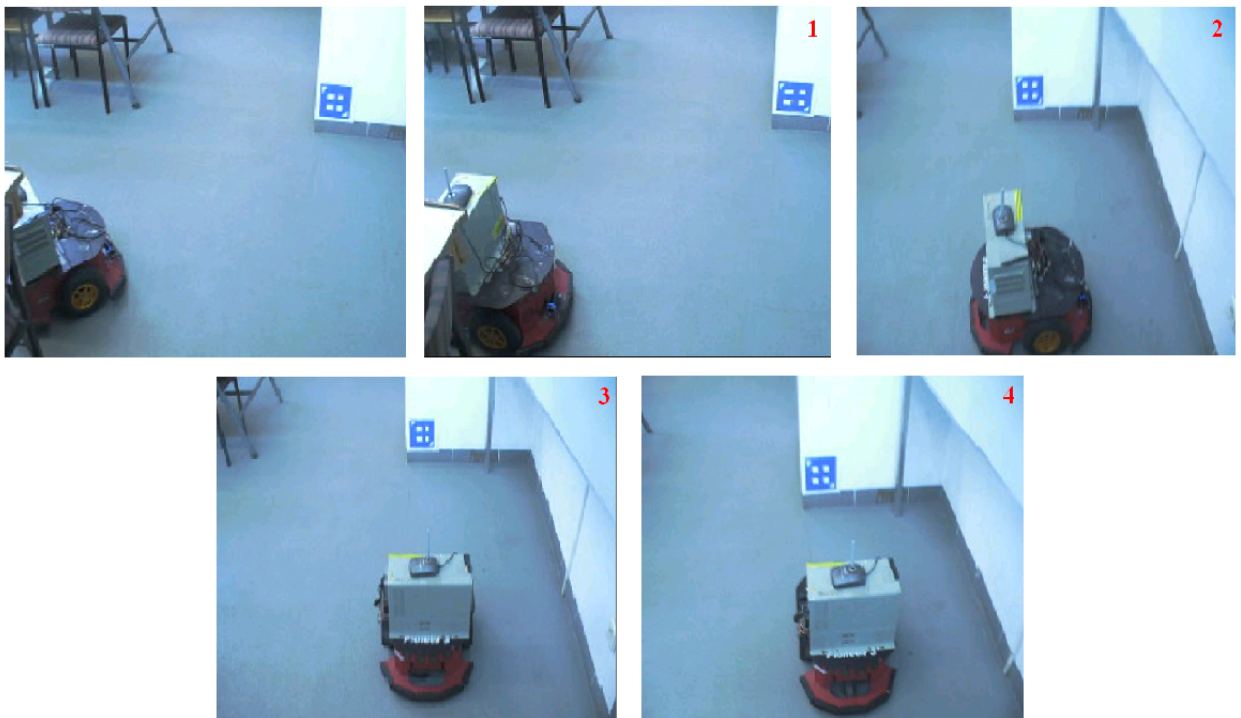
Dodatkowo, w procesie przemieszczenia robota należy uwzględnić zmianę pozycji kamery w wyniku skrętu. Wynika to z faktu, iż kamera nie znajduje się w centralnym punkcie osi obrotu robota, tylko na przednim zderzaku (oddalonym o 26 cm [14] od osi skrętu), zatem zmienia swoje położenie względem znacznika podczas skrętu robota. Uwzględniając tą zależność wyprowadzane są rzeczywiste wzory na odległość jaką robot musi przemierzyć.

$$x_r = X * \cos(\alpha + \arcsin(\frac{Y}{X})) - 26 * \cos(\alpha)$$

$$y_r = X * \sin(\alpha + \arcsin(\frac{Y}{X})) - 26 * \sin(\alpha)$$

Następnie w celu przemieszczenia się naprzeciwko znacznika na odległość równą 70cm robot wykonuje kolejno szereg poleceń bazując na pomiarze odometrycznym oraz parametrach x_r , y_r , α wyznaczonych za pomocą kamery (**Rysunek 5-4**):

1. obraca się o kąt α (tak by był skierowany równolegle do płaszczyzny wyznaczonej przez znacznik)
2. przemierza odległość y_r
3. odwraca się o kąt by 90 stopni przodem do znacznika
4. przemierza odległość równą ($x_r - 70\text{cm}$)

