

EPSON RC+ 7.0 オプション

*Vision Guide 7.0*

Ver. 7.5

ソフトウェア編

Rev.5

JAM231S5523F

翻訳版



EPSON RC+ 7.0 オプション

# *Vision Guide 7.0* (Ver.7.5) ソフトウェア編

Rev.5

©Seiko Epson Corporation 2012-2023

## はじめに

このたびは当社のロボットシステムをお求めいただきましてありがとうございます。  
本マニュアルは、EPSON RC+ ソフトウェアを正しくお使いいただくために必要な事項を記載したものです。  
システムをご使用になる前に、本マニュアルおよび関連マニュアルをお読みいただき、正しくお使いください。  
お読みになった後は、いつでも取り出せる所に保管し、不明な点があったら再読してください。

当社は、厳密な試験や検査を行い、当社のロボットシステムの性能が、当社規格に満足していることを確認しております。マニュアルに記載されている使用条件を超えて、当社ロボットシステムを使用した場合は、製品の基本性能は発揮されませんのでご注意ください。

本書の内容は、当社が予見する範囲の、危険やトラブルについて記載しています。当社のロボットシステムを、安全に正しくお使いいただくため、本書に記載されている安全に関するご注意は、必ず守ってください。

## ソフトウェアライセンス

コンパクトビジョンを使用する場合、Vision Guideを使用する前に以下のソフトウェアライセンス契約書を必ずお読みください。(ハードウェア&セットアップ編)

Appendix A: エンドユーザーソフトウェアライセンス契約(コンパクトビジョン)

Appendix B: オープンソースソフトウェアのライセンス契約(コンパクトビジョン)

## 商標

Microsoft、Windows、Windows ロゴ、Visual Basic、Visual C++ は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。その他の社名、ブランド名、および製品名は、各社の登録商標または商標です。

## 表記について

Microsoft® Windows® 8 operating system 日本語版

Microsoft® Windows® 10 operating system 日本語版

Microsoft® Windows® 11 operating system 日本語版

本取扱説明書では、上記オペレーティングシステムをそれぞれ、Windows 8, Windows 10, Windows 11 と表記しています。また、Windows 8, Windows 10, Windows 11 を総称して、Windows と表記することがあります。

## ご注意

本取扱説明書の一部、または全部を無断で複製や転載をすることはできません。

本書に記載の内容は、将来予告なく変更することがあります。

本書の内容について、誤りや、お気づきの点がありましたら、ご連絡くださいますようお願いいたします。

## 製造元

**セイコーエプソン 株式会社**

## お問い合わせ先

お問い合わせ先の詳細は、以下のマニュアル冒頭“販売元”に記載しています。

「ロボットシステム 安全マニュアル はじめにお読みください」

## ご使用の前に

マニュアルのご使用の前に、知っておいていただきたいことを記載しています。



### 安全上の注意

ロボットおよび関連機器の運搬と設置は、有資格者が行ってください。また、必ず各国の法規と法令に従ってください。

ご使用になる前に、本マニュアル、ならびに関連マニュアルをよくお読みの上、正しくお使いください。

お読みになった後は、いつでも取り出せる所に保管し、不明な点があったら再読してください。

### 記号の意味

 警告	この表示を無視して誤った取り扱いをすると、人が死亡、または重傷を負う可能性が想定される内容を示しています。
 注意	この表示を無視して誤った取り扱いをすると、人が障害を負う可能性が想定される内容および物的損害のみの発生が想定される内容を示しています。

## トレーニング

Vision Guide 7.0を使用する方は、弊社で行っている「Vision Guide導入トレーニング」を必ず受講してください。お客様に、製品をご理解いただくために、当社では定期的、または、都度トレーニングを実施しています。

正規のトレーニングを受講されると、製品が簡単に使用できるようになり、生産性を高めることができます。トレーニングの詳細は、販売元にお問い合わせください。

<b>1. マニュアル ヘルプ</b>	<b>1</b>
1.1 本マニュアルの構成 .....	1
1.2 関連マニュアル .....	2
1.3 オンラインヘルプの使用方法 .....	2
<b>2. 安全</b>	<b>3</b>
2.1 本文中の記号について .....	3
2.2 安全に関する注意事項 .....	3
2.3 ロボットの安全 .....	4
<b>3. はじめに</b>	<b>5</b>
3.1 Vision Guide 7.0の概要 .....	5
3.1.1 Vision Guide 7.0の目的 .....	5
3.1.2 Vision Guide 7.0の特徴 .....	5
3.2 前提となるEPSON RC+ 7.0の基本的知識 .....	6
<b>4. Vision Guideの概要</b>	<b>7</b>
4.1 概要 .....	7
4.2 Vision Guide 7.0を使用するために必要な基本的概念 .....	7
4.2.1 ビジョンシーケンス .....	7
4.2.2 ビジョンオブジェクト .....	7
4.2.3 プロパティ .....	8
4.2.4 リザルト .....	8
4.2.5 実行時ビジョンコマンド .....	8
4.2.6 EPSON RC+ 7.0プロジェクトでのVision Guide 7.0 .....	9
4.3 座標系 .....	9
4.3.1 イメージ画像座標系 .....	9
4.3.2 カメラ座標系 .....	9
4.3.3 ロボット座標系 .....	10
4.4 Vision Guideウィンドウを開く .....	10
4.5 Vision Guideウィンドウの構成 .....	11
4.5.1 タイトルバー .....	12
4.5.2 ツールバー .....	12
4.5.3 画像イメージ表示部 .....	17
4.5.4 Vision Guideウィンドウタブ .....	17
4.5.5 実行パネル .....	17
4.5.6 フローチャート .....	19

