

POLITECHNIKA WROCŁAWSKA
WYDZIAŁ ELEKTRONIKI

KIERUNEK: Automatyka i Robotyka
SPECJALNOŚĆ: Robotyka (ARR)

PRACA DYPLOMOWA
INŻYNIERSKA

Wykorzystanie systemu iRVision
w sterowaniu manipulatorami
przemysłowymi Fanuc

iRVision based Fanuc industrial
manipulators control

AUTOR:
Krzysztof Dąbek

PROWADZĄCY PRACĘ:
dr inż. Robert Muszyński

OCENA PRACY:

Spis treści

1	Wprowadzenie	7
2	System iRVision	9
2.1	Czym jest iRVision	9
2.2	Elementy składowe systemu iRVision	9
2.2.1	Sprzęt	9
2.2.2	Oprogramowanie	10
2.3	Typowe zastosowania systemu iRVision	10
3	Instalacja i konfiguracja iRVision	11
3.1	Połączenie z PC przez Ethernet	11
3.2	Dobór kamery, obiektywu i oświetlenia	13
3.3	Siatka kalibracyjna	16
3.4	Ustawienia kamery i obiektywu	16
3.5	Definiowanie narzędzia roboczego (TCP)	17
3.6	Definiowanie układu współrzędnych użytkownika	18
3.6.1	Ręcznie	18
3.6.2	Automatycznie	19
3.7	Kalibracja kamery	21
3.7.1	Przeprowadzenie kalibracji	21
3.7.2	Weryfikacja wyników kalibracji	22
3.8	Podstawowy proces wizyjny 2D	24
3.8.1	Opcje procesu wizyjnego	24
3.8.2	Uczenie modelu (GPM Locator)	26
3.8.3	Testowanie procesu	29
3.9	Podstawowe funkcje programowe iRVision	30
4	Przegląd możliwości systemu iRVision	31
4.1	Procesy wizyjne	31
4.2	Narzędzia wizyjne	32
4.2.1	Lokalizacja obiektów	32
4.2.2	Kontrola jakości	34
4.2.3	Narzędzia matematyczne	34
4.3	Zaawansowane funkcje programowe iRVision	35
5	Procedura Vision TCP	37
5.1	Przygotowanie stanowiska	37
5.1.1	Sprzęt	37
5.1.2	Oprogramowanie	37

5.2	Znacznik TCP	38
5.3	Kamera USB	39
5.4	iRCalibration	40
5.4.1	Ustawienia	40
5.4.2	Uruchamianie	42
6	System iRVision w laboratorium 010	45
6.1	Wyposażenie	45
6.2	Uwagi dotyczące korzystania	46
7	Przykładowa implementacja iRVision	49
7.1	Zawartość rejestrów robota	49
7.1.1	Rejestry ogólne	49
7.1.2	Rejestry pozycyjne	49
7.1.3	Rejestry wizyjne	50
7.1.4	Układy współrzędnych	50
7.2	Program testowy narzędzia lokalizacji	50
7.3	Program testowy narzędzia inspekcji	51
7.4	Podstawowy program iRVision	52
7.5	Aplikacja demonstracyjna	52
8	Podsumowanie	55
	Bibliografia	57

Spis rysunków

3.1	Okno konfiguracji TCP/IP	12
3.2	Strona startowa robota	12
3.3	Okno instalacji sterowników iRVision	13
3.4	Proponowane obiektywy na stanowisko w laboratorium 010	14
3.5	Metody oświetlania sceny [1]	15
3.6	Przykładowa siatka kalibracyjna	16
3.7	Okno konfiguracji wizji	16
3.8	Okna ustawień kamery	17
3.9	Położenie pokręteł na obiektywie firmy Ricoh [2]	18
3.10	Przypadki zastosowania automatycznej definicji układu współrzędnych [4]	19
3.11	Ekran ustawień automatycznej definicji układu współrzędnych	20
3.12	Okno kalibracji kamery – ustawienia	21
3.13	Okno kalibracji kamery – wynik kalibracji	23
3.14	Okno pojedynczego procesu wizyjnego 2D	25
5.1	Przykładowe znaczniki do kalibracji	38
5.2	Przykładowy adapter znacznika	38
5.3	Kamery USB dla robotów FANUC	39
5.4	Położenie portu USB na iPendancie	39
5.5	Podstawowe okna opcji Vision TCP Set	40
5.6	Okna konfiguracji funkcji Vision TCP Set	41
5.7	Okno uczenia modelu – Vision TCP Set	41
5.8	Okno parametrów wykrywania znacznika – Vision TCP Set	42
5.9	Przykładowy, wygenerowany program VTCP1	43
7.1	Kod źródłowy programu <i>IRV_T_LOCATOR_TEST</i>	50
7.2	Kod źródłowy programu <i>IRV_T_INSPECTION_TEST</i>	51
7.3	Kod źródłowy programu <i>IRV_T_BASIC</i>	52
7.4	Kod źródłowy programu <i>IRV_SWEEP_KLOCKI</i>	53

Spis tablic

5.1	Parametry wyznaczania osi X, Y TCP	43
5.2	Parametry wyznaczania osi Z TCP	44

Rozdział 1

Wprowadzenie

Wprowadzenie na rynek światowy robotów przemysłowych, a w szczególności manipulatorów o sześciu stopniach swobody zrewolucjonizowało procesy wytwórcze. Ta rewolucja została nazwana robotyzacją środowisk przemysłowych. W odróżnieniu od maszyn, przystosowanych do wykonywania zwykle jednego zadania, roboty charakteryzują się dużą wszechstronnością działania. Pracując na jednym stanowisku robot może zostać zaprogramowany do dowolnego zadania manipulacyjnego lub diagnostycznego, a po modyfikacji linii produkcyjnej przeprogramowany, by służyć innemu celowi. Możliwości ruchowe robotów są znacznie większe niż innych maszyn. Powoduje to znaczne zmniejszenie kosztów produkcji i modyfikacji linii produkcyjnej oraz skrócenie czasu potrzebnego na wykonanie zadania.

Mimo ich wszechstronności, możliwości manipulatorów są wciąż ograniczone. Problemem większości systemów zrobotyzowanych jest konieczność dużego uporządkowania środowiska. Przenoszone przez roboty i obrabiane przez maszyny obiekty muszą znaleźć się zawsze dokładnie w tym samym miejscu, gdy rozpoczyna się proces. Takie rozwiązania są kosztowne, zwłaszcza biorąc pod uwagę koszty pomyłek operatorów oraz błędów maszyn podczas produkcji. Zaletą wykorzystania systemów wizyjnych jest to, że umożliwiają pracę robota w środowisku, które nie jest uporządkowane. Obrabiany lub przenoszony element może znajdować się w dowolnym położeniu i orientacji w przestrzeni roboczej lub przesuwać się na pasie transportowym.

W laboratorium 010 w budynku C-3 Politechniki Wrocławskiej znajdują się trzy manipulatory przemysłowe, z których dwa zostały stworzone przez japońską firmę FANUC. FANUC jest jednym z wiodących producentów robotów do zastosowań w przemyśle, specjalizującym się w manipulatorach. Oba roboty wyposażone są w system wizyjny iRVision, który odpowiednio skonfigurowany, znacznie zwiększa możliwości robota.

Niniejsza praca została napisana w celu zaprezentowania możliwości i ograniczeń tego systemu oraz przedstawienia czytelnikowi sposobu korzystania z niego. Opis iRVision ograniczono do wybranych, najczęściej wykorzystywanych części systemu. Uwzględniono odniesienia do odpowiednich rozdziałów dokumentacji. Zgodnie z założeniami praca ta ma służyć jako poradnik w czasie pracy z systemem iRVision w podstawowym zakresie. Praca została także wzbogacona o opis ponadpodstawowych procesów oraz przykłady ich implementacji.

Układ pracy jest następujący. W drugim rozdziale opisano ogólnie system wizyjny iRVision i jego elementy składowe. Rozdział trzeci zawiera opis sposobu instalacji i konfiguracji systemu iRVision. Czwarty rozdział przedstawia możliwości

systemu i udostępnionego interfejsu użytkownika. W rozdziale piątym zawarto opis procedury ułatwiającej konfigurację podstawowych układów współrzędnych robota. W szóstym rozdziale wymieniono elementy wyposażenia 010 związane z iRVision oraz uwagi dotyczące pracy w laboratorium. W siódmym rozdziale przedstawiono przykładowe aplikacje testowe i demonstracyjne. Ostatni rozdział podsumowuje całość.

Rozdział 2

System iRVision

2.1 Czym jest iRVision

iRVision jest systemem wizyjnym przygotowanym przez firmę FANUC, stosowanym w manipulatorach przemysłowych [4]. Stanowi on zestaw rozwiązań sprzętowych i programowych implementowanych przez firmę FANUC głównie na potrzeby przemysłu. iRVision zwiększa uniwersalność zastosowania robotów w aplikacjach przemysłowych na produkcji, w magazynach i innych częściach fabryk. Zadań manipulatorów w przemyśle jest wiele. Najczęstsze aplikacje to przenoszenie elementów, spawanie, malowanie. Dzięki systemom wizyjnym proces automatyzacji oraz robotyzacji realizowanych zadań jest znacznie ułatwiony. iRVision zawiera funkcje umożliwiające wykonanie większości zadań stawianych obecnie systemom wizyjnym.

2.2 Elementy składowe systemu iRVision

Niżej wymienione elementy są niezbędne do implementacji systemu wizyjnego a następnie pracy z nim. Część z nich jest dostarczana wraz z systemem przez firmę FANUC, część operator robota lub firma, w której pracuje zapewnia we własnym zakresie.

2.2.1 Sprzęt

W skład wyposażenia systemu wizyjnego firmy FANUC wchodzi:

- kamera przemysłowa ze złączem 12-pinowym,
- obiektyw do kamery,
- oświetlenie,
- kabel 12-żyłowy do podłączenia kamery,
- kabel Ethernet z wtykami RJ-45,
- siatka kalibracyjna.

2.2.2 Oprogramowanie

W zestawie oprogramowania systemu iRVision znajdują się:

- programy do kalibracji kamery w przestrzeni roboczej,
- oprogramowanie akwizycji i przetwarzania obrazów,
- oprogramowanie współpracy robota z systemem wizyjnym,
- procesy wizyjne,
- narzędzia wizyjne,
- oprogramowanie iRCalibration do kalibracji układów współrzędnych robota z użyciem wizji,
- interfejs użytkownika.

2.3 Typowe zastosowania systemu iRVision

System wizyjny iRVision znajduje zastosowanie w manipulatorach firmy FANUC w niemal wszystkich gałęziach przemysłu. W każdej jego gałęzi, w której wykorzystuje się roboty, wizja ułatwia ich pracę i rozszerza możliwości. Główne zastosowania iRVision to:

Lokalizacja obiektów: Najbardziej podstawowe zastosowanie wizji, związane bezpośrednio z częścią oprogramowania, obliczającą pozycję obiektów w różnych układach współrzędnych. Robot jest w stanie zlokalizować obiekt, a następnie przeprowadzić na nim operacje. Najczęściej jest to zadanie *pick and place* lub spawanie (zgrzewanie) elementów. Innym zadaniem może być nałożenie na obiekt kleju lub innej substancji. Wszystkie zadania sprowadzają się do dojazdu robota do określonego punktu na obiekcie lub śledzenia jego krawędzi.

Kontrola jakości: Alternatywnie obraz z kamery może zostać wykorzystany do kontroli jakości obiektu. Podczas zautomatyzowanego procesu produkcyjnego, mogą pojawić się błędy w wykonaniu elementów. System wizyjny ma możliwość wykrycia błędów produkcyjnych zarówno na krawędziach obiektu, jak i na jego powierzchni. W efekcie, robot może przykładowo przenieść wadliwy element do odpowiedniego kontenera. Dokładność dokonywanej przez system kontroli jakości zależy od kalibracji, oświetlenia oraz ustawień procesu wizyjnego. Kontrola jakości zwykle jest dodatkową funkcjonalnością robota, wykorzystującą wizję, realizowane wraz z zadaniem lokalizacji.

Depaletyzacja: Paletyzacja jest procesem, który nie wymaga zastosowania systemu wizyjnego. Proces odwrotny, czyli depaletyzacja, współdziała z kamerą. W zadaniu paletyzacji obiekty są układane w znany wzór i łatwo zapamiętać, który element został już położony na palecie. Podczas depaletyzacji, system wizyjny jest w stanie określić ile elementów (warstw), ułożonych pionowo znajduje się na palecie i wyznaczyć na jakiej wysokości zacząć operację depaletyzacji. Wizję można także wykorzystać do lokalizacji palety.

Rozdział 3

Instalacja i konfiguracja iRVision

Ten rozdział poświęcony jest dobór sprzętu, instalacja, konfiguracja systemu iRVision oraz sposób przygotowania prostej aplikacji, wykorzystującej wizję. W poniższych podrozdziałach szczegółowo opisano kolejne kroki, które są konieczne do rozpoczęcia wygodnej pracy z wizją. Dokładne wyjaśnienia i rady dotyczące konfiguracji pozwolą uniknąć błędów podczas pracy z systemem.

3.1 Połączenie z PC przez Ethernet

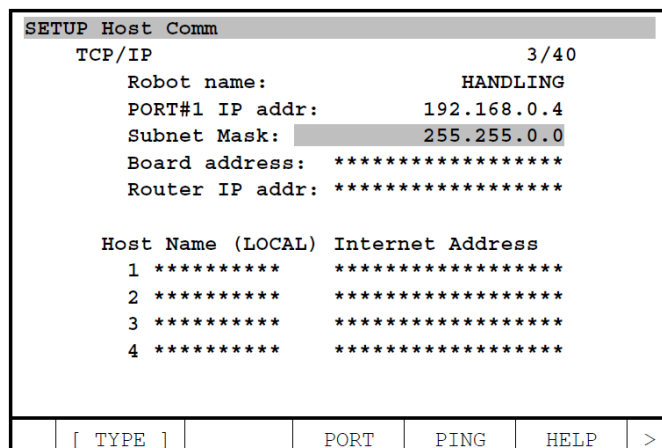
Praca z systemem iRVision jest możliwa z poziomu „wędki” robota (*ang. Teach Pendant lub iPendant*). Jednakże przeprowadzenie jego kalibracji i konfiguracji jest wygodniejsze z poziomu komputera. Musi być to komputer PC z zainstalowanym systemem Windows oraz przeglądarką internetową *Microsoft Internet Explorer 7* lub nowszą. Aby rozpocząć pracę z systemem za pośrednictwem komputera, należy wykonać poniższe kroki:

1. Połączyć kartę sieciową komputera z portem CD38A lub CD38B w kontrolerze za pomocą skrętki Ethernetowej z wtyczkami RJ-45. Nie ma znaczenia czy kabel jest prosty, czy krosowany (*ang. Cross-Over*). Jednakże z uwagi na większą przepustowość zaleca się stosowanie kabli krosowanych.
2. Przejść do konfiguracji IP portu w kontrolerze dla połączenia z użyciem protokołu TCP, wybierając opcję:

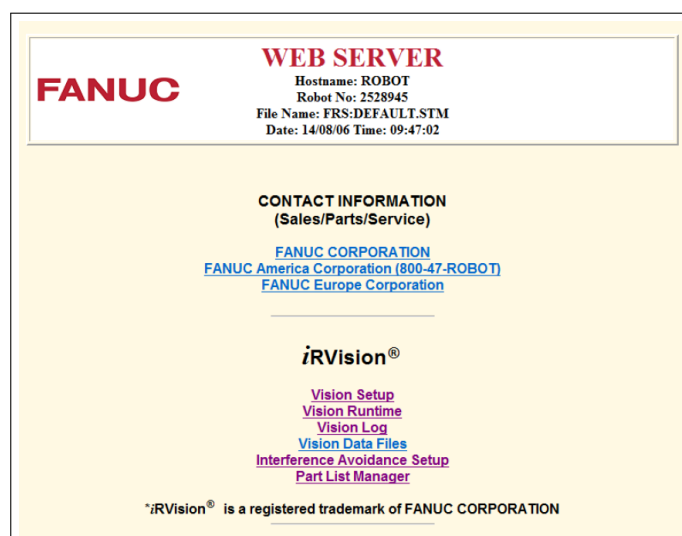
- *MENU* → *SETUP* → *Host Comm* → *TCP/IP*

Okno konfiguracji przedstawiono na rysunku [3.1](#).