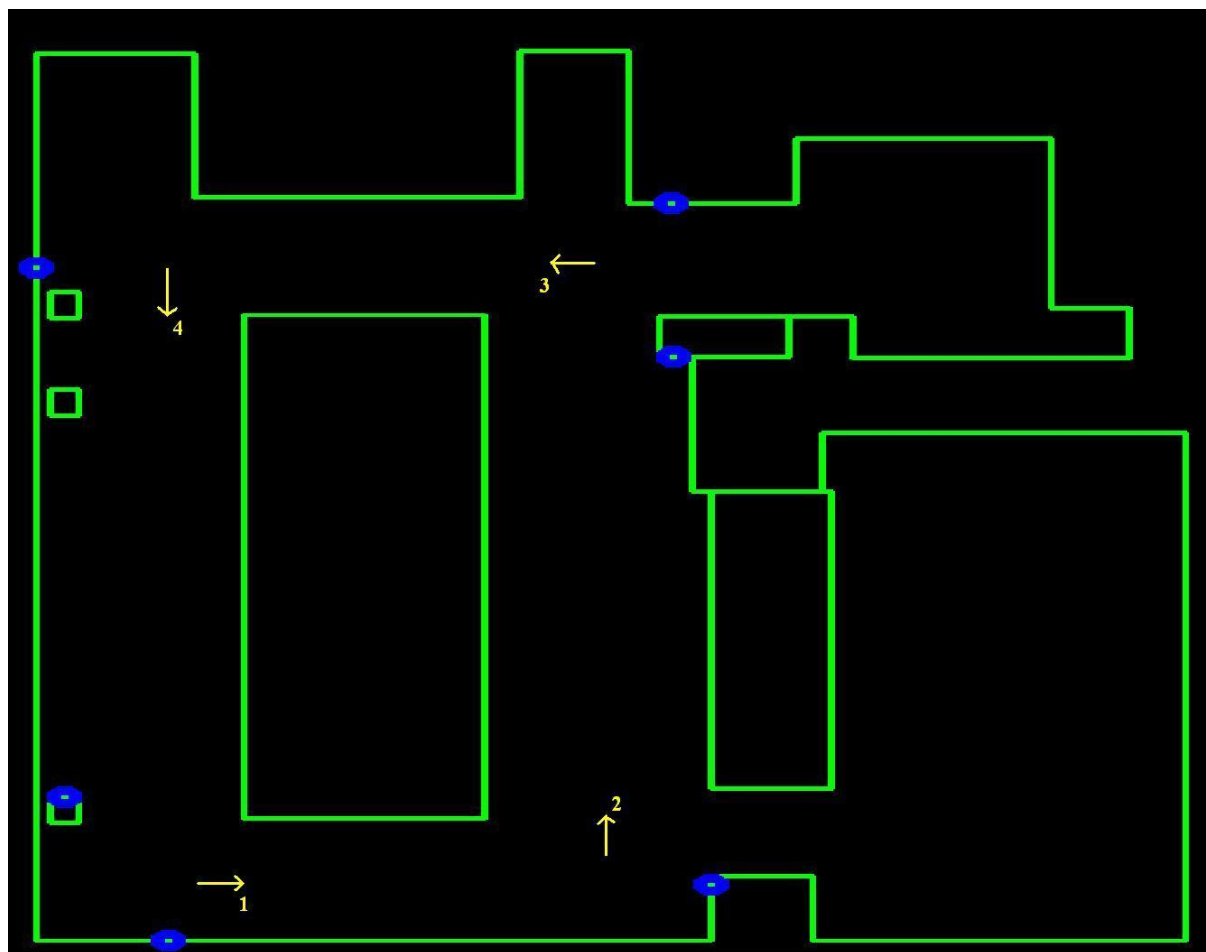


Rysunek 5-4. Przemieszczenie robota względem znacznika w procesie identyfikacji

Robot znajdując się w odległości 70 cm naprzeciwko znacznika, przystępuje do procesu identyfikacji, podczas którego wychwytywane jest kolejne 10 klatek obrazu. Numer znacznika pojawiający jest najczęściej spośród otrzymanych pomiarów zostaje uznany przez algorytm faktycznym numerem znacznika. Praktycznie algorytmowi identyfikacji nie zawsze udaje się poprawnie posegmentować znacznik na pojedynczym obrazie, głównie z powodu złego doboru wstępnej filtracji obrazu. W takim przypadku algorytm działa w pętli do momentu uchwycenia 10 poprawnych pomiarów, dając czas na manualną regulację parametrów kamery.

Jeśli znacznik zostaje zidentyfikowany poprawnie następuje odwołanie się algorytmu do bazy danych o położeniu znaczników w terenie. Znajac pozycję wobec zidentyfikowanego znacznika aktualna pozycja robota zostaje naniesiona na mapę pomieszczenia. Zbudowana

mapa (**Rysunek 5-5**), odzwierciedla faktyczny kształt pomieszczenia Laboratorium Teorii Sterowania w gmachu Starej Kotłowni Politechniki Warszawskiej, w którym przeprowadzone zostały eksperymenty.