4.5.7 シーケンス, キャリブレーションツリー .....	20
4.6 Vision Guideウィンドウ .....	21
4.6.1 プロパティリスト ツールバーとボタン .....	21
4.6.2 シーケンスウィンドウ .....	22
4.6.3 オブジェクトウィンドウ .....	23
4.6.4 キャリブレーションウィンドウ .....	26
4.6.5 ジョグタブ .....	27
4.6.6 ロボットタブ .....	28
4.7 ビジョンシーケンスとオブジェクト .....	29
4.7.1 ビジョンシーケンスの概要 .....	29
4.7.2 ビジョンシーケンスの定義と一般的な知識 .....	29
4.7.3 ビジョンオブジェクトの概要 .....	30
4.7.4 ビジョンオブジェクトの定義 .....	31
4.8 プロパティとリザルト とは .....	32
4.8.1 ビジョンシーケンスのプロパティ .....	32
4.8.2 ビジョンシーケンスのリザルト .....	34
4.9 キャリブレーション .....	35
4.9.1 キャリブレーションについて .....	35
4.9.2 キャリブレーションの定義 .....	35
4.9.3 撮像歪み補正用の撮像パターン .....	36
<b>5. ビジョンシーケンス .....</b>	<b>37</b>
5.1 ビジョンシーケンスウィンドウ .....	37
5.1.1 ビジョンシーケンスの選択 .....	37
5.1.2 シーケンスウィンドウのプロパティリスト .....	37
5.1.3 シーケンスウィンドウのリザルトリスト .....	38
5.1.4 シーケンスのフローチャート .....	38
5.2 ビジョンシーケンスの新規作成 .....	39
5.2.1 概要 .....	39
5.2.2 ショートカット .....	39
5.2.3 ダイアログボックスのオプション .....	39
5.3 ビジョンシーケンスの削除 .....	40
5.3.1 概要 .....	40
5.3.2 ショートカット .....	40
5.3.3 ダイアログボックスのオプション .....	40
5.4 ビジョンオブジェクトの削除 .....	41
5.5 シーケンスステップ順序の変更 .....	41
5.6 ビジョンシーケンスの実行 .....	42
5.6.1 ビジョンシーケンスの選択 .....	42



5.6.2 ビジョンシーケンスパラメーター (プロパティ)の設定	42
5.6.3 ビジョンシーケンスのリザルト	42
5.6.4 シーケンスのフローチャート	42
5.6.5 ビジョンシーケンスの実行	42
5.6.6 シーケンスの複数回数実行	43
5.6.7 ビジョンシーケンス実行中の画像イメージ表示部	43
5.6.8 ビジョンシーケンスの中断	43
5.7 ビジョンシーケンスのテストとデバッグ	44
5.7.1 ビジョンオブジェクトの色の確認	44
5.7.2 個々のビジョンオブジェクトリザルトの確認	44
5.7.3 ビジョンシーケンスでの逐次ステップ実行	45
5.7.4 ビジョン統計機能の使用	45
5.7.5 RUNウィンドウとVision Guideウィンドウの切り替え	45
5.8 SPEL+からビジョンシーケンスを実行する方法	46
5.9 画像イメージの取り込み	46
5.9.1 画像イメージはいつ取り込まれるか	46
5.9.2 同一の画像イメージを2つ以上の異なるビジョンシーケンスで使用する 方法	47
5.9.3 イメージバッファの使用法	47
5.9.4 外部トリガーでのイメージ取り込みの使用法	48
5.9.5 カラー画像の使用法	49

## 6. ビジョンオブジェクト 50

6.1 ビジョンオブジェクトの基本事項	50
6.1.1 サーチウィンドウ	50
6.1.2 モデルウィンドウ	55
6.1.3 モデル原点	56
6.2 ビジョンオブジェクトの使い方	58
6.2.1 ImageOp オブジェクト (画像操作)	58
6.2.2 Geometricオブジェクト (幾何学的形状)	67
6.2.3 Correlationオブジェクト (相関サーチ)	94
6.2.4 Blob オブジェクト (2値検査)	122
6.2.5 Edge オブジェクト (エッジ検出)	141
6.2.6 Polarオブジェクト (ポーラサーチ)	150
6.2.7 OCRオブジェクト	162
6.2.8 CodeReaderオブジェクト	169
6.2.9 ColorMatch オブジェクト	175
6.2.10 LineFinder オブジェクト (直線検出)	182
6.2.11 LineInspector オブジェクト (直線検査)	190

6.2.12	ArcFinder オブジェクト (円弧/楕円弧検出).....	198
6.2.13	ArcInspectorオブジェクト (円弧/楕円弧検査).....	207
6.2.14	DefectFinder オブジェクト (差分検査) .....	216
6.2.15	Frame オブジェクト (フレーム) .....	226
6.2.16	Line オブジェクト (ライン).....	232
6.2.17	Point オブジェクト (ポイント).....	238
6.2.18	BoxFinder オブジェクト (長方形検出) .....	249
6.2.19	CornerFinder オブジェクト (コーナー検出).....	258
6.2.20	Contour オブジェクト (輪郭抽出) .....	267
6.2.21	Textオブジェクト .....	283
6.2.22	Decisionオブジェクト .....	288
6.2.23	Coordinatesオブジェクト.....	292
6.2.24	1つのオブジェクトから複数のリザルトを返す .....	296
6.2.25	自動マルチオブジェクト検索.....	302
6.2.26	すべてのビジョンオブジェクトラベルの表示、非表示の切り替え	308
6.2.27	すべてのビジョンオブジェクトグラフィックの表示 .....	308
6.2.28	カレントオブジェクトのみ表示 .....	309