3. Wciskając *F3 – PORT* przejść do ustawień portu, do którego został podłączony kabel Ethernet (CD38A – Port 1, CD38B – Port 2).
4. Wpisać adres sieciowy portu (*ang. PORT IP addr*) oraz maskę podsieci (*ang. Subnet Mask*). Powinien to być adres lokalny (np. adres klasy C: 192.168.0.3 z maską 255.255.255.0). Reszta pól nie musi być wypełniona.
5. W ustawieniach sieci komputera zmienić adres sieci na adres z tej samej podsieci, co port w kontrolerze (np. 192.168.0.2 z maską 255.255.255.0).



Rysunek 3.1: Okno konfiguracji TCP/IP



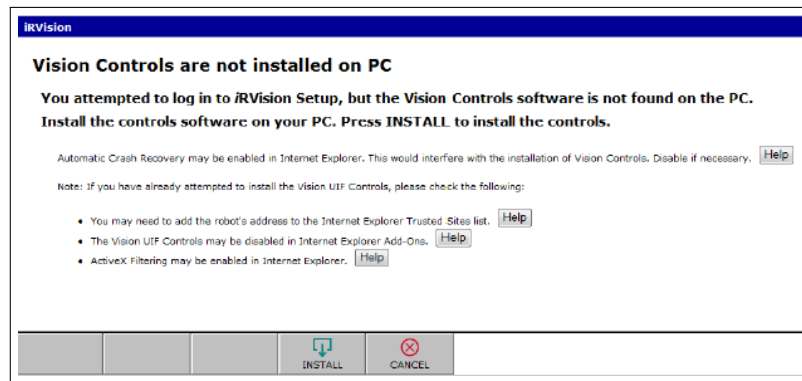
Rysunek 3.2: Strona startowa robota

6. Na komputerze uruchomić przeglądarkę *Internet Explorer* i w pasku adresu wpisać adres portu na kontrolerze (np. 192.168.0.3). Pojawi się strona startowa robota (rysunek 3.2).
7. Wybrać iRVision Setup. Wyświetli się komunikat z prośbą o zainstalowanie sterowników systemu wizyjnego (jeśli nie zostały wcześniej zainstalowane) (rysunek 3.3). Należy zainstalować sterowniki, żeby umożliwić pracę z systemem.

Praca z robotem jest także możliwa zdalnie, po podłączeniu go kablem Ethernet do routera sieci lokalnej. Wtedy z każdego komputera w sieci, można bez problemu połączyć się z robotem, wpisując jego adres sieciowy, nawet przez połączenie bezprzewodowe. Możliwe jest także podłączenie kilku robotów do jednego komputera przez przełącznik (*ang. switch*).

Dokładny opis konfiguracji połączenia oraz posługiwania się wizją przez Ethernet znajduje się w [4] (dodatek A).

Dokładny opis posługiwania się robotem przez Ethernet można znaleźć w [3] (rozdział 9.17).



Rysunek 3.3: Okno instalacji sterowników iRVision

3.2 Dobór kamery, obiektywu i oświetlenia

Podstawą systemu wizyjnego jest kamera, która przechwytyje obrazy i umożliwia dalszą pracę na ich podstawie. System wizyjny nie może także obyć się bez obiektywu, który określa dystans i obszar roboczy kamery. Podobnie jedną z najważniejszych części całego systemu jest dobre oświetlenie.

Kamera: Kamerę należy wybrać z listy kamer kompatybilnych ze sterownikami robotów firmy FANUC. Lista została przedstawiona poniżej:

- Analogowe
 - Sony XC-56
 - Sony XC-HR50
 - Sony XC-HR57
- Cyfrowe
 - Kowa SC130C
 - Kowa SC130E
 - Kowa SC310M
- USB
 - Kowa SC36MF
 - Basler acA640-20um

Wszystkie kamery dają podobne wyniki, najczęstszym i rekomendowanym wyborem jest kamera Sony XC-56.

Obiektyw: Obiektyw w aplikacjach iRVision jest absolutnie konieczny, gdyż kamery kompatybilne z robotami firmy FANUC nie mają wbudowanych. Daje to możliwość wyboru obiektywu odpowiedniego do aplikacji. Wybrany model powinien mieć możliwość ręcznego ustawienia przesłony (*ang. iris control*) oraz ostrości (*ang. focus control*). Przy wyborze obiektywu trzeba zwrócić szczególną uwagę na:

- wielkość matrycy sensora optycznego. Jeśli obiektyw będzie przystosowany do matryc mniejszych niż wykorzystywana, światło nie będzie docierać do całej powierzchni matrycy. W laboratorium 010 kamera wyposażona jest w sensor typu 1/3".



(a) Kowa LM6NCL [6]



(b) Kowa LM8JC [7]

Rysunek 3.4: Proponowane obiektywy na stanowisko w laboratorium 010

- ogniskową (*ang. focal length*). Ten parametr wpływa bezpośrednio na kąt widzenia kamery. Im większa ogniskowa, tym lepszy wynik kalibracji można uzyskać, jednak jeśli będzie zbyt duża, znacznie ograniczy pole widzenia kamery. Z uwagi na to, że w laboratorium 010 kamera znajduje się na ramieniu robota typu *LR Mate*, zaleca się zastosowanie obiektywu o ogniskowej nie większej niż 8mm.
- zakres ostrości (*ang. focusing range*). Zakres ten ma zwykle tylko dolne ograniczenie, które warunkuje jak blisko może znajdować się obiekt. W laboratorium 010 sposób umieszczenia robota ogranicza odległość z jakiej może zostać zrobione zdjęcie do 450 mm.
- sposób montażu (*ang. mount*). Żeby uniknąć stosowania przejściówek lub adapterów, należy dobrać odpowiedni sposób mocowania obiektywu. W przypadku kamer dostosowanych do iRVision stosowane jest mocowanie typu C-Mount.

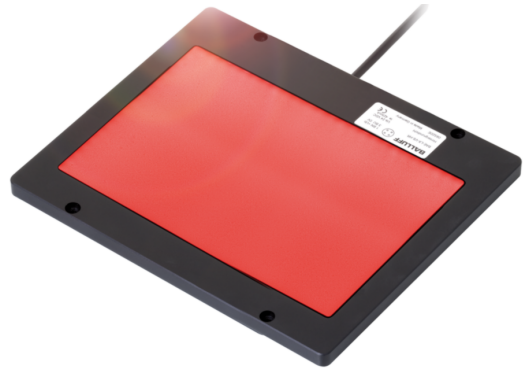
Firmy produkujące obiektywy to między innymi: Ricoh, Tamron, Kowa. Na stanowisku obecnie do kamery zamontowano obiektyw o zmiennej ogniskowej 10 – 40 mm. Do dalszej, wygodniejszej pracy z systemem iRVision zaleca się jego wymianę na obiektyw o ogniskowej 6 mm lub 8 mm. Propozycje pokazano na rysunku 3.4.

Oświetlenie: Tylko dobrze oświetlony obiekt może być poprawnie zidentyfikowany na obrazie. Dobierając oświetlenie sceny należy kierować się poniższymi zasadami:

- Oświetlenie powinno być niezmiennie w czasie. Ponadto identyczne podczas kalibracji i konfiguracji procesu wizyjnego.
- Światło może mieć dowolną barwę. Najlepsze rezultaty uzyskuje się, gdy światło ma barwę kontrastową do koloru obiektu wykrywanego lub białą.



(a) Oświetlacz pierścieniowy



(b) Podświetlenie



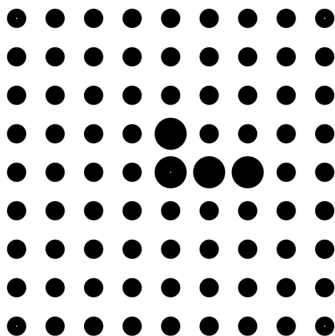
(c) Oświetlacz stojący

Rysunek 3.5: Metody oświetlania sceny [1]

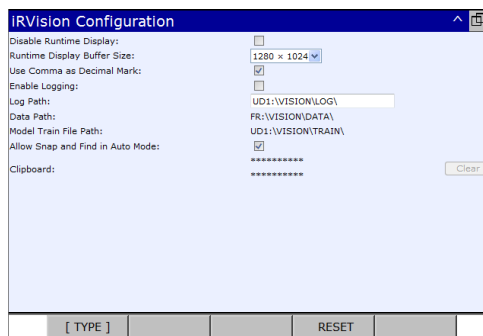
- Wiązka światła powinna być rozproszona, by uniknąć refleksów na dobrze odbijających (np. polerowanych) powierzchniach.
- Konieczna jest możliwie najlepsza redukcja cieni obiektów na całej scenie lub aktualnie branej pod uwagę jej części.

Oświetlenie musi być dobrane do aplikacji. Najbardziej efektywnymi, testowanymi w kilku aplikacjach metodami oświetlania sceny są:

1. oświetlacz pierścieniowy wokół kamery (rysunek 3.5a),
2. podświetlenie (*ang. backlight*) (rysunek 3.5b),
3. statyczny oświetlacz przemysłowy halogenowy lub LED (3.5c).



Rysunek 3.6: Przykładowa siatka kalibracyjna



Rysunek 3.7: Okno konfiguracji wizji

3.3 Siatka kalibracyjna

Do procesu konfiguracji oraz kalibracji kamery konieczna jest siatka kalibracyjna, przedstawiona na rysunku 3.6. Rozstaw kół powinien być dobrany do pola widzenia kamery, tak żeby było widoczne od 6 do 10 rzędów i kolumn. W laboratorium 010 została przygotowana siatka o rozstawie 30 mm wydrukowana na papierze.

Dokładne wytyczne, określające wygląd siatki, rozstaw oraz wielkość kół znajdują się w [4] (rozdział 11.1).

W celu dokonania ustawień i kalibracji iRVision siatkę kalibracyjną należy ułożyć w przestrzeni roboczej, w odległości roboczej od kamery, zgodnie z osiami układu współrzędnych, którego chcemy użyć w naszej aplikacji. Trzy duże koła wyznaczają oś X, dwa duże koła oś Y. UWAGA: Siatki nie można poruszyć podczas całego procesu kalibracji!

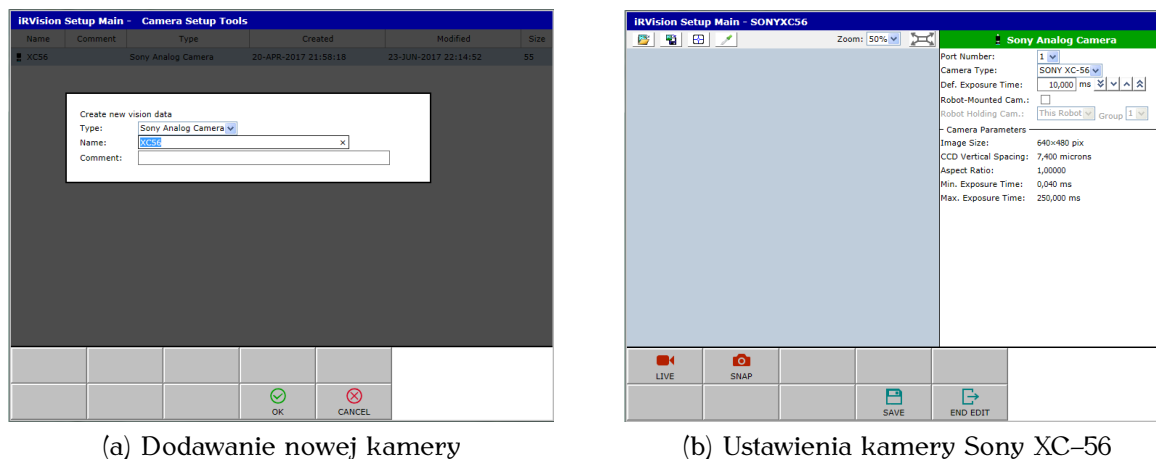
3.4 Ustawienia kamery i obiektywu

Informacje o kamerze i obiektywie muszą zostać wprowadzone do ustawień systemu. Aby rozpocząć ustawianie kamery należy wybrać opcję

- *MENU* → *iRVision* → *Vision Config*

Pojawi się ekran konfiguracji wizji (rysunek 3.7). Do aplikacji w laboratorium 010 zaleca się pozostawienie domyślnych opcji. Dokładny opis wszystkich opcji w menu *Vision Config* znajduje się w [4] (rozdział 3.4).

Następnie należy przejść do menu



Rysunek 3.8: Okna ustawień kamery

- *MENU* → *iRVision* → *Vision Setup*

i dodać nową kamerę (rysunek 3.8a). W laboratorium 010 jest to Sony Analog Camera XC-56. Nazwa kamery może być dowolna. Dokładny opis menu *Vision Setup* znajduje się w dokumentacji (rozdział 3.1).

Wybierając pole *EDIT* lub wciskając klawisz *Enter* przejść do ustawień dodanej kamery. Na ekranie ustawień (rysunek 3.8b) wybrać typ kamery (*ang. Camera Type*). W opisywanym przypadku będzie to SONY XC-56. Zaznaczyć, czy kamera jest zamontowana na robocie (*ang. Robot-Mounted Camera*). Kolejno ustawić czas naświetlania, jednocześnie dopasowując przesłoneę i ogniskową obiektywu kamery. Następujące działania należy wykonać ze szczególną uwagą, gdyż wpływają na jakość całej kalibracji.

1. Ustawić kamerę (ramię robota) w pozycji robienia zdjęcia.
2. Ustawić czas naświetlania (*ang. Exposure Time*) na 10 ms.
3. Przesłoneę, pokrętle najbliższej gwintu obiektywu (rysunek 3.9), ustawić tak, żeby tło siatki było jasno szare. Białe lub zbyt ciemne tło może pogorszyć wynik kalibracji.
4. Drugim pokrętle (rysunek 3.9) ustawić ogniskową tak, aby obraz siatki kalibracyjnej był wyraźny, a kropki nie były ani powiększone, ani pomniejszone.