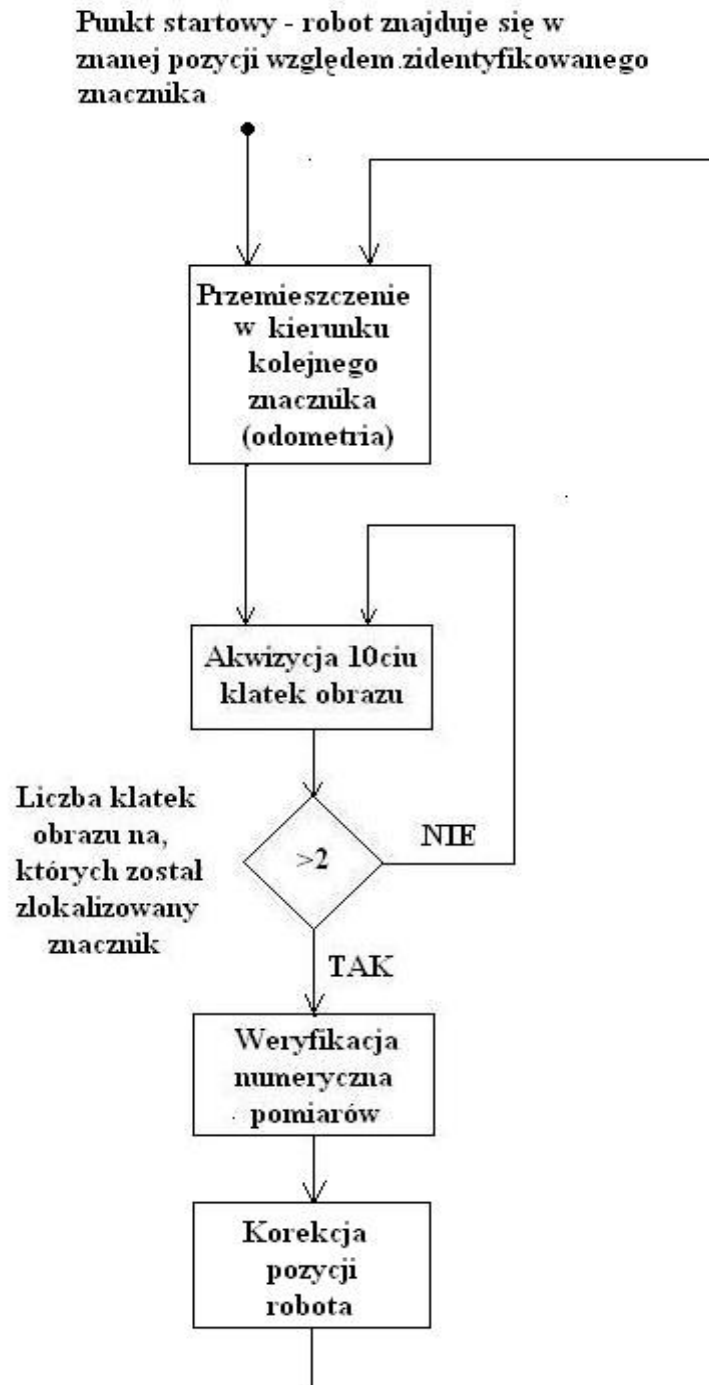
Rysunek 5-5. Mapa otoczenia robota wraz z rozmieszczonymi znacznikami

Niebieskie elipsy na mapie odzwierciedlają położenie znaczników, zielone linie odzwierciedlają kształty przeszkód (ścian, stołów, szaf) żółte strzałki natomiast określają miejsca, z których zrobione zostały fotografie pokazujące rzeczywisty obraz pomieszczenia (**Rysunek 5-6**).