## 7. ビジョンキャリブレーション 310

7.1	カメラ設置 .....	312
7.2	レンズの歪み、およびカメラチルト補正.....	313
7.3	基準点とカメラポイント.....	314
7.3.1	可動カメラ基準点 .....	316
7.3.2	下向き固定カメラ, スタンドアローンカメラ基準点.....	317
7.3.3	TwoRefPointsによる教示.....	317
7.4	キャリブレーション用ビジョンシーケンスの作成 .....	318
7.4.1	ターゲット1個を検出するビジョンシーケンス .....	318
7.4.2	ターゲット9個を検出するビジョンシーケンス .....	319
7.4.3	歪み補正に必要なビジョンシーケンス .....	320
7.4.4	ローカル設定, ツール設定, 増設アーム設定に必要な ビジョンシーケ ス.....	320
7.5	キャリブレーションGUI.....	321
7.5.1	新規キャリブレーションの作成 .....	321
7.5.2	キャリブレーションの削除.....	322
7.5.3	キャリブレーションプロパティとリザルト .....	322
7.5.4	歪み検出.....	326
7.5.5	ポイントティーチング .....	327
7.5.6	キャリブレーションボタン .....	328
7.5.7	[キャリブレーションの完了]ダイアログ .....	329

7.6	キャリブレーション手順	331
7.6.1	キャリブレーション手順: 可動カメラ	331
7.6.2	キャリブレーション手順: 下向き固定カメラ	341
7.6.3	キャリブレーション手順: 上向き固定カメラ	348
7.6.4	キャリブレーション手順: スタンドアローンカメラ	355
7.7	カメラを使用したローカル検出	357
7.7.1	作業面のローカル定義	357
7.8	可動カメラ設置位置の検出	363
7.8.1	カメラ設置位置のツール設定	363
7.8.2	カメラ設置位置のアーム設定	367
7.9	カメラを使用したツール設定	370
7.10	カメラを使用した3Dツール設定	374
<b>8.</b>	<b>ヒストグラムツール</b>	<b>380</b>
8.1	ヒストグラムの使用方法	380
8.2	ヒストグラムの例	381
8.2.1	Correlationオブジェクトで使うヒストグラム	382
8.2.2	Blobオブジェクトで使うヒストグラム	382
<b>9.</b>	<b>統計ツール</b>	<b>385</b>
9.1	ダイアログボックスオプションとその概要	386
9.2	ビジョンオブジェクトとサポートされる統計	387
9.3	サポートされるビジョンオブジェクトリザルト	388
9.4	SPEL+から使用可能なビジョンオブジェクト統計	389
<b>10.</b>	<b>チュートリアル</b>	<b>393</b>
10.1	クイックスタート: Vision Guide 7.0のチュートリアル	393
10.1.1	チュートリアルの概要	393
10.1.2	本チュートリアルを使用するにあたって必要となる事項	394
10.1.3	EPSON RC+ 7.0の起動と新規プロジェクトの作成	397
10.1.4	新規ビジョンシーケンスの作成	397
10.1.5	チュートリアルでのカメラレンズの構成	397
10.1.6	Blobオブジェクトを使ってパーツを検出する方法	399
10.1.7	ビジョンシーケンスを使用するSPEL+プログラムの作成	405
10.1.8	可動カメラでロボットをキャリブレーションするには	409
10.1.9	ビジョンガイダンスで使用するポイントの教示	414
10.1.10	ビジョンとロボットを使ってパーツへの移動をする方法	416

11. SPEL+でVision Guide 7.0を使う方法	419
11.1 概要	419
11.2 Vision Guide 7.0 SPEL+ コマンド	419
11.3 SPEL+からVRunコマンドでビジョンシーケンスを実行する方法	420
11.4 VGet, VSetコマンドで SPEL+のリザルトやプロパティにアクセスする方法	423
11.4.1 VGetの使用方法	424
11.4.2 VSetの使用方法	424
11.5 シーケンス名とオブジェクト名に変数を使用する方法	425
11.6 SPEL+でシーケンスリザルトを使用する方法	426
11.7 SPEL+言語から複数リザルトにアクセスする方法	426
11.8 マルチタスクでビジョンコマンドを利用する方法	427
11.9 ロボットでビジョンを使用する方法	428
11.9.1 位置リザルト	428
11.9.2 ツールの定義	429
11.9.3 ロボットが把持している基板でのツール計算	432
11.9.4 パレット検索用カメラの位置決め	433

# 1. マニュアル ヘルプ

## 1.1 本マニュアルの構成

### はじめに

本マニュアル全体の概要について説明します。EPSON RC+ 7.0の基本機能を理解していただくために、ヘルプ機能、安全機能、参考事例の使い方についても説明します。

### インストールのしかた

必要システム、Vision Guide 7.0の製品構成、ハードウェア、ソフトウェアのインストール手順について説明します。

### クイックスタート: ファーストVision Guide 7.0アプリケーション

Vision Guide 7.0を初めてお使いになる方のために、サンプルアプリケーションを使って説明します。ビジョンオブジェクトの新規作成から、Vision Guide 7.0 可動カメラのキャリブレーションの方法と、Vision Guide 7.0で検出されるパーツへのロボットの実際の動作までを、全体的に説明します。