Gdy przesłona i ogniskowa są ustawione, można przejść do procesu kalibracji.

3.5 Definiowanie narzędzia roboczego (TCP)

Przed skalibrowaniem kamery należy zająć się definicją narzędzia roboczego (*ang. Tool Frame, User Tool*). Jest to podstawowa operacja wykonywana na robotach firmy FANUC, polegająca na kalibracji chwytaka, która może być wykonana kilkoma metodami:

- metoda trójpunktowa (*ang. Three-Point Method*) – najprostsza metoda kalibracji. Zwraca przesunięcie narzędzia względem flanszy robota,



Rysunek 3.9: Położenie pokręteł na obiektywie firmy Ricoh [2]

- metoda czteropunktowa (*ang. Four-Point Method*) – zwraca przesunięcie narzędzia oraz zwrot osi Z,
- metoda sześciopunktowa (*ang. Six-Point Method*) – zwraca przesunięcie i orientację narzędzia względem flanszy robota,
- ręczne wprowadzenie danych (*ang. Direct Entry*) – korzystać tylko, jeśli metoda montażu narzędzia daje dobrą dokładność. Przesunięcie i obrót chwytaka wprowadzone zostaje przez użytkownika,
- iRCalibration – Vision TCP (zobacz rozdział 5) – wykorzystanie wizji do kalibracji narzędzia.

Dokładny opis czterech pierwszych procedur kalibracji znajduje się w [3] (rozdział 3.9.1).

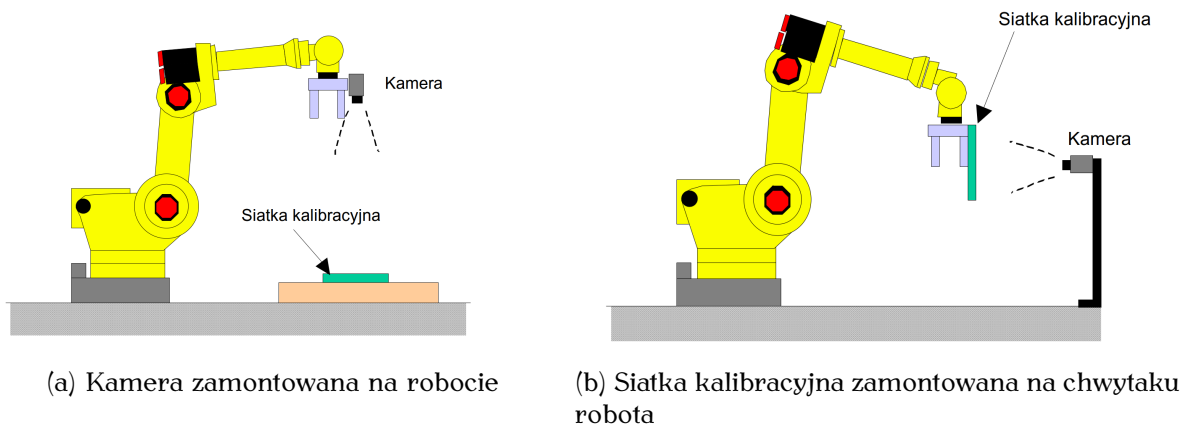
3.6 Definiowanie układu współrzędnych użytkownika

Po zdefiniowaniu narzędzia roboczego, można przejść do definicji układu współrzędnych użytkownika (*ang. User Frame*). W aplikacjach wykorzystujących wizję jest on nazywany także układem siatki kalibracyjnej (*ang. Calibration Grid Frame*). Względem niego obliczana będzie pozycja nauczonych elementów oraz offset dla punktów w programie wykorzystującym iRVision (zobacz [4] rozdział 11). Są dwie metody definiowania układu współrzędnych użytkownika: automatyczna i ręczna.

3.6.1 Ręcznie

Ręczne definiowanie układu współrzędnych użytkownika nie różni się od definiowania układu współrzędnych do celów innych niż aplikacje, wykorzystujące wizję. Jest to podstawowa operacja wykonywana na robotach firmy FANUC. Układ współrzędnych użytkownika można zdefiniować kilkoma metodami:

- Metoda trójpunktowa (*ang. Three-Point Method*) – Polega na dotknięciu końcówką narzędzia roboczego trzech punktów na siatce kalibracyjnej (Początek układu, kierunek osi X, kierunek osi Y).



Rysunek 3.10: Przypadki zastosowania automatycznej definicji układu współrzędnych [4]

- Metoda czteropunktowa (*ang. Four-Point Method*) – Polega na dotknięciu końcówką narzędzia roboczego czterech punktów na siatce kalibracyjnej (Początek siatki, kierunek osi X, kierunek osi Z, nowy początek układu).
- Ręczne wprowadzenie danych (*ang. Direct Entry*) – Umożliwia użytkownikowi ręczne wpisanie przesunięcia i orientacji nowego układu współrzędnych, względem światowego układu współrzędnych.

Dokładny opis ręcznych metod definiowania układu współrzędnych użytkownika znajduje się w [3] (rozdział 3.9.2).

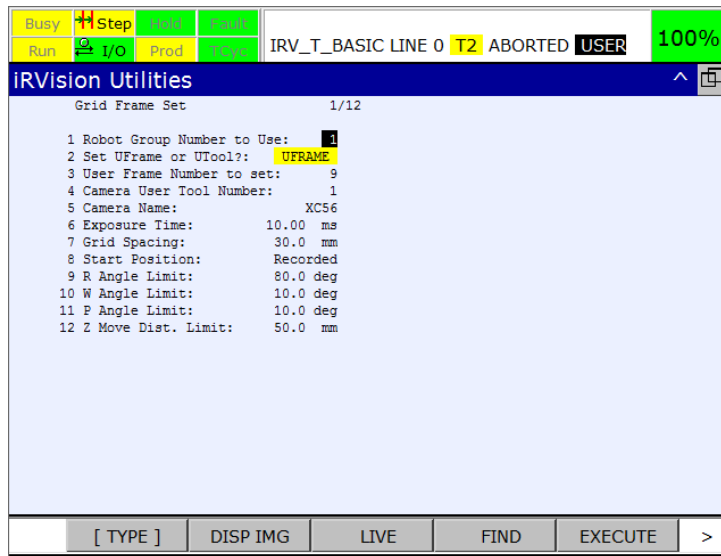
3.6.2 Automatycznie

Automatyczna kalibracja układu współrzędnych użytkownika jest opcją instalowaną na kontrolerach FANUC wraz z modułem iRVision. Pozwala skalibrować układ współrzędnych użytkownika (np. dla wizji) w przypadku, gdy kamera lub siatka kalibracyjna jest umieszczona na robocie (rysunek 3.10). Aby przejść do automatycznej kalibracji, należy ustawić robota w pozycji robienia zdjęcia (rozdział 3.4) a następnie wybrać opcję:

- MENU → iRVision → Utilities → Automatic Grid Frame Set

Ukaże się ekran automatycznej definicji układu współrzędnych (rysunek 3.11), w którym należy wybrać opcje odpowiednie dla naszej aplikacji.

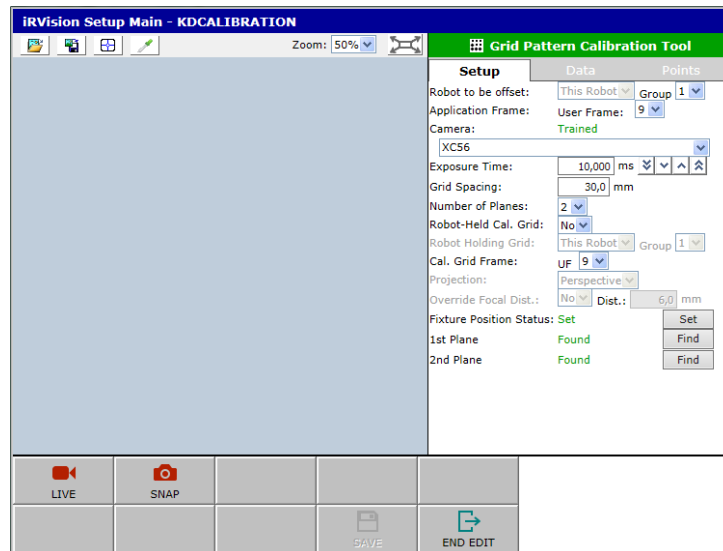
- **Robot Group Number to Use:** Nazwa grupy robota używanej podczas kalibracji. Pozostawić domyślne: 1.
- **Set UFrame or UTool?:** Możliwość wybrania czy ustawiamy układ współrzędnych użytkownika czy kamery. Wybrać: UFRAME.
- **User Frame Number to set:** Numer układu współrzędnych użytkownika do ustawienia. Wybrać odpowiedni.
- **Camera User Tool Number:** Numer układu współrzędnych kamery do ustawienia. Wybrać pusty.



Rysunek 3.11: Ekran ustawień automatycznej definicji układu współrzędnych

- **Camera Name:** Nazwa kamery wykorzystywanej w procesie. Wybrać wcześniej dodaną kamerę (rozdział 3.4).
- **Exposure Time:** Czas ekspozycji, wpływa na oświetlenie siatki. Ustawić: 10 ms.
- **Grid Spacing:** Rozstaw kropek na siatce kalibracyjnej. Ustawić rzeczywisty rozstaw kropek: 15 mm.
- **Start Position:** Pozycja startowa procesu definiowania układu. Wcisnąc *SHIFT* nagrać pozycję robienia zdjęcia za pomocą *F4* – *Record*. Optymalna pozycja jest zapisana w rejestrze PR[91: irv_refpos].
 UWAGA: Sprawdzić czy robot na pewno jest w pozycji robienia zdjęcia podczas nagrywania pozycji.
- **R Angle Limit:** Maksymalny możliwy kąt obrotu wokół osi optycznej kamery. Zwiększenie, daje większą dokładność kalibracji. Ustawić: 80.0 deg.
- **W Angle Limit:** Maksymalny możliwy kąt wychylenia robota wokół osi X prostopadłej do osi kamery. Zwiększenie, daje większą dokładność kalibracji. Ustawić: 10.0 deg.
- **P Angle Limit:** Maksymalny możliwy kąt wychylenia robota wokół osi Y prostopadłej do osi kamery. Zwiększenie, daje większą dokładność kalibracji. Ustawić: 10.0 deg.
- **Z Move Distance Limit:** Odległość w globalnej osi z na jaką robot może się poruszyć względem pozycji początkowej. Zwiększenie, daje większą dokładność kalibracji. Ustawić: 50 mm.

Następnie należy wcisnąć *DEADMAN SWITCH* oraz *SHIFT* i uruchomić proces definiowania przyciskiem *F4*. Robot zacznie poruszać kamerą (lub siatką kalibracyjną), robiąc zdjęcia siatki z różnych pozycji i orientacji. Podczas procesu można zwiększyć szybkość poruszania się robota, żeby proces wykonać szybciej. Jeśli proces zostanie przerwany z powodu błędu, można wznowić go przyciskiem *RESUME*



Rysunek 3.12: Okno kalibracji kamery – ustawienia

lub wystartować ponownie przyciskiem *RESTART*. Jeśli proces zostanie przerwany z powodu osiągnięcia limitu pozycji, naruszenia strefy DCS lub przez użytkownika w sytuacji niebezpiecznej, należy zidentyfikować zagrożenie, zmniejszyć odpowiednią współrzędną i wznowić proces przyciskiem *RESUME*.

UWAGA: Cały czas należy kontrolować, czy robot nie wywoła niebezpiecznej sytuacji, która może skutkować zagrożeniem zdrowia lub uszkodzeniem sprzętu! w razie potrzeby, zmniejszyć prędkość poruszania się robota.

Dokładny opis procesu kalibracji znajduje się w [4] (rozdział 10.2).

3.7 Kalibracja kamery

3.7.1 Przeprowadzenie kalibracji

Siatka kalibracyjna powinna być umieszczona dokładnie w tym samym miejscu, co podczas definiowania układu współrzędnych użytkownika. Aby rozpocząć kalibrację kamery, należy wybrać opcje:

- *MENU* → *iRVision* → *Vision Setup*
- *F4* – *VTYPE* → *Camera Calibration Tools*

Przyciskiem *F2* – *CREATE* dodać nową kalibrację kamery, wybierając opcję *Grid Pattern Calibration* i nadając jej własną nazwę. Wciskając *Enter* przejść do okna kalibracji (pokazanego na rysunku 3.12).

Należy ustawić kolejno wszystkie opcje kalibracji kamery. Opis opcji kalibracji przedstawiono poniżej:

Robot to be offset: Robot na którym przeprowadzona zostanie kalibracja. Wybrać aktualnie używanego robota (*ang. This Robot*) oraz grupę (*ang. Group*) 1.

Application Frame: Układ współrzędnych użytkownika (*ang. User Frame*), w którym będzie napisana aplikacja. Wybrać wcześniej zdefiniowany układ.

Camera: Kamera używana do kalibracji. Wybrać wcześniej zdefiniowaną i ustawioną kamerę (Sony XC-56).

Exposure Time: Czas ekspozycji. Wybrać taki sam jak podczas ustawiania kamery.
Zalecany: 10 ms.

Grid Spacing: Rozstaw kół siatki kalibracyjnej. Wpisać 30 mm.

Number of Planes: Liczba płaszczyzn używanych do kalibracji. Do kalibracji z kamerą umieszczoną na robocie lub siatką kalibracyjną trzymaną przez robota należy wybrać liczbę płaszczyzn: 2.

Robot-Held Cal. Grid: Czy siatka kalibracyjna jest trzymana przez robota. Wybrać *Nie*.

Robot Holding Grid: Robot trzymający siatkę kalibracyjną. Opcja aktywna tylko gdy w poprzedniej wybrano *Tak*.

Cal. Grid Frame: Układ współrzędnych używany do kalibracji. Wybrać ten sam, co *Application Frame*.

Projection: Projekcja. Zalecana opcja to projekcja perspektywiczna (*ang. Perspective*). Do projekcji ortogonalnej (*ang. Orthogonal*) jest wykorzystywana soczewka telecentryczna i nie może być użyta dla dwóch płaszczyzn kalibracji.

Override Focal Dist.: Ręczne ustawienie ogniskowej obiektywu. Jeśli ogniskowa obiektywu jest źle wyliczona przez proces kalibracji, należy wybrać *Tak* i wpisać prawidłową wartość ogniskowej obiektywu.

Fixture Position Status: Położenie siatki kalibracyjnej, zgodne ze zdefiniowanym układem współrzędnych aplikacji. Wcisnąć *Set* aby system obliczył i zapisał położenie.

1st Plane: Poszukiwanie punktów siatki kalibracyjnej na pierwszej płaszczyźnie. Ustawić robota w pozycji robienia zdjęcia (pierwsza płaszczyzna). Następnie wcisnąć *Find* i zaznaczyć obszar na obrazie, w którym mają być wykrywane punkty. W miarę możliwości ograniczyć obszar do jak najmniej zniekształconego przez soczewkę.