Rysunek 5-6. Zdjęcia obrazujące pomieszczenie Laboratorium Teorii Sterowania w gmachu Starej Kotłowni Politechniki Warszawskiej

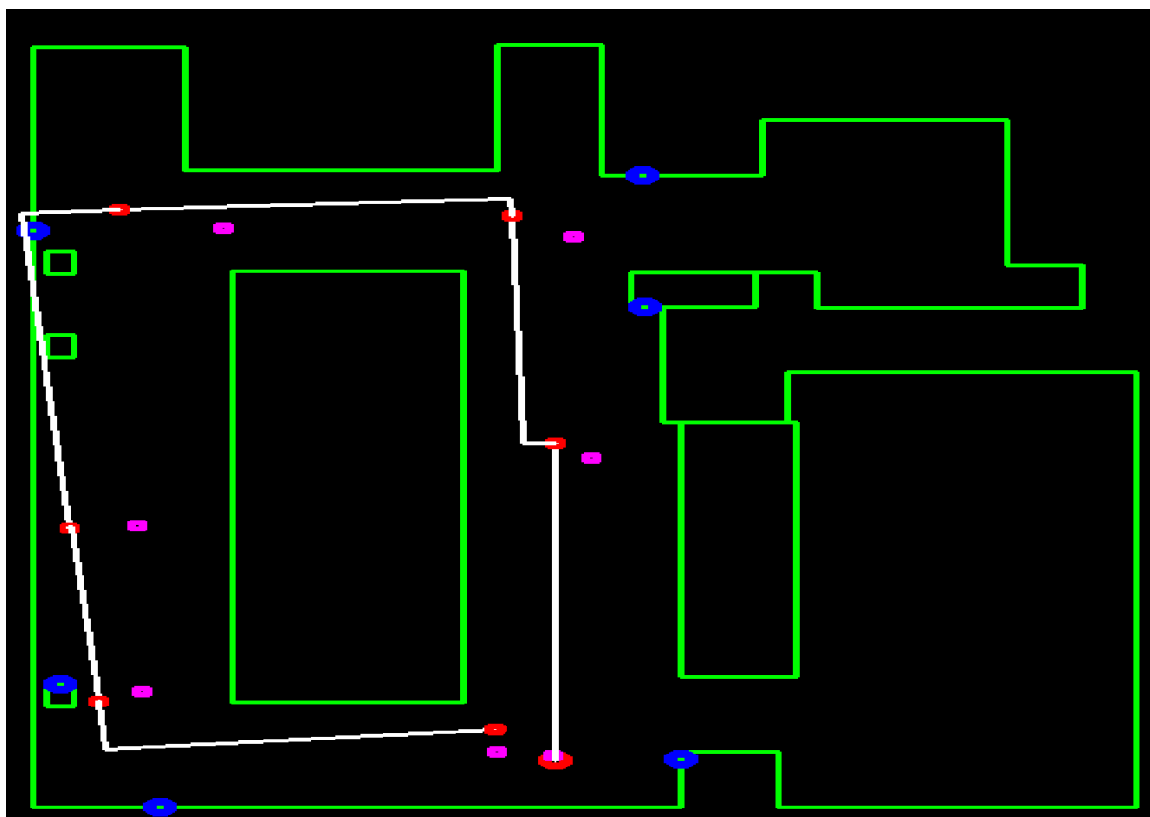
Druga część algorytmu systemu sterowania robotem mobilnym odpowiada za proces przemieszczenia robota wokół pomieszczenia po określonym torze. Robot w tym celu porusza się na podstawie pomiarów z czujników odometrycznych, co pewien czas weryfikując swoją pozycję poprzez odczyt obrazu z kamery. Kierunek ruchu robota przedstawiają poszczególne strzałki na mapie otoczenia (**Rysunek 5-5**). Faktycznie do rozwiązania tego problemu zostało użytych sześć z ośmiu stworzonych znaczników. Schemat blokowy algorytmu przemieszczenia się robota po pomieszczeniu laboratorium przedstawia rysunek (**Rysunek 5-7**).