### Vision Guideウィンドウ

Vision Guideウィンドウのレイアウトと、使用方法を説明します。また、Vision Guide 7.0のツールバー、画像イメージ表示部、Run(実行)パネル、オブジェクト、シーケンス、キャリブレーションタブについて説明します。

### ビジョンオブジェクト

Vision Guide 7.0で利用できるビジョンツールと、ツールの使用方法について説明します。

### ヒストグラムツールと統計ツール

ブローブオブジェクト、相関サーチオブジェクト、ポーラオブジェクトなどの、多様なタイプのビジョンオブジェクトを使った、ヒストグラムの方法について説明します。また、統計ダイアログをもつVision GuideウィンドウからのVision Guide統計ツールと、統計プロパティにアクセスするためのSPEL+言語からのVision Guide統計ツールについても説明します。

### ビジョンシーケンス

ビジョンシーケンスとは何か、その使用方法、応用方法について説明します。また、Vision Guideシーケンスのデバッグの方法についても説明します。

### キャリブレーション

様々なタイプのキャリブレーションを行う方法について説明します。

### SPEL+でVision Guide 7.0を使う方法

SPEL+言語からビジョンシーケンスを実行する方法と、ビジョンオブジェクトのプロパティとリザルトへのアクセス方法について説明します。また、ロボットのガイダンスに、Vision Guide 7.0のリザルトを使用する方法について説明します。

### 1.2 関連マニュアル

Vision Guide 7.0ご使用時は、本マニュアルの他に、次のマニュアルを参照してください。

#### Vision Guide 7.0 ハードウェア & セットアップ

Vision Guideのハードウェアとセットアップ方法について、適切な使用方法、使用上の注意、警告などの説明があります。

#### Vision Guide 7.0 プロパティ & リザルトリファレンス

ビジョンシーケンスやビジョンオブジェクトに利用できるプロパティとリザルトについてのマニュアルです。各プロパティやリザルトについて、適切な使用方法、使用上の注意、警告などの説明があります。

#### EPSON RC+ 7.0 ユーザーズガイド

エプソンロボットコントロールシステムの使用法について説明します。

#### SPEL+ ランゲージリファレンス

SPEL+言語の命令について説明します。

### 1.3 オンラインヘルプの使用方法

EPSON RC+ 7.0は、オンラインヘルプを参照できるシステムを持っています。従来のマニュアルを使う方法に比べ、簡単に知りたい情報を調べることができます。

EPSON RC+ 7.0でヘルプを参照するには、次のようないくつかの方法があります。

- F1ファンクションキーを押すと、ヘルプを参照できます。作業中の場合は、作業中の項目についてヘルプ情報が表示されます。画面、あるいはダイアログ中で理解できない項目を知りたいときに便利です。プログラムを編集している場合は、カーソル位置にあるSPEL+言語のキーワードについて、ヘルプ情報が表示されます。このヘルプ情報では、SPEL+言語を使用するための構文説明を参照することができます。
- <ヘルプ>ボタンのあるダイアログボックスでは、<ヘルプ>ボタンをクリックして参照できます。
- 目次を参照し、トピックを選択するには、ヘルプメニューから目次を選択してください。トピックは、グリーンで強調表示されアンダーラインがひかれているテキストをクリックすることによって選択します。(この操作によって、目的のトピックにジャンプすることができます。)
- ヘルプメニューから目次を選択し<S>キーを押すか、<検索>ボタンを押すと、特定のトピックに関する情報を検索できます。

ヘルプシステム中には、強調表示され、アンダーラインがひかれている項目があります。これらは、ハイパーテキストリンクであることを示しており、この強調表示テキストをクリックすると、ヘルプシステム内で強調表示されたテキストに関連のある領域に、システムが自動的にジャンプします。いくつかのテキストは、グリーンに強調表示されてテキストの下に(点線)がついている場合があります。このようなテキストをクリックすると、ポップアップウィンドウが表示されて、強調表示されたテキストについてさらに詳細な説明を得ることができます。また、それに関連する情報の参照先が示される場合もあります。




本マニュアルに記載していることの大半は、Vision Guide 7.0のヘルプシステムでも参照することができます。ただし、記載の順番は多少異なります。Vision Guide 7.0のヘルプシステムの方が、ハイパーテキストリンクで適切に関連情報を参照できるため、利用しやすくなっています。

## 2. 安全




ご使用になる前に、本マニュアルをお読みいただき、正しくお使いください。  
お読みになった後は、いつでも取り出せる所に保管し、不明な点があったら再読してください。

### 2.1 本文中の記号について

下記のマークを用いて、安全に関する注意事項を記載しています。必ずお読みください。

 警告	この表示を無視して誤った取り扱いをすると、人が死亡、または重傷を負う可能性が想定される内容を示しています。
 警告	この表示を無視して誤った取り扱いをすると、人が感電により、負傷する可能性が想定される内容を示しています。
 注意	この表示を無視して誤った取り扱いをすると、人が傷害を負う可能性が想定される内容および物的損害のみの発生が想定される内容を示しています。

### 2.2 安全に関する注意事項

 警告	<ul style="list-style-type: none"> <li>■ 本製品を、安全を確保する目的には使用しないでください。</li> <li>■ 本製品はマニュアルに記載された使用条件でお使いください。 使用条件を満たさない環境での使用は、製品寿命を短くするばかりでなく、重大な安全上の問題を引き起こす可能性があります。</li> </ul>
 注意	<ul style="list-style-type: none"> <li>■ カメラ、カメラケーブルは、当社販売元から購入してください。 当社販売元以外から購入されたものは、保証対象外となります。</li> </ul>
 注意	<ul style="list-style-type: none"> <li>■ 画像処理の結果は、光源の変動や外らん光、および周辺の電磁ノイズなどの影響を受け正しい結果が得られない場合があります。 画像処理の結果に基づきロボットを動作させる場合など、カメラ視野内の全領域に対してロボットが動作する可能性を考慮し、ロボットシステムを設計してください。</li> </ul>