2nd Plane: Poszukiwanie punktów siatki kalibracyjnej na drugiej płaszczyźnie. Ustawić robota około 150 mm niżej niż podczas szukania na pierwszej płaszczyźnie. Wcisnąć *Find* i dostosować obszar wykrywania punktów siatki.

Po wykryciu punktów siatki na obu płaszczyznach pojawi się komunikat, że kalibracja została zakończona a punkty przeliczone (*ang. Points have been recalculated*).

3.7.2 Weryfikacja wyników kalibracji

Należy sprawdzić czy wyniki kalibracji są zgodne z oczekiwaniami. Sprawdzenie jest dwuetapowe:

Sprawdzenie punktów siatki kalibracyjnej: W zakładce *Points* okna kalibracji (rysunek 3.13a) wykonać następujące czynności:

Grid Pattern Calibration Tool							
Setup		Data			Points		
Plane #:	2						
#	Vt	Hx	X	Y	Z	Err	
1	240,0	319,9	-0,0	0,0	150,0	0,049	▲
2	239,4	380,7	30,0	0,0	150,0	0,029	
3	179,3	319,2	-0,0	30,0	150,0	0,021	
4	238,7	441,4	60,0	0,0	150,0	0,045	
5	56,4	439,5	60,0	90,0	150,0	0,012	
6	57,8	318,0	-0,0	90,0	150,0	0,038	
7	59,0	196,4	-60,0	90,0	150,0	0,055	
8	58,4	257,2	-30,0	90,0	150,0	0,045	
9	60,3	75,0	-120,0	90,0	150,0	0,095	
10	59,7	135,7	-90,0	90,0	150,0	0,045	
11	61,0	14,2	-150,0	90,0	150,0	0,164	
12	115,2	622,4	150,0	60,0	150,0	0,039	
13	115,9	561,7	120,0	60,0	150,0	0,068	
14	117,1	440,1	60,0	60,0	150,0	0,046	▼
15	116,6	500,9	90,0	60,0	150,0	0,060	
Point Number:			Delete				

(a) Punkty siatki

Grid Pattern Calibration Tool							
Setup		Data			Points		
Focal Distance:	5,993 mm						
Lens Distortion:	0,00000						
Magnification:	0,612 mm/pix						
Image Center:	(239,5, 319,5) pix						
CCD Vertical Spacing:	7,40 microns						
Aspect Ratio:	1,00001						
Mean error value:	0,060 pix						
Maximum error value:	0,166 pix						
Position of Camera Relative to Cal. Grid							
X:	0,1	Y:	0,0	Z:	543,8		
W:	0,0	P:	-0,0	R:	89,4		
Position of Cal. Grid Relative to App. UFrame							
X:	0,0	Y:	0,0	Z:	0,0		
W:	-0,0	P:	-0,0	R:	-0,0		
Position of Robot Holding Camera							
X:	-1,0	Y:	-84,3	Z:	592,4		
W:	-180,0	P:	-0,0	R:	89,4		

(b) Wyliczone dane

Rysunek 3.13: Okno kalibracji kamery – wynik kalibracji

1. Sprawdzić, czy niebieskie krzyżyki znajdują się wewnątrz dużych kropek.
2. Sprawdzić, czy zielone krzyżyki znajdują się wewnątrz małych kropek.
3. Znaleźć punkty o największym błędzie wykrycia (*ang. Err*). Jeśli błąd jest zbyt duży, usunąć punkt wpisując jego numer # w polu *Port Number* oraz wciskając *Delete*.

Sprawdzenie wyliczonych danych: W zakładce *Data* okna kalibracji (rysunek 3.13b) sprawdzić następujące wartości:

1. Ogniskową (*ang. Focal Distance*) – czy zgadza się z ogniskową obiektywu.
2. Dystorsję (*ang. Lens Distortion*) – czy zgadza się z dokumentacją obiektywu.
3. Powiększenie (*ang. Magnification*) – czy zgadza się z dokumentacją obiektywu.
4. Wartość średniej błędów (*ang. Mean error value*) – Powinna być mniejsza niż 0.5 piksela.
5. Wartość maksymalnego błędu (*ang. Maximum error value*) – Powinna być mniejsza niż 1 piksel.
6. Pozycję kamery względem siatki (*ang. Position of Camera Relative to Cal. Grid*) – Jeśli rzeczywista pozycja może być określona.
7. Pozycję siatki względem układu współrzędnych aplikacji (*ang. Position of Cal. Grid Relative to App. UFrame*) – Jeśli rzeczywista pozycja może być określona.

Jeśli któraś z wyliczonych wartości jest niezadowolająca, należy jeszcze raz przeprowadzić kalibrację.

Przed wyjściem z okna kalibracji wcisnąć *Next* oraz *F5 – SAVE*, żeby zapisać kalibrację.

Dokładny opis procesu kalibracji na podstawie siatki kalibracyjnej znajduje się w dokumentacji iRVision [4] (rozdział 5.1). Można tam także znaleźć opis procesu automatycznej rekalkibracji (rozdział 5.1.4), a także opis innych narzędzi do kalibracji (rozdziały 5.2, 5.3, 5.4).

3.8 Podstawowy proces wizyjny 2D

Najbardziej podstawowym procesem wizyjnym iRVision jest Pojedynczy Proces 2D (*ang. 2D Single View Vision Process*), korzystający między innymi z podstawowego narzędzia lokalizacji na podstawie konturów (*ang. GPM Locator Tool*). Zarówno dla procesu jak i narzędzia muszą zostać ustawione odpowiednie opcje. Całość uczenia musi zostać przetestowana przed tworzeniem aplikacji.

Obiekt, który ma być lokalizowany w przestrzeni roboczej, należy ustawić tak, aby na zdjęciu z pozycji, z której przeprowadzana była kalibracja, znajdował się mniej więcej na środku, z rotacją bliską 0.

3.8.1 Opcje procesu wizyjnego

Aby dodać nowy proces wizyjny należy wybrać opcje:

- *MENU* → *iRVision* → *Vision Setup*
- *F4 – VTYPE* → *Vision Process Tools*.

Następnie przyciskiem *F2 – CREATE* stworzyć nowy proces wizyjny, wybierając *2D Single View Vision Process* i nadając mu własną nazwę. Po wciśnięciu *Enter* pojawi się okno procesu wizyjnego, przedstawione na rysunku 3.14a.

Kolejno należy ustawić odpowiednie opcje procesu wizyjnego. Opcje zostały opisane poniżej.

Camera Calibration: Nazwa przeprowadzonej kalibracji kamery. Wybrać nazwę nadaną właściwej kalibracji w rozdziale 3.7.

Camera Setup: Nazwa kamery zdefiniowanej w ustawieniach kamery. Ustawić taką jak w rozdziale 3.4.

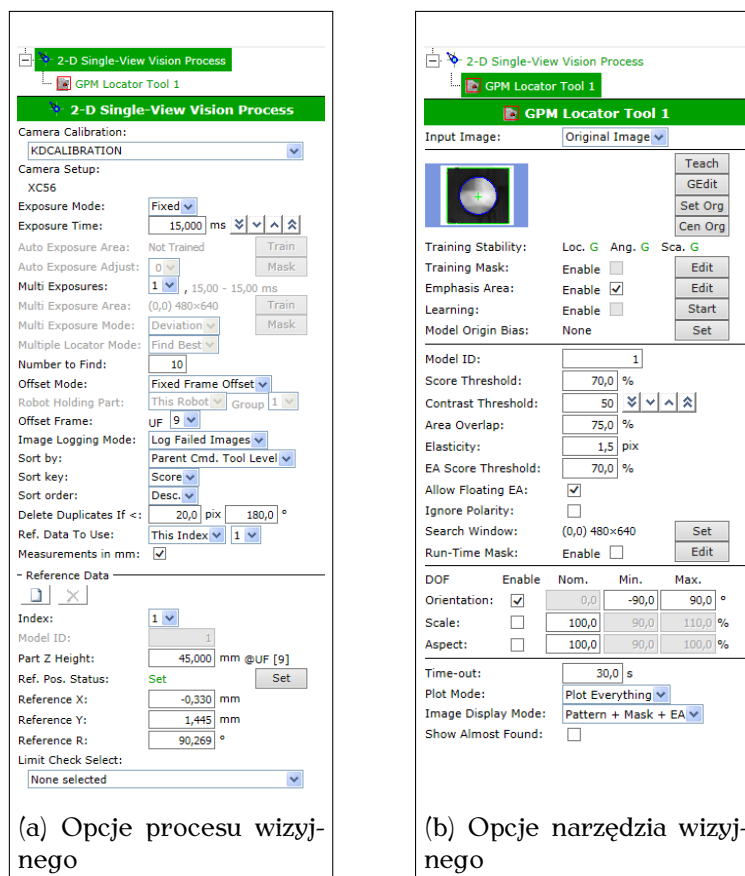
Exposure Mode: Tryb ekspozycji. Może być stały (*ang. Fixed*) lub automatyczny (*ang. Auto*). Wybrać stały.

Exposure Time: Czas ekspozycji. Ustawić taki jak podczas kalibracji lub nieco większy, jeśli obiekt nie jest dobrze widoczny (np. 15 ms).

Auto Exposure Area, Adjust: Opcje używane dla automatycznego trybu ekspozycji.

Multi Exposures: Wielokrotna ekspozycja. Może poprawić widoczność elementu bez zmiany czasu ekspozycji, zwiększa czas robienia zdjęcia. Zalecane wybranie 1.

Multi Exposure Area, Mode: Opcje używane, gdy *Multi Exposures* ma wartość większą niż 1.



Rysunek 3.14: Okno pojedynczego procesu wizyjnego 2D

Number to Find: Liczba obiektów do znalezienia. Maksymalnie 100. Jeśli liczba obiektów na obrazie jest większa niż *Number To Find*, za znalezione zostaną uznane obiekty z największym wynikiem (*Score*).

Offset Mode: Typ układu współrzędnych, względem którego ma zostać wyliczony offset. Możliwe opcje to przemieszczenie układu współrzędnych użytkownika (*ang. Fixed Frame Offset*), narzędzia (*ang. Tool Offset*) oraz brak przemieszczenia, za przemieszczenie wstawiana jest pozycja znalezionej elementu (*ang. Found Position*). Ustawić *Fixed Frame Offset*.

Robot Holding Part: Wybór robota trzymającego obiekt. Opcja dostępna, gdy wybrano *Offset Mode: Tool Offset*.

Offset Frame: Numer układu współrzędnych, względem którego wyliczany będzie offset (*User Frame*). Wybrać układ współrzędnych aplikacji taki jak w rozdziale 3.7.

Image Logging Mode: Tryb przechowywania obrazów w pamięci. Wybrać opcję niezapisywania obrazów (*ang. Do Not Log*).

Sort by: Tryb sortowania znalezionych elementów. Pozostawić domyślny (*Parent Cmd. Tool Level*).

Sort key: Klucz, po którym wyniki będą sortowane. Ustawić Wynik (*ang. Score*), czyli jak bardzo znaleziony element jest zgodny z nauczonym.

Sort order: Porządek sortowania. Można wybrać rosnąco (*ang. Ascending*) lub malejąco (*ang. Descending*). Wybrać *Desc*.

Delete Duplicates If: Próg sprawdzenia, czy dany obiekt został znaleziony wielokrotnie. Zwielokrotnienia są usuwane. Pozostawić domyślne (*20.0 pix, 180.0°*).

Ref. Data To Use: Wybór danych referencyjnych od których będzie wyliczany offset. Pozostawić domyślne (*This Index: 1*).

Measurements in mm: Pomiar i obliczenia wykonywane i zapisywane w rejestrze w milimetrach. Zaznaczyć.

Reference Data: Dane referencyjne, ustawiane po nauczaniu elementu. Strony danych referencyjnych mogą być usuwane i tworzone, a następnie wypełniane dla każdego nauczonego modelu obiektu osobno.

Index: Ustawić taki sam jak *Ref. Data To Use: 1*.

Model ID: Opcja wykorzystywana, gdy proces korzysta z kilku narzędzi wizyjnych, do wykrywania wielu modeli. Ustawić 1.

Part Z Height: Wysokość w osi Z szukanego elementu. Ustawić właściwą wysokość obiektów (*45 mm*).

UWAGA: Źle ustawiona wysokość, spowoduje błędne wyliczanie offsetu!

Ref. Pos. Status: Pozycja referencyjna w osiach X, Y oraz orientacja obiektu wokół osi Z. Po nauczaniu modelu obiektu wcisnąć *F3 – SNAP*, *F4 – FIND* oraz na ekranie *Set*. Pozycja może zostać wpisana ręcznie.

Limit Check Select: Limit offsetu. Jeśli obliczony offset jest zbyt duży, obiekt zostaje odrzucony. Ustawić brak limitu (*ang. None selected*). Więcej informacji w [4] (rozdział 8.2).

Dokładny opis procesu wizyjnego, automatycznej rekaliibracji i innych opcji znajduje się w [4] (rozdział 6.1). Przed wyjściem z okna procesu wcisnąć *Next* oraz *F5 – SAVE*, żeby zapisać ustawienia procesu wizyjnego.

Skrócona instrukcja do ustawiania opcji procesu wizyjnego znajduje się na rysunku 3.14a.

3.8.2 Uczenie modelu (GPM Locator)

Aby przejść do narzędzia wizyjnego GPM Locator, należy wybrać w oknie procesu wizyjnego (rysunek 3.14a) zakładkę *TREE* i z drzewa narzędzi wizyjnych *GPM Locator Tool*. Ukaze się okno przedstawione na rysunku 3.14b.

Wykonać poniższe czynności, żeby nauczyć robota modelu obiektu:

1. Określić obraz wykorzystywany do lokalizacji obiektu.

Input Image: Wybrać odpowiednią opcję. Może to być obraz aktualnie przechwycony przez proces wizyjny, obraz przetworzony przez narzędzia wizyjne (filtry, balans oświetlenia, narzędzia arytmetyczne itp.) lub obraz z pliku w pamięci kontrolera. Wybrać aktualny obraz (*ang. Original Image*).

2. Wykonać zdjęcie wciskając *F3 – SNAP*.

3. Wskazać obiekt na obrazie i stworzyć jego model.

- Wybrać na ekranie **Teach** oraz ograniczyć obszar na obrazie, do uczonego obiektu.
UWAGA: Należy zostawić kilka do kilkunastu pikseli marginesu wokół obiektu!
- Jeśli model wymaga dodania kształtów, które nie zostały uchwycone automatycznie, można to zrobić wciskając **GEdit** i dodając kształty ręcznie.
- Jeśli środek modelu został błędnie wykryty, można powtórzyć automatyczne centrowanie za pomocą **Cen Org – Center Origin** lub ustawić centrum modelu ręcznie **Set Org – Set Origin**.
- Stabilność wykrywania można sprawdzić w wynikach **Training Stability**, które oznaczają kolejno **Loc.** lokalizacja (*ang. Localization*), **Ang.** kąt (*ang. Angle*), **Sca.** skala (*ang. Scale*). Wynik może przyjmować trzy wartości: **Good**, **Poor**, **None**, które kolejno oznaczają stabilność dobrą, słabą i nie do zdefiniowania.
- Na obraz można nałożyć maskę (*ang. Training Mask*). Wewnątrz obszarów maski, krawędzie nie będą wyszukiwane.
- Istnieje możliwość zdefiniowania obszarów wyróżnionych (*ang. Emphasis Area*). Mają one większą wagę podczas obliczania Wyniku wyszukiwania obiektu (*ang. Score*).
- Funkcja **Model Origin Bias** umożliwia zaprzestanie wykrywania elementu na rzecz wykrycia przez inne zdefiniowane narzędzie do wykrywania. Jeśli tylko jedno narzędzie jest zdefiniowane w procesie, opcja jest nieaktywna.