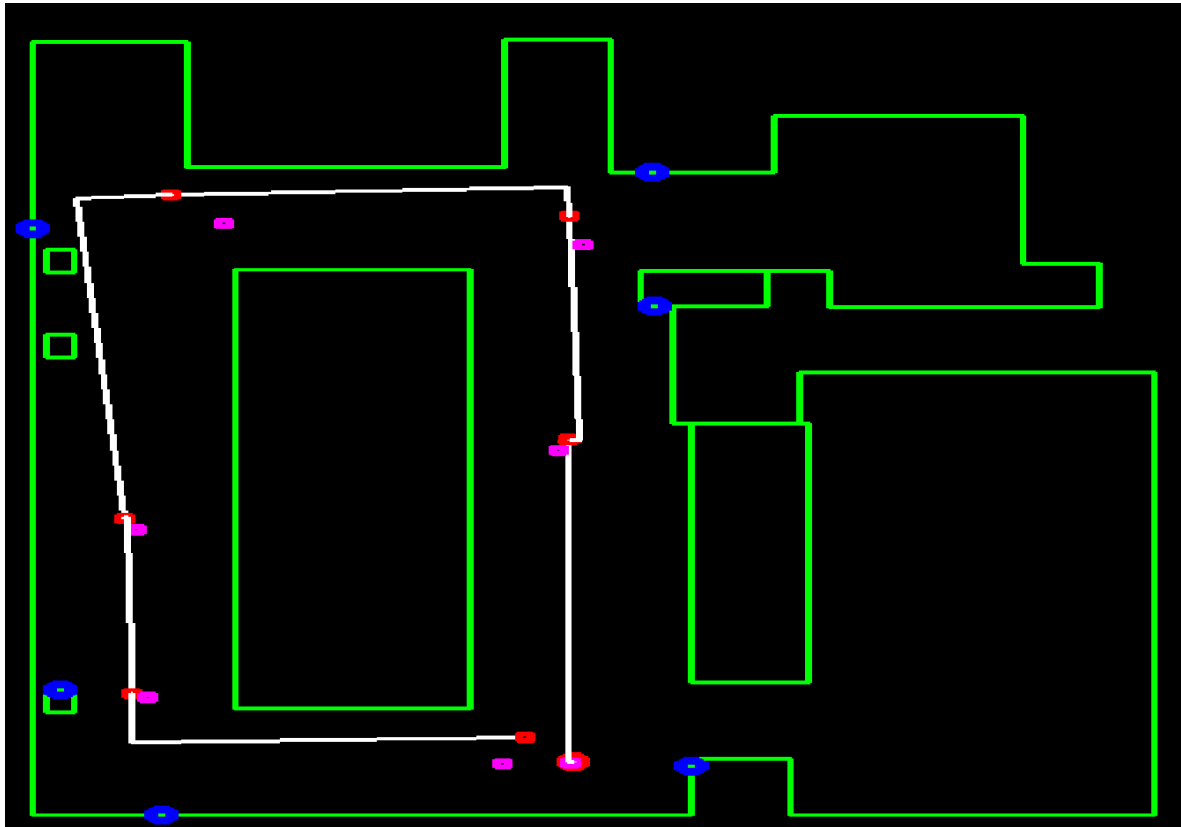
Rysunek 5-7. Schemat blokowy algorytmu przemieszczania się robota po otoczeniu

Proces przemieszczania się robota po otoczeniu laboratorium zakłada, że znaczniki występują w kolejności po sobie na trasie, jaką robot przemierza. Przykładowo, jeśli robot zidentyfikował swoją pozycję względem znacznika nr 2, to kolejnym znacznikiem na trasie robota będzie znacznik nr 3. Takie rozwiązanie upraszcza algorytm i przyspiesza jego działanie, gdyż wymaga wykonywania procesu identyfikacji tylko jednorazowo. Przemieszczanie odbywa się za pomocą instrukcji odometrycznych, odzwierciedlających

Podczas testów działania algorytmu skonfrontowano położenie robota uzyskiwane przez odometrię z wynikami korygowania pozycji przez kamerę. Rezultaty tego eksperymentu przedstawiają rysunki (**Rysunek 5-8**, *Rysunek 5-9*), na których została zobrazowana trasa robota uzyskana podczas dwóch różnych przejazdów, zaczynając od tego samego punktu. Białe linie obrazują drogę, jaka została wyznaczona na podstawie sensorów odometrycznych. W punktach oznaczonych czerwonymi elipsami, pozycja robota była weryfikowana przez system wizyjny. Otrzymaną przez kamerę skorygowaną pozycję przedstawiają różowe elipsy. Odległość pomiędzy parami czerwonych i różowych elips obrazuje błąd pomiędzy rzeczywistą pozycją, a tą wynikającą z odometrii.



Rysunek 5-8. Trasa robota wyznaczona przez odometrię z korekcją wizyjną pozycji - przejazd 1



Rysunek 5-9. Trasa robota wyznaczona przez odometrię z korekcją wizyjną pozycji - przejazd 2

Analizując obydwie trasy wyznaczone przez odometrię można zauważyć znaczne różnice, podczas gdy położenie odpowiadających sobie punktów wyznaczonych przy pomocy kamery na obydwu rysunkach różni się nieznacznie. Pokazuje to, że wizyjny algorytm korekcji położenia robota spełnia swoją rolę. Pomiar przejechanej odległości wyznaczonej przez odometrię jest pomiarem zawyżonym, co widać wyraźnie w pierwszej części pokonywanej trasy. W drugiej części trasy robota odległość pomiędzy pozycją wyznaczoną przez kamerę, a tą wyznaczoną na podstawie pomiaru odometrycznego wyraźnie się zmniejsza. Wynika to z kumulowania się błędu otrzymanego podczas jazdy w dwóch przeciwnych kierunkach. Oprócz błędu przejechanej odległości istotny jest również błąd wyznaczenia pozycji kątowej. Pomiar kąta podczas wykonywania pętli wymagającej nieustannego skrętu w lewo jest zawyżony, co jest zobrazowane przez coraz większe odchylenie toru jazdy od równoległego wobec ścian.