### 2.3 ロボットの安全

ロボットや他の自動装置を動作させているときは、安全を最優先してください。コントローラー/EPSON RC+ 7.0には、多くの安全機能が組み込まれています。E-Stopライン、ロボットポーズ入力、安全ガード入力、様々な出入力システムなどの方法がお使いいただけます。これらの安全機能は、ロボットセルを設計する場合に使用してください。

安全情報とガイドラインについては、EPSON RC+ 7.0 ユーザーズガイド「安全について」を参照してください。



## 3. はじめに

### 3.1 Vision Guide 7.0の概要

#### 3.1.1 Vision Guide 7.0の目的

Vision Guide 7.0は、多様なマシンビジョンのアプリケーションに使用でき、また、ロボットガイダンスアプリケーションに必要なビジョンツールを提供することを主な目的としています。Vision Guide 7.0の一般的なアプリケーションとして、部品の位置決めと移動、部品検査、寸法計測などがあります。移動とビジョンの両方を組み込む必要のあるアプリケーションのほとんどに、Vision Guide 7.0とエプソンロボットを使用することができます。

Vision Guide 7.0と他のビジョンシステムとの主な違いは、Vision Guide 7.0が統合ロボットビジョンシステムとして設計されているという点です。エプソンロボットを使用することにより、Vision Guide 7.0システムに組み込まれたカメラキャリブレーション手順などの多くの機能をお使いいただけます。ビジョンを使用したアプリケーションを完了する時間を大幅に短縮します。

#### 3.1.2 Vision Guide 7.0の特徴

Vision Guide 7.0に装備されている基本的な特徴 ツールは、次のとおりです。

- 多様なカメラ方向と、キャリブレーションをサポートするツールを装備しています。キャリブレーションは、他のロボットビジョンシステムよりも簡単で、高精度です。
- GUI操作により簡単にプロトタイプを作成できます。ここで作成したプロトタイプビジョンシーケンスを、最終アプリケーションに使用することもできます。
- EPSON RC+ 7.0プログラミングと開発環境に完全統合できます。
- 2値検査ツールは、オブジェクトのサイズ、形状、位置をさまざまに計測できます。穴を数えるツールや与えられたあいまいな形状の丸みを示すツールもあります。
- Geometricサーチツールは、幾何学的形状の対象物に基づいてモデルを検出します。
- 正規化相関サーチツールは、テンプレートマッチング先端技術を利用して、多様な光条件の下でオブジェクトの位置決めします。
- エッジ検出ツールは、サブピクセル精度で、特定エッジの位置決めします。
- LineFinderツール、ArcFinderツールは、サブピクセル精度で、簡単に直線や円弧のエッジを検出でき、正確な位置決めを可能にします。
- ポーラーサーチツールは、複雑なオブジェクトの回転を迅速に計測する高速の角度検索ツールです。このツールは、ビジョンガイダンスロボットのピックアンドプレイスアプリケーションに非常に有益です。
- DefectFinderツールは、テンプレート画像との差分検査を実施し、画像上の欠陥検出を可能にします。
- ライン ポイントツールは、複数の点を結ぶ線を作成するメカニズムと、測定能力を持っています。
- フレームツールは、すべてのビジョンツールを「リファレンスフレーム」に基づいてダイナミックに位置決めをすることができます。

- オブジェクトの参照メカニズムは、あるビジョンツールの位置決めをするときに、他のビジョンツールの結果を基準とすることができます。したがって、開発時間を節約することができます。
- ヒストグラムチャートは、ピクセルデータをより詳細に見るだけでなく、それを必要とするツールの正常範囲を設定するのにも、強力なメカニズムとなります。
- 統計演算は、平均、標準偏差、範囲、最小値、最大値、その他の統計が組み込まれていて、設計時または実行時に各ビジョンツールに参照させることができます。
- カメラレンズ、カメラ、ロボット周辺部品の、角度のずれの自動補正機能を装備しています。
- ColorMatchツールやColorFilter操作など、カラーカメラ、カラー画像をサポートします。

## 3.2 前提となるEPSON RC+ 7.0の基本的知識

Vision Guide 7.0は、EPSON RC+ 7.0環境をコアとしたオプションです。Vision Guide 7.0をご利用いただくためには、EPSON RC+ 7.0の開発環境、エプソンロボットについての知識が必要です。本マニュアルの内容は、次の事項について知識がある方を対象として説明しています。

- EPSON RC+ 7.0プロジェクト管理の概念と使用方法
- EPSON RC+ 7.0でSPEL+プログラムを作成し、編集する方法
- RunウィンドウからSPEL+プログラムを実行する方法
- SPEL+の基本的な言語構造や機能、使用方法など

EPSON RC+ 7.0を初めて使用する方は、当社で行っている導入トレーニングを必ず受講してください。

## 4. Vision Guideの概要

### 4.1 概要

ここでは、Vision Guide 7.0とそのコンポーネントを完全に理解していただくために、いくつかの概念と定義について説明します。本章は、次の項目で構成されています。