UWAGA: Dokładny opis tworzenia modelu obiektu wraz ze wszystkimi możliwościami jego edycji znajduje się w dokumentacji (manual 7.1.6).

4. Ustawić opcje narzędzia wizyjnego. Opis opcji przedstawiono poniżej.

Model ID: Identyfikator modelu. Ustawić taki jak dla danych referencyjnych w procesie wizyjnym (1).

Score Threshold: Próg wykrywania. Minimalny *Wynik*, dla którego wykryty obiekt może być uznany za zgodny z modelem. Ustawić wedle uznania (zalecane 70,0%).

Contrast Threshold: Próg kontrastu. Minimalny kontrast, dla którego obraz jest przetwarzany w celu znalezienia obiektu. Dobrać minimalny, ale wystarczająco duży, żeby nie zostały znalezione zabrudzenia i inne nieporządane obiekty na obrazie (zalecane 50).

Area Overlap: Jeśli znalezione obiekty pokrywają się w części większej niż określona, tylko obiekt o wyższym *Wyniku* będzie uznany za wykryty. Ustawienie 100% powoduje, że wszystkie obiekty zostają wykryte mimo pokrywania się ich modeli.

Elasticity: Maksymalne odchylenie (w pikselach) znalezionej krawędzi od krawędzi modelu. Ustawić większe jeśli krawędzie wykrywanych obiektów są nierówne (zalecane 2 piksele).

- EA Score Threshold:** Próg wykrywania dla *Obszarów wyróżnionych*, definiowany osobno pozwala kontrolować wagę tych obszarów.
- Allow Floating EA:** Pozwala na przesunięcie *Obszarów wyróżnionych* o 2–3 piksele względem modelu.
- Ignore Polarity:** Ignorowanie polarności. Zaznaczenie spowoduje, że obiekt będzie wykryty także w przypadku odwróconych kolorów na obrazie. Nie zaznaczać.
- Search Window:** Obszar wyszukiwania. Obszar na obrazie, w którym system będzie szukał obiektów. Może zostać ograniczony, co przyspiesza obliczenia oraz zmniejsza wpływ efektu rybiego oka obiektywu. Zalecany cały obraz.
- Run-Time Mask:** Maskowanie obszaru wyszukiwania. Można na tej podstawie zdefiniować część obszaru wyszukiwania, która nie będzie przeszukiwana.
- DOF:** Minimalna i maksymalna różnica między modelem a znalezionym obiektem, dla których obiekt jest uznawany za zgodny z modelem.
- Orientation:** Możliwa rotacja wokół osi Z. Jeśli robot, niezależnie od sytuacji, nie powinien obracać narzędzia, żeby chwycić obiekt o kąt większy niż określony, ograniczyć *DOF – Orientation*. Np. jeśli obiekt jest prostokątem, po obrocie o więcej niż 90° można uznać, że jest obrócony o więcej niż -90° , ustawić $(-90^\circ, +90^\circ)$. Dla kwadratu będzie to $(-45^\circ, +45^\circ)$.
- Scale:** Możliwa zmiana wielkości obiektu na obrazie. Jeśli obiekt może być widziany jako większy lub mniejszy, niż na modelu, trzeba to uwzględnić. Dla aplikacji w laboratorium wyłączyć tę opcję.
- Aspect:** Możliwa różnica w wymiarach obiektu względem modelu. Wyłączyć tę opcję.
- Time-out:** Określenie maksymalnego czasu na poszukiwanie obiektów. Jeśli wyszukiwanie trwa dłużej, proces kończy się. Pozostawić domyślne.
- Result Plotting Mode:** Tryb wyświetlania wyników. Wybrać między wyświetlaniem krawędzi modelu, wielkości obszaru modelu, punktów środkowych modelu. Można wyświetlić wszystkie lub żadną z tych opcji.
- Image Display mode:** Tryb wyświetlania obrazu. Wybrać między wyświetlaniem obrazu, wyników, krawędzi obiektu, modelu oraz maski i obszarów wyróżnionych.
- Show Almost Found:** W wynikach pokazuje także obiekty, które nie zostały uznane za zgodne z modelem, jednak są podobne. Zaznaczyć wedle uznania.

Dokładny opis narzędzia wizyjnego *GPM Locator*, procesu uczenia modelu na podstawie wielu zdjęć i innych opcji znajduje się w [4] (rozdział 7.1). Przed wyjściem z okna narzędzia wcisnąć *Next* oraz *F5 – SAVE*, żeby zapisać ustawienia narzędzia wizyjnego.

Skrócona instrukcja do ustawiania opcji narzędzia wizyjnego znajduje się na rysunku 3.14b.

3.8.3 Testowanie procesu

Aby przystąpić do testów ustawionego procesu 2D i kalibracji kamery, należy: wrócić do okna procesu 2D (rysunek 3.14a), rozwinąć na ekranie listę **Results** oraz napisać krótką aplikację testową (rozdział 7.2) i aplikację podstawową (rozdział 7.4). Testy należy wykonać w kilku etapach. W każdym etapie należy sprawdzić w **Results** wartości: *Score*, *X*, *Y* i *Angle* oraz uruchomić program testowy i sprawdzić wartości w rejestrach `PR[]` i `VR[].offset`:

1. Obiekt w pozycji referencyjnej

- Sprawdzić, czy Wynik (*Score*) wynosi co najmniej 99,8% oraz czy Pozycja i Rotacja są równe danym referencyjnym.
- Sprawdzić, czy offset obliczony w pogramie testowym oraz offset w rejestrze wizyjnym są wyzerowane lub bardzo blisko zera.
- Uruchomić podstawową aplikację iRVision (rozdział 7.4) i sprawdzić czy robot dojeżdża do punktu z offsetem tak samo jak bez niego.

2. Obiekt przesunięty w osi X (Y) o znaną odległość

- Sprawdzić, czy Wynik (*Score*) wynosi co najmniej 70,0%, Rotacja oraz pozycja w osi Y (X) są bliskie danym referencyjnym oraz pozycja w osi X (Y) jest pozycją referencyjną powiększoną o znaną odległość.
- Sprawdzić, czy offset obliczony w pogramie testowym oraz offset w rejestrze wizyjnym mają wszystkie składowe blisko 0 oprócz składowej X (Y), która powinna wynosić w przybliżeniu znaną odległość.
- Uruchomić podstawową aplikację iRVision (rozdział 7.4) i sprawdzić czy robot dojeżdża do punktu, tak jak w pozycji referencyjnej.

3. Obiekt obrócony wokół osi Z o znany kąt

- Sprawdzić, czy Wynik (*Score*) wynosi co najmniej 70,0%, Pozycja w osiach X, Y są bliskie danym referencyjnym oraz rotacja *Angle* jest rotacją referencyjną powiększoną o znany kąt.
- Sprawdzić, czy offset obliczony w pogramie testowym oraz offset w rejestrze wizyjnym mają wszystkie składowe blisko 0 oprócz składowej R, która powinna wynosić w przybliżeniu znany kąt.
- Uruchomić podstawową aplikację iRVision (rozdział 7.4) i sprawdzić czy robot dojeżdża do punktu tak jak w pozycji referencyjnej.

4. Przesunięcie w osiach X, Y oraz obrót wokół osi Z

- Sprawdzić, czy dane o znalezionej pozycji i wynik spełniają oczekiwania.
- Sprawdzić, czy przemieszczenie wyliczone dwiema metodami jest prawidłowy.
- Uruchomić podstawową aplikację iRVision (rozdział 7.4) i sprawdzić czy robot dojeżdża do punktu, tak jak w pozycji referencyjnej.

3.9 Podstawowe funkcje programowe iRVision

Aby używać iRVision w programach na kontrolerze robota, należy skorzystać z dostępnych funkcji programowych. Aby dodać linię z funkcją iRVision należy w oknie edycji programu wybrać opcję: INST → iRVision.

Pojawi się lista funkcji. Najważniejsze z nich to:

- VISION RUN_FIND – Funkcja uruchamia proces wizyjny którego nazwa podana jest jako parametr. Przed wpisaniem tego polecenia należy umieścić robota w pozycji robienia zdjęcia.
- VISION GET_OFFSET – Funkcja zapisuje offset wyliczony z ostatnio uruchomionego procesu do rejestru wizyjnego $VR[a]$, gdzie a , to numer rejestru, podany w parametrze. Jeśli obiekt nie został znaleziony lub pojawił się inny błąd, program przejdzie do etykiety (*LABEL*) przekazanej jako parametr.
- VOFFSET,VR – Funkcję dopisuje się do polecenia ruchu robota, dodaje ona do współrzędnych punktu offset z rejestru wizyjnego $VR[a]$.

Przykładowy, podstawowy program wykorzystujący wizję pokazany jest w rozdziale 7.4.

Rozdział 4

Przegląd możliwości systemu iRVision

4.1 Procesy wizyjne

Proces wizyjny to program systemu iRVision, który wykorzystuje, narzędzia wizyjne do zadań: lokalizacji, kontroli jakości, depaletyzacji, obliczeń. Proces może być uruchomiony w programie robota przez funkcję *VISION RUN*. Najbardziej przydatne procesy wizyjne zostały opisane poniżej.

Pojedynczy proces 2D (*ang. 2D Single View Vision Process*) jest oparty na pojedynczym zdjęciu zrobionym przez kamerę. Na podstawie zdjęcia oraz modelu elementu, określonym przez wybrane narzędzie wizyjne, po ustawieniu odpowiednich opcji wylicza przemieszczenie (*ang. offset*): przesunięcie w osiach X, Y oraz obrót wokół osi Z elementu względem pozycji referencyjnej. Może wykorzystywać dane z różnych narzędzi wizyjnych jednocześnie. Używany jest głównie do lokalizacji obiektów w przestrzeni roboczej lub zliczania ich. Proces ten został opisany w rozdziale 3.8.1.

Opis w dokumentacji: [4] (rozdział 6.1).

Wielokrotny proces 2D (*ang. 2D Multi-View Vision Process*) może wykorzystywać od dwóch do czterech zdjęć (*ang. Camera View*) jednego elementu. Podobnie jak pojedynczy proces 2D służy do obliczenia przemieszczenia (*ang. offset*) elementu względem pozycji referencyjnej. Przemieszczenie jest obliczane na podstawie kilku zdjęć. Proces służy głównie do lokalizacji obiektów zbyt dużych by zmieściły się na pojedynczym zdjęciu. Do każdego zdjęcia można przyporządkować osobne narzędzie lokalizacyjne.

Opis w dokumentacji: [4] (rozdział 6.2).

Depaletyzacja (*ang. Depalletizing Vision Process*) to proces odwrotny do procesu paletyzacji. Polega na rozładowaniu obiektów z palety lub innego kontenera. Aby to umożliwić, system wizyjny w tym procesie oprócz przemieszczenia (*ang. offset*): przesunięcia w osiach X, Y oraz obrotu wokół osi Z, oblicza przesunięcie elementu w osi Z. Jest ono wyliczone na podstawie zmiany wielkości obiektu na obrazie, z uwzględnieniem perspektywy.

Opis w dokumentacji: [4] (rozdział 6.3).

Proces 2D bez kalibracji: (*ang. Calibration-Free Vision Process*), w odróżnieniu od pozostałych procesów, nie wymaga zdefiniowania układu współrzędnych

użytkownika, kalibracji kamery ani innych ustawień. Przemieszczenie (*ang. offset*) jest obliczane na podstawie pozycji robota nauczonej podczas konfiguracji procesu, zamiast na podstawie pozycji referencyjnej obiektu w układzie współrzędnych użytkownika (*ang. User Frame*). Proces zajmuje więcej czasu niż pojedynczy proces 2D, gdyż wymaga ruchu robota.

Opis w dokumentacji: [4] (rozdział 6.4).

Śledzenie trajektorii (*ang. Single View Visual Tracking*) śledzi przedmiot poruszający się na taśmie transportowej. System wyznacza przewidywaną pozycję, w której znajdzie się przedmiot po przemieszczeniu narzędzia roboczego w jego pobliże i rozpoczyna ruch do tego punktu. Trajektoria jest śledzona dla z góry określonego pasa transportowego.

Opis w dokumentacji: [4] (rozdział 6.10).

Kontrola jakości (*ang. Single View Inspection Vision Process*) określa na podstawie wykorzystywanych narzędzi wizyjnych: narzędzia kontroli powierzchni (*ang. Surface Flaw Inspection Tool*) oraz narzędzia kontroli krawędzi (*ang. Bead Inspection Tool*), czy dany obiekt spełnia warunki testu lub testów jakości. Możliwe jest użycie narzędzia ewaluacji (*ang. Evaluation Tool*), które tworzy test jakości z kilku narzędzi do kontroli jakości oraz narzędzi lokalizujących.

Opis w dokumentacji: [4] (rozdział 6.12).

Operacje na punktach obrazu (*ang. Image To Points Vision Process*) znajduje na obrazie punkty krawędzi obiektu. Danymi wyjściowymi są linie przecinające środek soczewki oraz dany punkt. Punkty można zlokalizować w przestrzeni na przecięciu tych linii z płaszczyzną powierzchni obiektu. Proces wykorzystuje narzędzie lokalizacji punktów obrazu (*ang. Edge Points Locator Tool*). Znajdowane oraz zliczane są także krawędzie określone jako łańcuchy punktów (*ang. chains*). Proces może zostać użyty do generowania trajektorii dla narzędzia robota.

Opis w dokumentacji: [4] (rozdział 6.14).

4.2 Narzędzia wizyjne

Narzędzie wizyjne, to podprogram wykorzystywany przez procesy wizyjne, który implementuje funkcjonalności systemu iRVision. Wiele narzędzi wizyjnych może zostać wykorzystanych w jednym procesie wizyjnym, aby połączyć ich funkcjonalność. Dokumentacja iRVision zawiera opis 46 różnych narzędzi. Najbardziej przydatne z nich zostały opisane poniżej.

4.2.1 Lokalizacja obiektów

Lokalizacja na podstawie kształtów geometrycznych: (*ang. GPM Locator Tool*) wykrywa krawędzie na wykonanym zdjęciu, porównuje je z nauczonego modelem i zwraca jego lokalizację. Umożliwia także edycję krawędzi modelu, dodanie

obszaru ignorowanego oraz obszaru szczególnie istotnego. Zwykle jest wykorzystywany jako narzędzie pojedynczego procesu 2D. Dokładny opis narzędzia i sposobu korzystania z niego znajduje się w rozdziale 3.8.2.

Opis w dokumentacji: [4] (rozdział 7.1).