Podczas testów zmierzono również błąd odometrii jaki występuje po przejechaniu 4ech kolejnych okrążeń wokół laboratorium:

Tabela 5-1. Błędy odometryczne

	1 okrążenie	2 okrążenie	3 okrążenie	4 okrążenie
Błąd przesunięcia [m]	0,38	0,87	1,46	2,47
Błąd pozycji kątowej [°]	5,0	9,8	15,0	20,0

Z otrzymanych wyników wyraźnie widać, iż błędy odometrii kumulują się. Błąd pozycji kątowej rośnie liniowo dla każdego przejechanego okrążenia, podczas gdy błąd przesunięcia ma tendencję do wzrostu wykładniczego. Jest to spowodowane tym, że pozycja kątowa jest powiązana bezpośrednio z przesunięciem, zatem im większy błąd w określaniu pozycji kątowej, tym błąd przesunięcia rośnie szybciej. Regularność wzrostu zarówno błędu pozycji jak i orientacji kątowej, wskazuje iż w pomiarze odometrycznym dominują błędy systematyczne. Podczas testów pozycja robota była bardzo skutecznie korygowana przez kamerę, dzięki czemu robot przemierzał bezproblemowo 4 okrążenie po wyznaczonym torze. Przesunięcie pozycji robota o 2,47m i 20° biorąc pod uwagę wymiary trasy 5 i 9 metrów z pewnością spowodowałoby najechanie na przeszkodę, gdyby korzystano jedynie z pomiaru odometrycznego.

6. Podsumowanie

W części teoretycznej pracy opisano najbardziej popularne metody lokalizacji robotów mobilnych oraz omówiono tematykę wykorzystanych operacji przetwarzania obrazu.

Głównym zadaniem praktycznym pracy było zbudowanie systemu lokalizacji robota opartego na systemie wizyjnym wykorzystującym pojedynczą kamerę. W tym celu została zastosowana metoda sztucznych znaczników. Zakres części praktycznej można podzielić na stworzenie systemu wizyjnego oraz zaimplementowanie odpowiedniego sterowania dla robota mobilnego. Podstawą systemu wizyjnego był algorytm napisany w języku C++ korzystający z biblioteki do przetwarzania obrazu OpenCV, który należało udoskonalić.

W pierwotnej wersji algorytmu, wartość progu dla operacji segmentacji ustawiona była jako stała. Rzeczywiste środowisko pracy robota, wymaga by wartość progu zmieniała się w zależności od oświetlenia, kąta obrotu znacznika czy odległości. By zautomatyzować ten proces przetestowano w praktyce działanie czterech sposobów doboru progu za pomocą metod: iteracyjnej, progowania adaptacyjnego, doboru progu na podstawie wielkości maski oraz doboru progu za pomocą histogramu. Z rozpatrywanych możliwych rozwiązań najlepszym okazała się metoda doboru progu za pomocą histogramu. Jej użycie w ostatecznej wersji algorytmu, było podyktowane tym, iż łączy w sobie skuteczność z szybkością działania. Warto podkreślić, iż metoda iteracyjna okazała się metodą bardziej niezawodną, natomiast wymagała również większej złożoności obliczeniowej.

Kolejne wprowadzone usprawnienie algorytmu, to zastosowanie filtracji sceny wizyjnej z użyciem niezmienników momentowych. W pierwotnej wersji algorytm do tego celu używał operacji morfologicznej - zamknięcia. Wadą takiego podejścia było to, iż w przypadku znalezienia na obrazie obszarów, które przerastają rozmiarem obszar zajmowany przez znacznik, nie były one odfiltrowywane i istotnie ograniczały skuteczność działania systemu. W przypadku zastosowania filtracji z użyciem niezmienników momentowych wszystkie obszary nie klasyfikowane jako kwadraty są automatycznie odrzucane. Rozwiązanie to również posiada wady, gdyż w przypadku znalezienia na obrazie innych elementów o kolorze i kształcie odpowiadającym znacznikowi, algorytm może zostać wprowadzony w błąd.

Ostatnim istotnym rozbudowaniem systemu wizyjnego było wprowadzenie możliwości rozróżniania poszczególnych znaczników. W tym celu zbudowano zestaw ośmiu znaczników, identyfikowalnych za pomocą kodu binarnego oraz zmodyfikowano algorytm analizy obrazu.