- Vision Guide 7.0を使用するために理解していただきたい基本的な定義
- EPSON RC+ 7.0からVision Guideウィンドウを開く方法
- Vision Guide 7.0画像表示についての説明
- すべてのビジョンオブジェクトの説明と、それを操作するツールバーのボタンについての説明
- プロパティとリザルトの説明
- シーケンスウィンドウ、オブジェクトウィンドウ、キャリブレーションウィンドウの説明
- ビジョンオブジェクト、ビジョンシーケンスの実行方法と使用例

### 4.2 Vision Guide 7.0を使用するために必要な基本的概念

本章の内容を理解していただくために、いくつかの基本的な概念を簡単に説明します。

#### 4.2.1 ビジョンシーケンス

ビジョンシーケンスとは、特定の順番でビジョンオブジェクトをグルーピングしたもので、Vision Guideウィンドウ、またはSPEL+言語から実行することができます。

ビジョンシーケンスは、ビジョンシーケンスの実行を設定するのに使われる特定のプロパティを備えています。例えば、Cameraプロパティは、どのカメラでビジョンシーケンスのイメージを取り込むかを定義し、RuntimeAcquireプロパティは、ビジョンシーケンスにイメージを取り込む方法を定義します。

ビジョンシーケンスは、特定のビジョンプロセス、あるいはプロセスの一部を処理するのに必要なビジョンオブジェクトのすべてを含む、コンテナのようなものと考えることができます。一般的に、ビジョンシーケンスは、すべてのビジョンプロセスの開始点となります。

#### 4.2.2 ビジョンオブジェクト

ビジョンオブジェクトとは、カメラにより取り込まれたイメージに適用できるビジョンツールです。

ビジョンオブジェクトに装備されているオブジェクトには、幾何学相関サーチ、相関サーチ、2値検査、ポーラサーチ、エッジ検出、直線サーチ、円弧サーチ、差分検査、ライン、ポイント、フレームなどがあります。

すべてのビジョンオブジェクトは、実行(現在のイメージに適用)後に、ビジョンオブジェクトの実行に要した時間、位置情報、角度情報、ビジョンオブジェクトの検出の有無、オブジェクトの合否、などのリザルトを返します。

ビジョンオブジェクトは、ビジョンオブジェクトの動作特性の定義に使用されるプロパティと、ビジョンオブジェクト実行後に返される値のリザルトを持っています。

### 4.2.3 プロパティ

プロパティとは、ビジョンオブジェクト、またはビジョンシーケンスに設定されたパラメーターと考えることができます。

ビジョンプロパティは、ビジョンアプリケーションを素早く作成しテストする方法を提供するVision Guideウィンドウから、ポイントクリックの操作で設定することができます。

ビジョンプロパティは、SPEL+言語から設定したり、設定値のチェックをすることができ、実行時にビジョンオブジェクトのダイナミックな修正が必要とされたときでも柔軟に対応できます。

ビジョンシーケンスプロパティとビジョンオブジェクトプロパティは、非常にパワフルです。なぜなら、これらは簡単に修正できるばかりでなく、ビジョンオブジェクトを理解しやすく、使用しやすくするからです。それと同時に、より複雑なアプリケーションに求められる柔軟性があります。

### 4.2.4 リザルト

リザルトとは、ビジョンオブジェクト、またはビジョンシーケンスの実行後に返される値です。通常用いられるリザルトの例は、次のとおりです。

**Time**           ビジョンオブジェクト、またはビジョンシーケンスの実行に要した時間を返します。

**RobotXYU**   ロボット座標で検出された特徴のX, Y, Uの位置を返します。

**Found**       ビジョンオブジェクトが検出されたかどうかを返します。

**Passed**     ビジョンオブジェクトの検出結果がOKかNGかを返します。

ビジョンリザルトは、Vision Guideウィンドウで、シーケンスウィンドウとオブジェクトウィンドウで見ることができます。SPEL+プログラムでは、ビジョンリザルトを使うこともできます。

### 4.2.5 実行時ビジョンコマンド

ロボットの動作とビジョンガイダンスのシームレスな統合のために、SPEL+言語には一連のビジョンコマンドが追加されています。

VRunコマンドは、1つのファンクションの呼び出しでSPEL+言語からビジョンシーケンスを起動することができます。

VGetコマンドは、ビジョンオブジェクト、ビジョンシーケンス、キャリブレーションから返されたリザルトを得ることができます。

ビジョンコマンドは、Vision Guide 7.0のポイントクリックの操作による開発環境で作成、修正、保守することができ、しかも、Vision Guide 7.0で作成されたコマンドはすべてSPEL+言語からもアクセスできるため、非常にパワフルです。

#### 必要なビジョンハードウェア

Vision Guide 7.0を使用するには、以下のいずれかのハードウェアが必要になります。

- コンパクトビジョン CV1 (Ver. 2.1.0.0 以降のファームウェア)
- コンパクトビジョン CV2 (Ver. 2.3.0.0 以降のファームウェア)
- PC ビジョン PV1

各ハードウェアの詳細は、「Vision Guide 7.0 ハードウェア&セットアップ編」を参照してください。

### 4.2.6 EPSON RC+ 7.0プロジェクトでのVision Guide 7.0

EPSON RC+ 7.0 は、ロボットアプリケーションに必要とされる、すべてのプログラム、ティーチングポイント、ロボット設定のコンテナであるプロジェクトをベースとしています。

このプロジェクトの構成により、複数のプロジェクトで1台のロボットを使用することや、過去の作業アプリケーションを破棄することなく、新規アイデアをチェックする環境テストなどが簡単に実行できます。