Lokalizacja na podstawie zakrzywionych powierzchni: (*ang. Curved Surface Locator Tool*) wykrywa obiekt i określa jego pozycję na podstawie światła odbitego od zakrzywionej powierzchni w nieregularny sposób. Możliwości edycji modelu są podobne jak w *GPM Locator Tool*, poza operacjami na krawędziach.

Opis w dokumentacji: [4] (rozdział 7.2).

Lokalizacja na podstawie powierzchni obiektu: (*ang. Blob Locator Tool*) wykonuje progowanie obrazu, pozostawiając obraz w dwóch kolorach. Umożliwia zmianę parametrów progowania, takich jak próg (*ang. threshold*). Następnie lokalizowane są regiony na obrazie, które symbolizują powierzchnię obiektów. Możliwe jest wyznaczenie środka obrotu elementu w miejscu wyznaczonego środka bezwładności.

Opis w dokumentacji: [4] (rozdział 7.3).

Lokalizacja na podstawie linii prostej: (*ang. Line Locator Tool*) szuka na obrazie linii o zadanej wcześniej długości (może to być krawędź lub podłużny element, zależnie od ustawienia). Istnieje możliwość zmiany grubości linii oraz ograniczenia zmiany orientacji (np. jeśli wiadomo że linia musi być pozioma). Narzędzie zwraca położenie linii. Wyjściowa pozycja linii jest interpretowana inaczej niż pozycja obiektu. W przypadku linii może to być pozycja gdziekolwiek na niej.

Opis w dokumentacji: [4] (rozdział 7.4).

Połączenie wielu narzędzi lokalizacji: (*ang. Combination Locator Tool*) umożliwia wyznaczenie położenia obiektu na podstawie kilku skonfigurowanych narzędzi do lokalizacji (np. dwóch narzędzi lokalizacji linii). Są one dodane do narzędzia *Combination Locator Tool* jako składowe, razem z narzędziem do obliczania pozycji (opisanym w rozdziale 4.2.3).

Opis w dokumentacji: [4] (rozdział 7.5).

Lokalizacja wielu różnych elementów w jednym procesie: (*ang. Multi-Locator Tool*) pozwala dołączyć wiele narzędzi lokalizacji jako składowe. Proces uruchamia jednocześnie wszystkie narzędzia i zapisuje w rejestrze wizyjnym $VR[]$ dane wyjściowe z narzędzia o numerze odpowiadającym wartości rejestru $R[]$ (określonego w ustawieniach). Do procesu nie mogą być dodane narzędzia różnego typu.

Opis w dokumentacji: [4] (rozdział 7.10).

Lokalizacja w różnych częściach obrazu: (*ang. Multi-Window Tool*) powoduje uruchomienie składowego narzędzia lokalizacji dla ustalonego fragmentu obrazu (okna – *ang. window*). Umożliwia dodanie kilku okien i wybranie aktywnego poprzez wartość rejestru $R[]$. Numer rejestru można wybrać w opcjach narzędzia.

Opis w dokumentacji: [4] (rodział 7.11).

Lokalizacja punktów krawędzi na obrazie: (*ang. Edge Points Locator Tool*) pobiera z obrazu punkty na krawędziach obiektu. Narzędzie wykorzystywane przez proces wizyjny (*ang. Image To Points Vision Process*). Istnieje możliwość wykrywania krawędzi nawet gdy na obrazie jest ona nieciągła. Narzędzie zwraca punkty lub łańcuchy punktów (ciągłe krawędzie). System udostępnia także narzędzie do lokalizacji niektórych punktów krawędzi.

Opis w dokumentacji: [4] (rozdziały 7.31 i 7.32).

4.2.2 Kontrola jakości

Kontrola jakości elementu na krawędziach: (*ang. Bead Inspection Tool*) tworzy punkty kontrolne wzdłuż linii określonych podczas uczenia. Narzędzia składowe wykonują kontrolę w punktach kontrolnych. Służy głównie do sprawdzenia ciągłości krawędzi oraz tego, czy jest poszarpana.

Opis w dokumentacji: [4] (rozdział 7.19).

Kontrola jakości elementu na powierzchni: (*ang. Surface Flaw Inspection Tool*) znajduje obszary o innej barwie (wady) na powierzchni elementu. Jest możliwość ustawienia wielkości wad oraz ich minimalnego kontrastu z powierzchnią elementu. Można także wprowadzić maskę na wybrany obszar powierzchni elementu. Narzędzie zawiera dodatkowe opcje usuwania tła (statyczną, dynamiczną, wyrównywanie oświetlenia) oraz filtry, które można nałożyć na obraz (wyostanie, rozmycie, erozja, rozszerzanie). Narzędzie zlicza liczbę wad, ich powierzchnię oraz stosunek powierzchni wad do powierzchni elementu.

Opis w dokumentacji: [4] (rozdział 7.18).

Kontrola odległości między krawędziami: (*ang. Edge Pair locator Tool*) znajduje równoległe krawędzie i mierzy odległość między nimi. Może zostać wykorzystany do kontroli wymiarów obiektu lub jego części.

Opis w dokumentacji: [4] (rozdział 7.6).

Tworzenie warunków logicznych testu jakości: (*ang. Evaluation Tool*) jest podstawowym narzędziem tworzenia testów jakościowych dla elementu. Tworzy warunki logiczne na podstawie zmiennych zwracanych przez inne narzędzia do kontroli jakości oraz narzędzia do lokalizacji znajdujące się na tym samym poziomie. Istnieje możliwość określenia warunku powodzenia testu: wszystkie warunki spełnione, jeden spełniony, ostatni spełniony.

Opis w dokumentacji: [4] (rozdział 7.17).

4.2.3 Narzędzia matematyczne

Obliczenia arytmetyczne na obrazach: (*ang. Arithmetic Calculation Tool*) przeprowadza podstawowe operacje matematyczne (suma, różnica, iloczyn, iloraz, reszta z ilorazu) na wartościach zwracanych przez inne narzędzia wizyjne. Opis w dokumentacji: [4] (rozdział 7.21).

Obliczenia geometryczne na obrazach: (*ang. Geometric Calculation Tool*) implementuje działania takie jak wyznaczanie odległości między dwoma punktami,

wyznaczanie kąta między prostymi, zdefiniowanymi przez punkt przecięcia oraz po jednym punkcie na płaszczyźnie dla każdej prostej.

Opis w dokumentacji: [4] (rozdział 7.22).

Obliczenia statystyczne na obrazach: (*ang. Statistic Calculation Tool*) pozwala obliczyć średnią oraz odchylenie standardowe dla wartości zwracanych przez narzędzia do lokalizacji obiektów na podstawie wielu znalezionych obiektów (np. średnia powierzchnia elementu).

Opis w dokumentacji: [4] (rozdział 7.23).

Obliczenia pozycji na obrazach: (*ang. Position Calculation Tool*) wylicza pozycję na podstawie fragmentów obrazu znalezionych przez inne narzędzia. możliwe obliczenia to: przecięcie dwóch linii, środek odcinka między dwoma punktami, środek okręgu, punkt przecięcia linii z prostopadłą przechodzącą przez punkt wejściowy, prostopadła do linii, linia równoległa, linia przechodząca przez dwa punkty.

Opis w dokumentacji: [4] (rozdział 7.24).

4.3 Zaawansowane funkcje programowe iRVision

System iRVision zawiera także dodatkowe funkcje programowe, pozwalające w pełni wykorzystać jego możliwości w bardziej skomplikowanych aplikacjach.

- **VOFFSET CONDITION** – Pozwala zdefiniować domyślny rejestr wizyjny, wykorzystywany w programie, przekazywany jako parametr. Po zdefiniowaniu można wykorzystać funkcję **VOFFSET** bez parametru, zamiast **VOFFSET,VR**.
- **LOCK VREG, UNLOCK VREG** – Funkcje umożliwiają blokowanie rejestru wizyjnego. Zablockowany rejestr wizyjny nie może być aktualizowany.
- **CAMERA_VIEW[a]** – Opcja metody **RUN_FIND**, która umożliwia wybór innej kamery.
- **GET_NFOUND** – Zapisuje w rejestrze $R[a]$ liczbę znalezionych obiektów przez ostatnio uruchomiony proces wizyjny.
- **GET_PASSFAIL** – Zapisuje w rejestrze $R[a]$ informację o tym, czy obiekt przeszedł test jakości czy nie.
- **SET_REFERENCE** – Ustawia aktualną pozycję obiektu jako pozycję referencyjną (od której liczony jest offset).
- **CAMERA_CALIB** – Przeprowadza kalibrację kamery.
- **VISION_OVERRIDE** – Pozwala przeprowadzić proces wizyjny ze zmienionymi ustawieniami.

Dokładne opisy wszystkich funkcji programowych, rejestrów wizyjnych oraz programów KAREL znajduje się w dokumentacji [4] w rozdziałach kolejno: 9.2.2, 9.2.3 oraz 9.4.

Rozdział 5

Procedura Vision TCP

Instrukcja została przygotowana dla kamery podłączonej przez interfejs USB 3.0 (lub USB 2.0) do iPendanta robota. Opis przygotowano dla kontrolerów *R-30iB* z wersją oprogramowania starszą niż *V8.30*, ale nowszą niż *V7.30*. Proces kalibracji był testowany na rzeczywistym robocie *Arc Mate 100iC*. Kontroler robota w laboratorium 010 (*R-30iB Mate*) spełnia warunki, jednakże aktualnie nie jest na nim zainstalowana wymagana opcja *iRCalibration*.

5.1 Przygotowanie stanowiska

5.1.1 Sprzęt

Należy przygotować następujące elementy:

- znacznik TCP do Vision TCP,
- kamerę z interfejsem USB 3.0 lub 2.0.

Kamery USB obsługiwane przez sterownik:

- Basler acA640 20um,
- Kowa SC36MF.

5.1.2 Oprogramowanie

Do oprogramowania sterownika robota powinny być dołączone następujące moduły:

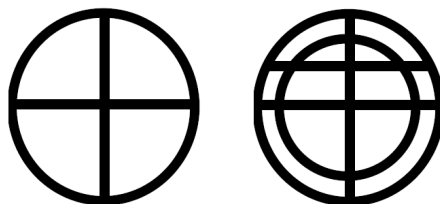
- *iRCalibration VTCP (J990)*,
- *KAREL (R632)*,
- *iRVision 2D pkg (R685)*.

Aby sprawdzić jakie moduły są zainstalowane należy wybrać opcję:

- *MENU* → *NEXT* → *STATUS* → *versionID* → *NEXT* → *ORDERFILE*

Na sterowniku muszą się znaleźć opcje:

- *MENU* → *UTILITIES* → *iRCalibration* → *Vision TCP Set*



Rysunek 5.1: Przykładowe znaczniki do kalibracji



Rysunek 5.2: Przykładowy adapter znacznika

Biblioteka sterownika musi zawierać programy KAREL:

- VSETTCP
- VCRSMCHECK

Dodatkowo biblioteka może zawierać programy KAREL:

- VTOFSTUT

5.2 Znacznik TCP

Poniższe wytyczne przedstawiają w jaki sposób dobrać i zamocować znacznik narzędzia robota.

Przygotowanie znacznika: Do przeprowadzenia procesu kalibracji z użyciem wizji należy przygotować znacznik, który można zamontować na flanszy robota zamiast narzędzia lub na narzędziu. Przykładowe znaczniki pokazano na rysunku 5.1. Barwy znacznika muszą być kontrastujące, linie proste oraz obramowanie widoczne żeby mogły być bezbłędnie rozpoznawalne przez system wizyjny.

Mocowanie znacznika: Znacznik trzeba zamocować na ostatnim przegubie robota (j6) w taki sposób, żeby nie przesunął się w trakcie kalibracji. Środek znacznika powinien być w środku układu współrzędnych docelowego TCP, na płaszczyźnie XY tego układu. Robot musi mieć możliwość podniesienia znacznika 400mm powyżej kamery. Przykładowy adapter znacznika mocowany jako końcówka robocza narzędzia spawającego został przedstawiony na rysunku 5.2.

Oświetlenie znacznika: Należy przygotować oświetlenie znacznika od dołu białym, rozproszonym światłem, żeby był lepiej widoczny dla kamery.



(a) Kamera USB 2.0 Kowa SC36MF



(b) Kamera USB 3.0 Basler acA640

Rysunek 5.3: Kamery USB dla robotów FANUC



Rysunek 5.4: Położenie portu USB na iPendancie

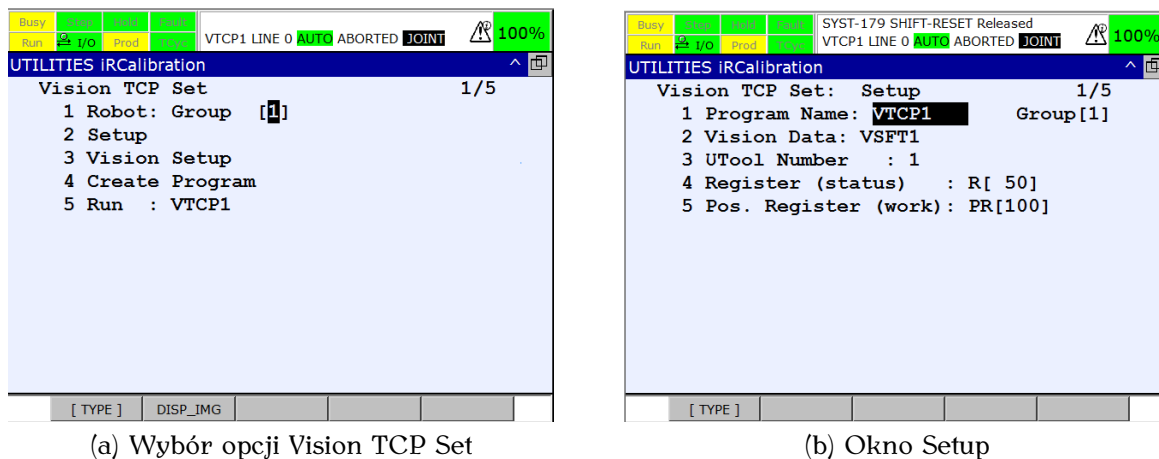
5.3 Kamera USB

Aby dobrać odpowiednią kamerę i obiektyw, a następnie dokonać ich konfiguracji, należy kierować się poniższymi wytycznymi.

Wybór kamery: Procedura ta może zostać wykonana z użyciem kamery z interfejsem przemysłowym. Musi jednak być umieszczona w położeniu, umożliwiającym wykonanie zdjęcia narzędzia robota. Kamera USB, która nie jest umieszczona na stałe w przestrzeni roboczej ułatwia wykonanie procedury, gdyż może być zamocowana w odpowiednim miejscu. Nie jest konieczna zmiana zamocowania głównej kamery i utraty kalibracji. Jedną kamerę USB można bez problemu wykorzystać do definicji układu współrzędnych narzędzia wielu robotów. Kamery USB obsługiwane przez sterownik zostały wypisane w rozdziale 5.1.1 pokazane na rysunku 5.3.