Zadaniem drugiej części pracy praktycznej było zaimplementowanie systemu sterowania dla robota w rzeczywistych warunkach pracy. Taki system został stworzony w pomieszczeniu Laboratorium Teorii Sterowania znajdującego się w gmachu Starej Kotłowni na terenie Politechniki Warszawskiej. W charakterystycznych punktach pomieszczenia zostały umieszczone znaczniki, wspomagające nawigację robota. Następnie stworzono cyfrową mapę pomieszczenia obrazującą jego kształt oraz zawierającą informację o położeniu poszczególnych znaczników.

Istota sterowania robotem polegała na tym by robot znajdując się w danym punkcie pomieszczenia, mając w polu widzenia kamery określony znacznik był w stanie

zidentyfikować swoją pozycję, a następnie zaczął się poruszać po z góry określonej trasie. Stworzona wersja algorytmu sterowania korzysta z odczytów sensorów odometrycznych oraz z systemu wizyjnego. System wizyjny odpowiada za określenie wstępnego położenia robota, natomiast odometria sprzężona z silnikami napędowymi, umożliwia ruch w odpowiednim kierunku. Ponieważ pomiar odometryczny obciążony jest błędem, który kumuluje się w czasie, pozycja jest co pewien czas korygowana z użyciem kamery.

Przeprowadzone testy pokazały, że zaimplementowane sterowanie sprawdza się bardzo dobrze w praktyce. Błąd odometryczny jest na tyle duży, że bazując jedynie na jego odczycie wykonanie przez robota kilku instrukcji ruchu powoduje znaczącą zmianę toru jazdy, czego konsekwencją może być najechanie na przeszkodę. W wyniku zastosowania korekcji wizyjnej, robot może poruszać się po otoczeniu bez przerwy, nie zbaczając znacząco z wyznaczonej trasy. Trasę robota można śledzić ponadto na zbudowanej mapie.

Wykonany system umożliwia wprowadzenie dalszych udoskonaleń. Możliwe jest usprawnienie zarówno: algorytmu systemu wizyjnego, sterowania robotem jak i zastosowanie dodatkowych sensorów.

Jednym z udoskonaleń na przyszłość dla systemu wizyjnego mogłoby być zautomatyzowanie doboru parametrów wejściowych kamery. Obecnie parametry takie jak jasność, poziom kolorów, kontrast, korekcja gamma i ekspozycja obrazu są dobierane manualnie w zależności od panujących warunków oświetleniowych. Automatyzacja tego procesu usprawniłaby działanie całego systemu.

Inną możliwością jest rozszerzenie działania systemu o użycie sonarów, w które to robot Pioneer 3DX jest wyposażony. Aktualna wersja algorytmu sterowania robotem bazuje na założeniu, że w terenie nie występują żadne nieprzewidziane przeszkody. W przypadku pojawienia się takich, system nie jest w stanie w żaden sposób tego zweryfikować. Aby zapobiec kolizjom robota z pojawiającymi się obcymi obiektami, zbudowany algorytm sterowania można rozszerzyć o system omijania przeszkód, które byłyby wykrywane na podstawie odczytów z sonarów. Sonary mogłyby być ponadto użyte do aktualizowania w czasie rzeczywistym mapy otoczenia.

Istotną cechą oprogramowania OpenCV jest to, iż posiada możliwość obsługiwanie kilku kamer jednocześnie. Korzystając z tej opcji możliwe staje się zastosowanie dodatkowej kamery wąskokątnej o wysokiej rozdzielczości, umieszczonej na robocie. W takim przypadku zadaniem stosowanej obecnie szerokokątnej kamery, byłoby dokonywanie wstępnej lokalizacji znacznika w terenie. Następnie obszar zajmowany przez znacznik byłby powiększany przez kamerę wąskokątną i na tej podstawie wyznaczana faktyczna pozycja robota. Takowe rozwiązanie z pewnością zwiększyłoby zarówno maksymalny dystans lokalizacji, jak i precyzję pomiarów.