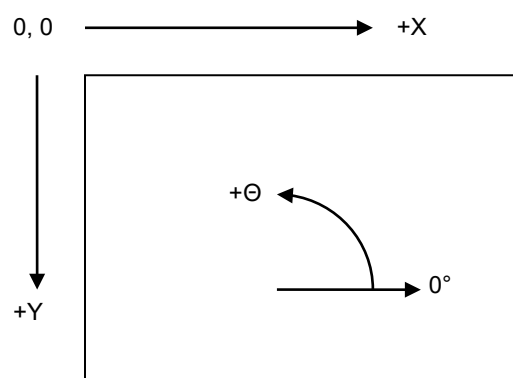
Vision Guide 7.0を含むEPSON RC+ 7.0アプリケーションを作成するとき、アプリケーションに必要な、すべての関連ビジョンシーケンスとビジョンオブジェクトは、プロジェクトに通常含まれるその他の項目と同様、プロジェクト内に保存されます。これにより、既存のプロジェクトを開く時には、プロジェクトに関連するすべてのものが利用可能になります。

## 4.3 座標系

このセクションでは、Vision Guide 7.0で使用する座標系について説明します。

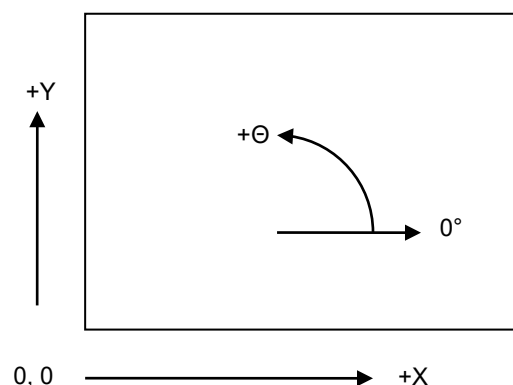
### 4.3.1 イメージ画像座標系

カメライメージ画像とイメージ画像表示で使用する座標系です。単位はピクセルです。



### 4.3.2 カメラ座標系

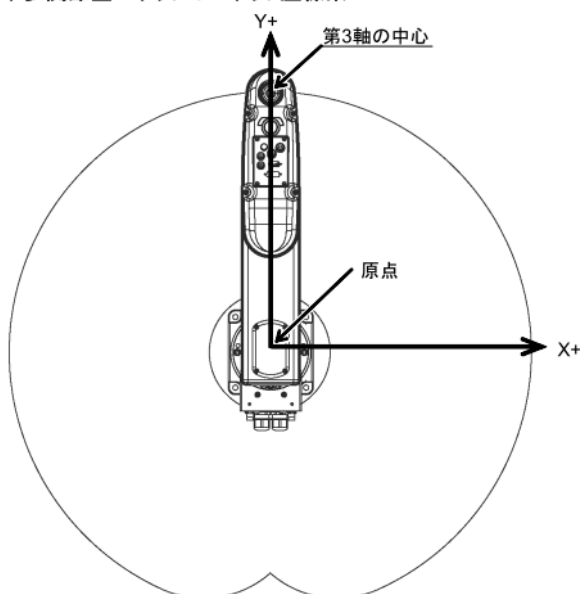
カメラ座標系は、カメラ視野内の座標です。単位はmmです。この座標系で座標を取得するためにはカメラはキャリブレーションされなければなりません。



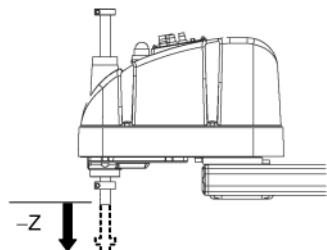
### 4.3.3 ロボット座標系

ロボット座標系は、ロボットが持っている座標です。単位は **mm** です。この座標系で座標を取得するためには、カメラはロボット座標でキャリブレーションされなければなりません。ロボット座標系の詳細については、EPSON RC+ 7.0 ユーザーズガイドを参照してください。

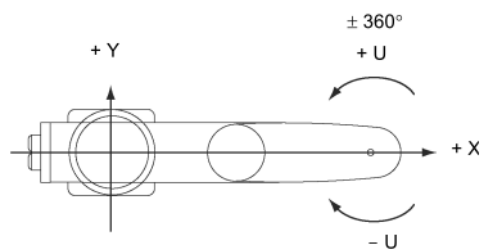
水平多関節型ロボットのロボット座標系



ロボット座標系の Z 軸




ロボット座標系の U 軸



## 4.4 Vision Guide ウィンドウを開く

Vision Guide ウィンドウは、EPSON RC+ 7.0 開発環境で開かれます。

EPSON RC+ 7.0 を起動した後で、Vision Guide ウィンドウは次の2つの方法で開くことができます。

- ツールバーから : EPSON RC+ 7.0 のツールバーには、 <ビジョン> ボタンがあります。このボタンをクリックすると、Vision Guide ウィンドウが表示されます。
- ツールメニューから : ツールメニュー - [ビジョン] を選択すると、Vision Guide ウィンドウが表示されます。