Mocowanie kamery: Kamerę należy zamontować nieruchomo względem układu współrzędnych globalnych, w taki sposób żeby nie przesuwała się podczas procesu kalibracji. Matryca powinna być zwrócona w górę, w kierunku znacznika TCP. Kamera powinna być umieszczona w przestrzeni roboczej tak, żeby robot był w stanie umieścić znacznik w odległości 400 mm od niej. Ustawić robota tak, żeby znacznik był skierowany w stronę kamery jak najbliżej jej osi optycznej w odległości 400 mm od obiektywu.

Podłączenie kamery: Kamerę należy podłączyć za pomocą kabla USB 3.0 lub 2.0 bezpośrednio do iPendanta robota. Położenie gniazda pokazane na rysunku 5.4.



(a) Wybór opcji Vision TCP Set

(b) Okno Setup

Rysunek 5.5: Podstawowe okna opcji Vision TCP Set

Dobranie obiektywu: Obiektyw dobrać według wytycznych opisanych w rozdziale 3.2. Podczas testów najlepszy okazał się obiektyw o ogniskowej 20 mm.

Setup kamery: Konfigurację kamery należy przeprowadzić dokładnie tak, jak do zwykłego procesu iRVision z kamerą podłączoną do kontrolera (rozdział 3.4). Należy wybrać odpowiedni typ kamery, używany podczas kalibracji (Basler USB lub Kowa USB).

Ustawienie przesłony i ogniskowej: Po ustawieniu znacznika w odpowiednim miejscu dokonać ręcznego ustawienia przesłony i ogniskowej, tak jak dla zwykłego procesu wizyjnego (rozdział 3.4).

5.4 iRCalibration

5.4.1 Ustawienia

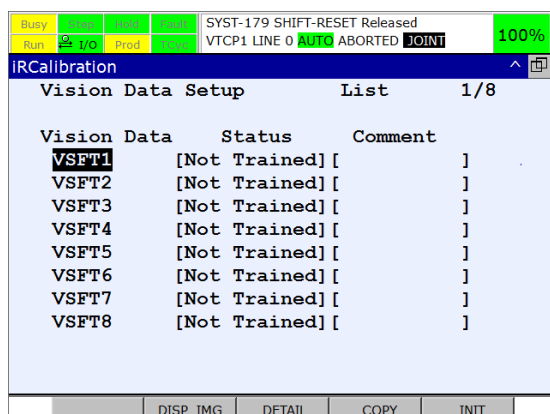
Vision TCP Set: Wybrać Vision TCP Set:

1. Na iPendancie wybrać opcję *Menu* → *Utilities* → *iRCalibration*.
2. Wybrać *Vision TCP Set* (rysunek 5.5a).
3. Wybrać grupę robota (domyślnie 1).

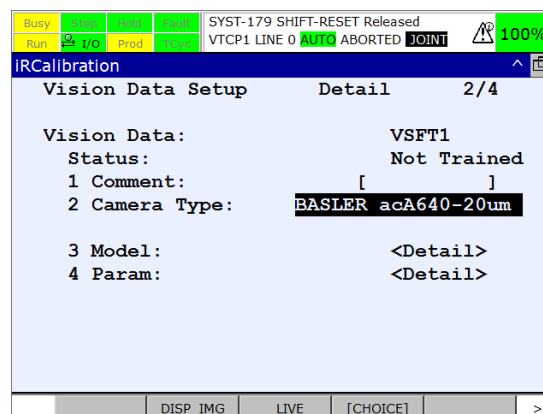
Setup: Wykonać *Setup* – Vision TCP (rysunek 5.5b):

1. Wpisać nazwę generowanego programu.
2. Wybrać *Vision Data*, proces robota uczony w *Vision Setup*.
3. Wybrać numer narzędzia *UTOOL* uczonego podczas procesu.
4. Wybrać numery rejestru i rejestru pozycyjnego, użytych w generowanym programie.

UWAGA: Sprawdzić czy rejestry nie są wykorzystywane w innym programie.

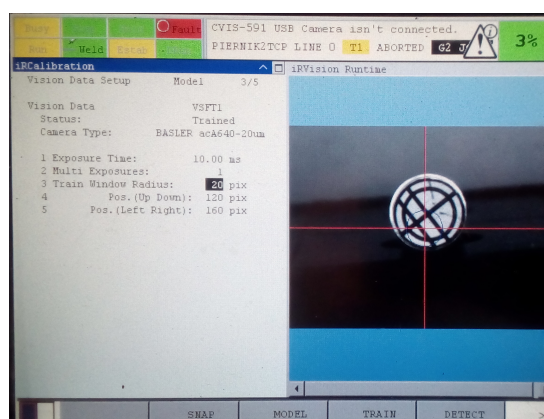


(a) Okno Vision Setup



(b) Okno Vision Data

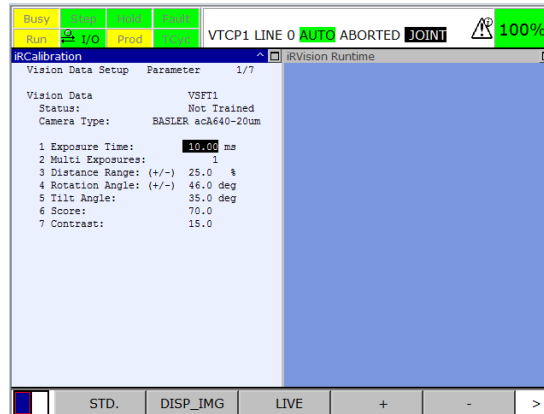
Rysunek 5.6: Okna konfiguracji funkcji Vision TCP Set



Rysunek 5.7: Okno uczenia modelu – Vision TCP Set

Vision Setup: Wykonać Vision Setup – Vision TCP:

1. Wybrać proces **Vision Data** do nauczania (wybrany w poprzednim punkcie) (rysunek 5.6a).
2. Wybrać typ kamery używanej do kalibracji w oknie pokazanym na rysunku (5.6b).
3. Nauczyć robota modelu **Vision Data** w oknie **Model** (rysunek 5.7).
 - (a) Wcisnąć przyciski **DISP_IMG** oraz **LIVE**, w celu wyświetlenia *Vision Runtime* w drugim oknie. Można to też zrobić w przeglądarce komputera.
 - (b) Ustawić *Exposure Time* na 10 ms, *Multi Exposures* na 1.
 - (c) Na obrazie, poprzez zmianę liczby pikseli w opcjach, ustawić markery w taki sposób żeby tylko znacznik znajdował się w okręgu:
 - Train Window Radius – Promień okręgu wykrywania,
 - Pos. (Up, Down) – Pozycja okręgu (górną–dół),
 - Pos. (Left, Right) – Pozycja okręgu (lewo–prawo).
 - (d) Przejść do następnych opcji (Next), wcisnąć **TRAIN**.
 - (e) Sprawdzić wykrywanie modelu wciskając **DETECT**.



Rysunek 5.8: Okno parametrów wykrywania znacznika – Vision TCP Set

4. Ustawić parametry wykrywania w oknie pokazanym na rysunku 5.8.
 - (a) **Exposure Time** (czas ekspozycji), taki jak podczas uczenia.
 - (b) **Multi Exposures** (ekspozycja wielokrotna), tak jak podczas uczenia.
 - (c) **Distance Range** (możliwa zmiana odległości).
 - (d) **Rotation Angle** (możliwa zmiana kąta obrotu znacznika na obrazie).
 - (e) **Tilt Angle** (możliwa zmiana kąta nachylenia znacznika).
 - (f) **Score** (próg wykrywania nauczonego znacznika).
 - (g) **Contrast** (próg kontrastu podczas wykrywania).

Parametry przedstawiają zakresy poszczególnych zmiennych, na podstawie których na obrazie następuje detekcja znacznika. Ustawić je w taki sposób, aby były jak najmniejsze, ale w każdej pozycji robot podczas procesu mógł wykryć znacznik.

Dokładny opis procesu kalibracji chwytaka z użyciem modułu iRCalibration znajduje się w dokumentacji [5] (rozdział 8).

5.4.2 Uruchamianie

Generowanie programu: Należy wrócić do menu *Vision TCP Set* (rysunek 5.5a). Następnie wybrać opcję *Create Program*. Zostanie wygenerowany *Teach Pendant Program* o podanej wcześniej nazwie (rysunek 5.9).

Zwiększanie dokładności: Żeby zwiększyć dokładność kalibracji TCP metodą *Vision TCP* należy wykorzystać program KAREL o nazwie *VTCPINIT*. Komenda *VTCPINIT* zmienia wartość podanego w pierwszym argumencie parametru na wartość podaną jako drugi argument. Linie zmieniające parametry wyznaczania osi TCP należy dodać do wygenerowanego programu w następnej linii po wywołaniu funkcji *VTCPINIT* bez parametrów (rysunek 5.9).

- Dokładność wyznaczania osi X, Y można zwiększyć, zmieniając maksymalny kąt obrotu znacznika oraz dzielnik. Dzielnik zapewnia większą liczbę kroków podczas kalibracji (np. gdy kąt będzie wynosił 90°, a dzielnik 3, kąty w kolejnych krokach będą wynosić: -90°, -60°, -30°, 0°, 30°, 60°, 90°). Nazwy oraz rekomendowane i domyślne wartości parametrów wyznaczania osi X, Y zostały podane w tablicy 5.1.

```

VTCP1 LINE 0 AUTO ABORTED JOINT 100%
VTCP1 1/57
1: CALL VCRSM_CHECK('START')
2: UFRAME_NUM=0
3: UTOOL_NUM=1
4:
5: IF $VCRSM_CFG.$STEP_NUM>=2,
  : JMP LBL[2]
6: CALL VCRSM_CHECK(1)
7:J @P[1] 30% FINE
8: CALL VTCPINIT('INIT',0)
9: CALL VTCPINIT
10: CALL VTCPSET('VSFT1',1,1)

VTCP1 8/63
7:J P[1] 30% FINE
8: CALL VTCPINIT('INIT',0)
9: CALL VTCPINIT
10: CALL VTCPSET('VSFT1',1,1)
11:
12: CALL VTCPINIT('MOVE_DIST_R',90)
13: CALL VTCPINIT('DIV_NUM_R',3)
14: CALL VTCPINIT('MOVE_DIST_W',45)
15: CALL VTCPINIT('MOVE_DIST_P',45)
16: CALL VTCPINIT('DIV_NUM_WP',4)
17:
  
```

Rysunek 5.9: Przykładowy, wygenerowany program VTCP1

Tablica 5.1: Parametry wyznaczania osi X, Y TCP

Parametr	Wyjaśnienie	Domyślna	Rekomendowana
MOVE_DIST_R	Kąt obrotu wokół osi optycznej kamery	15°	60 – 90°
DIV_NUM_R	Dzielnik kąta obrotu	1	2 – 3

- Dokładność wyznaczania osi Z można zwiększyć, zmieniając maksymalne kąty pochylenia znacznika oraz dzielnik. Nazwy oraz rekomendowane i domyślne wartości parametrów wyznaczania osi z zostały podane w tabelicy 5.2.

UWAGA: Ustawiając parametry wyznaczania TCP zwrócić uwagę na przeszkody wokół robota.

Działanie programu: Podczas wykonywania programu należy cały czas trzymać przycisk *Deadman* w pozycji 1 oraz wciśnięty SHIFT. Jeśli program zatrzyma się, należy zmniejszyć kąty obrotu i pochylenia lub zwiększyć tolerancję na błędnie wykryty model w *Vision Setup* → *Parameters* i wznowić wykonywanie programu. Gdy program zakończy działanie, określony w *Setup* → *UTOOL* TCP zostanie nadpisany i skalibrowany na środek znacznika.

Ręczna zmiana kalibracji Domyślnie oś Z po kalibracji jest ustawiona w głąb narzędzia, zgodnie z kierunkiem osi optycznej kamery i ze zwrotem od kamery do znacznika. Zdefiniowany układ współrzędnych można edytować, z użyciem programu KAREL *VTOFSTUT*. Wystarczy wywołać *VTOFSTUT* w dowolnym Teach Pendant Program z odpowiednimi parametrami. Składnia wywołania programu *VTOFSTUT* ma postać:

- CALL VTOFSTUT (GroupNumber, SourceUTool, DestinationUTool, dx, dy, dz, [dw, dp, dr])

Gdzie argumenty oznaczają kolejno:

- GroupNumber – numer robota,

Tablica 5.2: Parametry wyznaczania osi Z TCP

Parametr	Wyjaśnienie	Domyślna	Rekomendowana
MOVE_DIST_W	Kąt pochylenia znacznika względem osi X	15°	35 – 45°
MOVE_DIST_P	Kąt pochylenia znacznika względem osi Y	15°	35 – 45°
DIV_NUM_WP	Dzielnik kątów pochylenia	1	2 – 4

- SourceUTool – numer układu współrzędnych narzędzia (*ang. Tool Frame*, dla którego chcemy ustalić przemieszczenie (*ang. offset*),
- DestinationUTool – numer ToolFrame'a, który ma zostać nadpisany,
- dx, dy, dz – przesunięcie w kierunku osi X, Y, Z (w mm),
- dw, dp, dr – obrót wokół osi X, Y, Z (w °).

W ten sposób można przykładowo zmienić zwrot osi Z narzędzia robota, obracając jego układ współrzędnych wokół osi X lub Y o 180°.

Rozdział 6

System iRVision w laboratorium 010

W laboratorium 010 kamera została zintegrowana z robotem FANUC LR Mate 200iD 4S. System iRVision był instalowany przez pracownika wsparcia technicznego firmy FANUC. Poniżej przedstawiono wyposażenie stanowiska robota z systemem wizyjnym oraz uwagi i rady dotyczące wykorzystania wizji w testach i aplikacjach.

6.1 Wyposażenie

Kamera:

- model: Sony Analog Camera XC-56
- typ: kamera analogowa
- sensor: typ 1/3"

Obiektyw:

- ogniskowa: 10 – 40 mm
- montowanie: C-Mount

Oświetlenie:

- lampy halogenowe sufitowe (główne)
- lampa laboratoryjna led (dodatkowe)

Siatka kalibracyjna:

- siatka kalibracyjna 30 mm na kartce papieru A3
- siatka kalibracyjna 50 mm na kartce papieru A2

Przestrzeń robocza:

- stół półokrągły MDF przytwierdzony do podstawy robota
- czarne prostopadłościaki z wyciętym otworem o średnicy około 43 mm

Narzędzie robota:

- chwytak palcowy 40 mm
- chwytak palcowy 20 mm
- adapter dla obu chwytaków
- adapter kamery

6.2 Uwagi dotyczące korzystania

Oświetlenie stanowiska:

- Koniecznie należy włączyć lampy nad stanowiskiem robotycznym.
- Żaluzje mogą być zasłonięte lub odsłonięte.
- Opcjonalnie włączyć lampę laboratoryjną przyczepioną do stołu za stanowiskiem i skierować ją na przestrzeń roboczą robota.
- Jeśli będzie możliwość, dokupić oświetlacz pierścieniowy i zamontować wokół kamery (rozdział 3.2). Konieczne będzie przeprojektowanie mocowania kamery.