Podsumowując wykonaną pracę można stwierdzić, iż jej cel został osiągnięty – zaimplementowano w praktyce system lokalizacji robota mobilnego z użyciem kamery. Jak pokazały przeprowadzone testy jego dokładność jest wystarczająca dla rozwiązania problemu lokalizacji robota mobilnego w pomieszczeniu laboratorium, gdzie sprawdził się znakomicie. Należy podkreślić dodatkowo, iż stworzony podczas pracy system wizyjny jest uniwersalny. Istnieje możliwość zaadoptowania jego działania zarówno do różnych pomieszczeń jak i

typów robotów mobilnych. Możliwe jest wykorzystanie w nim różnych rodzajów kamer oraz współpraca z najrozmaitszymi sensorami i algorytmami sterowania. Oferując tak duże możliwości rozwoju stanowi, więc dobrą bazę dla kolejnych prac, które będą wykonywane w przyszłości w Zakładzie Sterowania Wydziału Elektrycznego Politechniki Warszawskiej.

Referencje

1. J. Borenstein, H. R. Everett, L. Feng. *"Where am I?". Sensors and Methods for Mobile Robots Positioning*. The University of Michigan 1996
2. <http://www.usno.navy.mil/USNO/time/gps/current-gps-constellation> (data dostępu: 29 Maj 2010)
3. Søren Riisgaard, Morten Rufus Blas. *SLAM for Dummies: A Tutorial Approach to Simultaneous Localization and Mapping*.
4. Niklas Karlsson, Luis Goncalves, Mario E. Munich, and Paolo Pirjanian. *The vSLAM Algorithm for Navigation in Natural Environments*. Evolution Robotics, Inc.
5. Pantelis Elinas, James J. Little. *Stereovision SLAM: Near real-time learning of 3D point-landmark and 2D occupancy-grid maps using particle filters*. University of British Columbia Laboratory for Computational Intelligence Vancouver, BC, Canada
6. Ryszard Tadeusiewicz, Przemysław Korohoda. *Komputerowa analiza i przetwarzanie obrazów*. Wydawnictwo Fundacji Postępu Telekomunikacji, Kraków 1997
7. Gary Bradski, Adrian Kaehler. *Learning OpenCV*. O'Reilly Media, Inc. 2008 USA
8. Mariusz Szablowski. *Analiza porównawcza wybranych bibliotek do przetwarzania obrazów*. Politechnika Warszawska. Wydział Elektryczny. Praca magisterska na kierunku elektrotechnika. Rok akademicki 2005/2006.
9. <http://pl.wikipedia.org/wiki/%C5%BByrooskop> (data dostępu: 29 Maj 2010)
10. Do-Eun Kim, Kyung-Hun Hwang, Dong-Hun Lee, Tae-Young Kuc. *A Simple Ultrasonic GPS System for Indoor Mobile Robot System using Kalman Filtering*. School of Information and Communication Engineering, SungKyunKwan University, Korea
11. <http://www.nec-tokin.com/english/product/piezodevice2/ceramicgyro.html> (data dostępu: 29 Maj 2010)
12. Paweł Buczyński. *Optymalna reprezentacja kolorów w analizie i przetwarzaniu obrazów komputerowych*.
13. http://www.vision.caltech.edu/bouguetj/calib_doc/ (data dostępu: 29 Maj 2010)
14. *Pioneer 3 Operations Manual*. Mobile Robots. ActivMedia Robotics 2006.
15. Marcin Iwanowski. *Wykład: Inteligentne maszyny i systemy*. Politechnika Warszawska 2008
16. Billur Barshan, Hugh F. Durrant-Whyte. *An Inertial Navigation System for a Mobile Robot*. International Conference on Intelligent Robots and Systems, Yokohama Japan July 26-30, 1993.

17. <http://www.dimensionengineering.com/accelerometers.htm> (data dostępu: 29 Maj 2010)

18. http://en.wikipedia.org/wiki/Bayer_filter (data dostępu: 29 Maj 2010)

19. Jakub Szymczak. *System wspomagania wyznaczania położenia robota mobilnego za pomocą czujników odometrycznych* Politechnika Warszawska. Wydział Elektryczny. Praca magisterska na kierunku elektrotechnika. Rok akademicki 2006/2007.

20. <http://sklep.kg.net.pl/files/sklep/imagecache/product/images/products/1/1805/pr-kamera-creative-live-cam-optia-af-1805-1.jpg> (data dostępu: 29 Maj 2010)

Spis Załączników

Do pracy załączono płytę CD zawierającą elektroniczną wersję pracy oraz stworzone na jej potrzeby oprogramowanie:

- Daniel_Chade_praca_magisterska.docx – plik z pracą magisterską w formacie docx
- Daniel_Chade_praca_magisterska.pdf – plik z pracą magisterską w formacie PDF
- System Wizyjny – katalog zawierający ostateczną wersję algorytmu systemu wizyjnego, bez sterowania dla robota
- System Sterowania - katalog zawierający ostateczną wersję algorytmu sterowania dla robota