Obiektyw:

- Ustawić minimalną ogniskową aktualnego obiektywu. Powyżej 10 mm kąt widzenia jest za mały dla tej aplikacji.
- Jeśli będzie możliwość, wymienić obiektyw na taki z ogniskową 8 mm lub 6 mm (rozdział 3.2).

Połączenie przez Ethernet:

- Zaleca się pracę z systemem iRVision z poziomu przeglądarki internetowej komputera, połączanego z kontrolerem przez Ethernet (rozdział 3.1).

RoboGuide:

- Możliwe jest przeniesienie programów oraz konfiguracji robota do środowiska RoboGuide i odwrotnie z RoboGuide na kontroler robota. Najłatwiej zrobić to przez backup.
- Zaleca się przetestowanie kalibracji oraz konfiguracji procesów i narzędzi roboczych w środowisku RoboGuide przed implementacją na rzeczywistym robocie.
- Aplikacje na robota można pisać w środowisku RoboGuide lub w pliku tekstowym, następnie importować do RoboGuide i na kontroler robota. Może to przyspieszyć pracę z bardziej skomplikowanymi aplikacjami.

Konfiguracja iRVision

- Konfigurację systemu wizyjnego przedstawiono na rysunku 3.7.
- Ustawienia kamery przedstawiono na rysunku 3.8.

- Ustawienia automatycznego definiowania układu współrzędnych przedstawiono na rysunku [3.11](#).
- Ustawienia kalibracji kamery przedstawiono na rysunku [3.12](#).
- Ustawienia procesu wizyjnego przedstawiono na rysunku [3.14a](#).
- Ustawienia narzędzia wizyjnego przedstawiono na rysunku [3.14b](#).
- Program testowy oraz podstawowy program wizyjny opisano w rozdziałach [7.2](#) oraz [7.4](#).

Rozdział 7

Przykładowa implementacja iRVision

W ramach pracy inżynierskiej zostały przygotowane programy demonstracyjne, które służą prezentacji podstawowych funkcji systemu iRVision na rzeczywistym robocie w laboratorium 010.

7.1 Zawartość rejestrów robota

Zdefiniowano rejestry robota, które są wykorzystywane przez aplikacje demonstracyjne systemu iRVision.

7.1.1 Rejestry ogólne

- R[180:irv_klocek _nfound] – liczba obiektów znalezionych w procesie wizyjnym
- R[181:irv_test_passed] – wskaźnik powodzenia testu jakości

7.1.2 Rejestry pozycyjne

- PR[88:irv_soffset] – przesunięcie w osi Z, które gwarantuje bezpieczny ruch chwytaka
- PR[88:irv_coffset] – przesunięcie w osiach X, Y, wykorzystywane przy przeszukiwaniu pozycji roboczej
- PR[88:irv_toffset] – przesunięcie chwytaka, podczas chwytania obiektu
- PR[88:irv_refpos] – pozycja robienia zdjęcia we współrzędnych globalnych
- PR[88:irv_UFrefpos] – pozycja robienia zdjęcia we współrzędnych użytkownika (User Frame 9)
- PR[88:irv_inspecpos] – pozycja robienia zdjęcia do procesu kontroli jakości
- PR[88:irv_basic_point] – pozycja w środku otworu obiektu
- PR[88:irv_foundpos] – pozycja znalezionego obiektu
- PR[88:irv_refposc] – pozycja referencyjna obiektu

```

1: UFRAME_NUM=8 ;
2: UTOOL_NUM=2 ;
3: !punkt robienia zdjecia ;
4: J P[1] 100% FINE ;
5: VISION RUN_FIND 'IRV_2D_PROCESS' ;
6: VISION GET_OFFSET 'IRV_2D_PROCESS' VR[8] JMP LBL[1] ;
7: !obliczanie offset ;
8: PR[95]=VR[8].FOUND_POS[1] ;
9: PR[87:irv_mvoffset]=PR[95:irv_foundpos]-PR[96:irv_refposc] ;
10: !dojazd do punktu ;
11: J P[2] 100% FINE Offset,PR[87:irv_mvoffset] ;
12: J P[2] 100% FINE VOFFSET,VR[8] ;
13: LBL[1] ;

```

Rysunek 7.1: Kod źródłowy programu *IRV_T_LOCATOR_TEST*

- PR[88:irv_goodpos] – pozycja odkładania elementów, które przeszły test jakości
- PR[88:irv_badpos] – pozycja odkładania elementów, które nie przeszły testu jakości
- PR[88:irv_zero] – wyzerowany rejestr, wykorzystywany do obliczeń

7.1.3 Rejestry wizyjne

- VR[9:irv_qtest_process] – rejestr wizyjny dla procesu kontroli jakości
- VR[10:irv_2D_process] – rejestr wizyjny dla procesu 2D

7.1.4 Układy współrzędnych

- Tool Frame 1 – układ współrzędnych kamery
- Tool Frame 2 – układ współrzędnych dużego chwytaka palcowego
- User Frame 9 – układ współrzędnych dla aplikacji demonstracyjnych

7.2 Program testowy narzędzia lokalizacji

Aplikacja służy sprawdzeniu, czy kalibracja została wykonana poprawnie. Wywołuje proces wizyjny 2D i oblicza przemieszczenie elementu względem pozycji referencyjnej dwiema metodami:

- przez system iRVision,
- jako różnica pozycji znalezionej i referencyjnej,

co pozwala sprawdzić, czy kalibracja kamery i ustawienia procesu wizyjnego zostały przeprowadzone poprawnie.

Kod źródłowy programu został przedstawiony na rysunku 7.1.

Jeśli różnice między offsetami są duże, należy powtórzyć kalibrację. W programie robot dojeżdża do punktu, z którego wykona podniesienie obiektu z każdym

```
1: UFRAME_NUM=9 ;
2: UTOOL_NUM=2 ;
3: !punkt robienia zdjecia ;
4: J P[1] 100% FINE ;
5: VISION RUN_FIND 'IRV_2D_PROCESS' ;
6: VISION GET_OFFSET 'IRV_2D_PROCESS' VR[10] JMP LBL[1] ;
7: !dojazd do punktu inspekcji ;
8: J PR[84:irv_qtest_basic] 100% FINE VOFFSET,VR[10] ;
9: VISION RUN_FIND 'IRV_INSPECTION_PROCESS' ;
10: LBL[1] ;
```

Rysunek 7.2: Kod źródłowy programu *IRV_T_INSPECTION_TEST*

z przesunięć, dla porównania. Wartości przesunięć można także sprawdzić w rejestrach VR[10] oraz PR[87].

UWAGA: Po przeprowadzeniu zmian w kalibracji lub procesie wizyjnym, należy wpisać pozycję referencyjną do rejestru PR[85] oraz ponownie nagrać punkt P[2], jako punkt w środku otworu obiektu.

UWAGA: Błędy mogą wynikać także z innych przyczyn niż niepoprawna kalibracja (np. niedokładnie wykonany element).

UWAGA: Porównanie przesunięć jest właściwym testem poprawności kalibracji tylko, gdy poprawnie zlokalizowano obiekt.

Wykorzystane procesy:

- 2D Single View Vision Process – nazwa: IRV_TEST_TABLEP

Wykorzystane narzędzia:

- GPM Locator Tool

7.3 Program testowy narzędzia inspekcji

Aplikacja umożliwia ręczne dostrojenie zakresów dopuszczalnych błędów w celu dobrania tolerancji dla wad obiektów. Wywołuje proces wizyjny kontroli jakości. Pozwala to sprawdzić w oknie *Vision Runtime*, czy test jakości się powiódł, a jeśli nie, które z jego narzędzi odrzuciło obiekt.

Kod źródłowy programu został przedstawiony na rysunku 7.2.

Wykorzystane procesy:

- Single View Inspection Vision Process – nazwa: IRV_INSPECTION_PROCESS

Wykorzystane narzędzia: Wykorzystane narzędzia:

- GPM Locator Tool
- Surface Flaw Inspection Tool
- Edge-Pair Locator Tool
- Arithmetic Calculation Tool
- Evaluation Tool

```

2: UTOOL_NUM=2 ;
3: !punkt robienia zdjecia ;
4: J P[1] 100% FINE ;
5: VISION RUN_FIND 'IRV_2D_PROCESS' ;
6: VISION GET_OFFSET 'IRV_2D_PROCESS' VR[10] JMP LBL[1] ;
7: !punkt dojazdu ;
8: J P[2] 100% FINE VOFFSET,VR[10] ;
9: !jesli klocka brak ;
10: LBL[1] ;

```

Rysunek 7.3: Kod źródłowy programu *IRV_T_BASIC*

7.4 Podstawowy program iRVision

Elementarny program wykorzystujący proces *2D Single View Process* oraz narzędzie *GPM Locator Tool* do lokalizacji obiektu na stole. Po zlokalizowaniu obiektu robot przesuwa narzędzie do punktu w środku otworu w obiekcie. Program demonstruje najprostsze zastosowanie systemu w aplikacji *Pick and Place*.

Kod źródłowy programu został przedstawiony na rysunku 7.3.

Wykorzystane procesy:

- 2D Single View Vision Process – nazwa: *IRV_2D_PROCESS*

Wykorzystane narzędzia:

- GPM Locator Tool

7.5 Aplikacja demonstracyjna

Program, pokazujący w jaki sposób można wykorzystać system iRVision do zadania lokalizacji obiektów oraz kontroli jakości.

Kod źródłowy aplikacji demonstracyjnej został przedstawiony na rysunku 7.4.

Wykorzystane procesy:

- 2D Single View Vision Process – nazwa: *IRV_2D_PROCESS*
- Single View Inspection Vision Process – nazwa: *IRV_INSPECTION_PROCESS*

Wykorzystane narzędzia:

- GPM Locator Tool
- Surface Flaw Inspection Tool
- Edge-Pair Locator Tool
- Arithmetic Calculation Tool
- Evaluation Tool

W aplikacji zastosowano podział na podprogramy. Wszystkie punkty są zapisane w rejestrach *PR* i opisane komentarzem. Przebieg działania aplikacji:

```
1: UFRAME_NUM=9 ;
2: UFRAME_NUM=2 ;
3: !zerowanie rejestrów ;
4: CALL IRV_CLEAR_REGS ;
5: LBL[1] ;
6: !szukanie klocek ;
7: CALL IRV_FIND_KLOCKI ;
8: IF (R[180:irv_klocek_nfoud]=0) THEN ;
9: !zwiększ X ;
10: CALL IRV_X_INCREMENT ;
11: !zwiększ Y ;
12: CALL IRV_Y_INCREMENT ;
13: ELSE ;
14: IF (R[180:irv_klocek_nfoud]>0) THEN ;
15: !obliczanie offsetu ;
16: CALL IRV_GETOFFSET ;
17: !test jakości ;
18: CALL IRV_QUALITY_TEST ;
19: CALL IRV_FIND_AFTERRKLOCKI ;
20: CALL IRV_GET_AFTEROFFSET ;
21: !podniesienie klocka ;
22: CALL IRV_GET_KLOCEK ;
23: !położenie klocka ;
24: CALL IRV_PUT_KLOCEK ;
25: ENDIF ;
26: ENDIF ;
27: JMP LBL[1] ;
```

Rysunek 7.4: Kod źródłowy programu *IRV_SWEEP_KLOCKI*

1. Zerowanie rejestrów: iteratorów pętli, przesunięć.
2. Poszukiwania obiektów we fragmencie przestrzeni roboczej: zrobienie zdjęcia w odpowiednim miejscu i wywołanie procesu wizyjnego *IRV_2D_PROCESS*.
3. Jeśli nie znaleziono obiektu, przesunięcie pozycji robienia zdjęcia w osi X, następnie Y.
4. Jeśli znaleziono co najmniej jeden obiekt, obliczanie przemieszczenia obiektu względem pozycji referencyjnej (*ang. offset*). Przemieszczenie obliczane jest dwiema metodami.
5. Wykonanie testu jakości. Sprawdzenie powierzchni i wymiarów obiektów w poszukiwaniu wad, wywołując proces wizyjny *IRV_INSPECTION_PROCESS*.
6. Kolejne poszukiwanie obiektu, służące zwiększeniu dokładności wyliczanego przemieszczenia.
7. Podniesienie znalezionej obiektu. Ustawienie robota w pozycji środka otworu elementu (z przemieszczeniem obliczonym obydwoma metodami dla porównania). Następnie wsunięcie chwytaka w otwór obiektu, otwarcie chwytaka i podniesienie obiektu. Wykorzystano przemieszczenia względem narzędzia (*ang. tool offset*) i pozycji robota (*ang. offset*).
8. Odłożenie obiektu na miejsce. Tymczasowo jest to stała pozycja w zaznaczonym na stole obszarze na paletę. Docelowo można połączyć aplikację z procesem paletyzacji.

Rozdział 8

Podsumowanie

Praca została napisana w celu ułatwienia jej czytelnikowi korzystania z systemu iRVision zainstalowanego na kontrolerze robota *LR Mate 200iD 4S* znajdującego się w laboratorium 010 w budynku C-3 Politechniki Wrocławskiej w stopniu podstawowym i średnio zaawansowanym. W zamyśleniu praca ma stanowić podstawę do przygotowania ćwiczeń, wykonywanych przez studentów w czasie zajęć w tym laboratorium. Celem ćwiczeń powinno być pokazanie studentom zalet zastosowania systemów wizyjnych w przemysłowych środowiskach zrobotyzowanych oraz nauczenie ich posługiwania się systemem iRVision. W ramach pracy opisano następujące zagadnienia:

- zastosowanie systemu wizyjnego oraz jego możliwości i ograniczenia,
- kolejne kroki prowadzące do stworzenia aplikacji wykorzystującej wizję (od doboru sprzętu do gotowego programu),
- najbardziej użyteczne funkcje i programy systemu,
- przykładowe aplikacje testowe oraz demonstracyjne,
- implementację systemu wizyjnego w laboratorium 010,
- wyszukiwanie niezbędnych informacji w dokumentacji od firmy FANUC.

Bibliografia

- [1] Balluff. Oświetlacze do aplikacji wizyjnych. <http://www.balluff.com/local/pl/products/accessories/lighting-for-vision-systems/>.
- [2] V. Dimension. Obiektyw ricoh c-mount. https://www.vision-dimension.com/media/image/84/97/c8/FL-CC2514A-2M_Download_600x600.jpg.
- [3] FANUC CORPORATION. *FANUC Robot series R-30iB/R-30iB Mate CONTROLLER OPERATOR'S MANUAL (Basic Operation)*, 2015.
- [4] FANUC CORPORATION. *FANUC Robot series R-30iB/R-30iB Mate CONTROLLER iRVision OPERATOR'S MANUAL (Reference)*, 2016.
- [5] FANUC_ADEE. *FANUC Robot series R-30iA/R-30iA Mate/R-30iB/R-30iB Mate iRCalibration OPERATOR'S MANUAL*, 2013.
- [6] Kowa. Obiektyw lm6ncl 6 mm.
<http://kwaoptimed.com/lenses/en/LM6NCL.561.php>.
- [7] Kowa. Obiektyw lm8jc 8 mm.
<http://kwaoptimed.com/lenses/en/LM8JC.1496.php>.