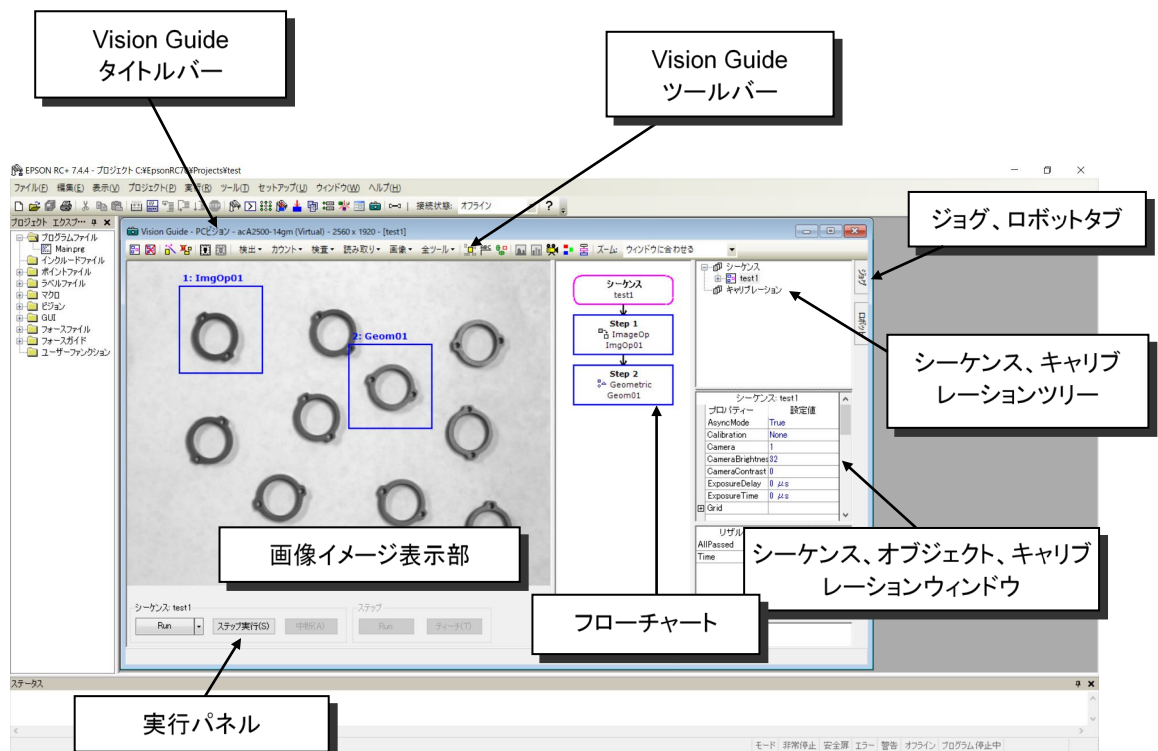
Vision Guide ウィンドウが開いたら、Vision Guide 7.0 を使用できます。次の数ページにわたって、Vision Guide ウィンドウの基本的な構成について説明します。

## 4.5 Vision Guideウィンドウの構成

Vision Guideウィンドウは、ポイント クリックの操作環境で、ほとんどのビジョンを開発することができます。

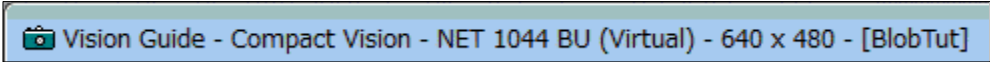
Vision Guide 7.0を使うために、このウィンドウと主な構成について理解することが必要です。Vision Guideウィンドウは、主に次の部分で構成されています。

- タイトルバー
- ツールバー
- フローチャート
- シーケンス, キャリブレーションツリー
- 画像イメージ表示部
  - シーケンス, オブジェクト, キャリブレーションウィンドウ
- ジョグ, ロボットタブ
- 実行パネル



### 4.5.1 タイトルバー

Vision Guideウィンドウには、ウィンドウのタイトル、カメラのタイプと解像度、現在のシーケンス名などがついた、タイトルバーがあります。ビジョンシーケンス名は、角カッコ内に入っています。タイトルバーの例を下に示します。これは、コンパクトビジョンを使用して Vision Guide 7.0を実行中で、現在“BlobTut”ビジョンシーケンスを使用していることを示しています。



EPSON RC+ 7.0環境のメインタイトルバーと、Vision Guideタイトルバーは異なりますので注意してください。

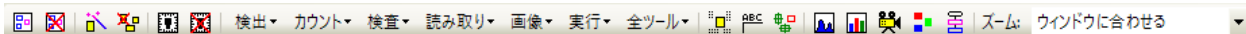
基本的な違いは、EPSON RC+ 7.0タイトルバーには、作業中のプロジェクトの名前が表示され、Vision Guideタイトルバーには、ビジョンを実行するために作業中のシーケンス名が表示されるということです。

### 4.5.2 ツールバー

ツールバーは、通常、ほとんどのMicrosoft Windows™アプリケーションに装備されており、これを使って、Microsoft Windows™製品の共通機能に素早くアクセスすることができます。

EPSON RC+ 7.0とVision Guide 7.0では、このツールバー機能が十分に活用されています。

Vision Guideツールバーは、Vision Guideウィンドウの上方、タイトルバーの真下にあり、次のように表示されています。

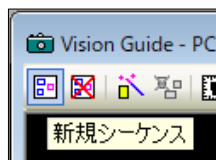


Vision GuideウィンドウのVision Guideツールバーボタンは、見つけやすく、また使いやすくするために、グループ別に分かれています。ツールバーボタンの各々のグループについて、簡単に説明します。

- 左端の2つのボタンのグループは、ビジョンシーケンスの作成と削除をするのに使います。
- 次の2つのツールバーボタンは、ステップウィザードの呼び出しとオブジェクトの削除をします。
- 次の2つのツールバーボタンは、キャリブレーションを作成、削除をします。
- 次のツールバーボタンのグループは、ビジョンオブジェクトのカテゴリ(検出, カウント, 検査, 読み取り, 画像, 全ツール)を選択するのに使います。
- 次のツールバーボタンのグループは、Vision Guide 7.0環境のユーティリティとして考えることができます。カレントオブジェクトのみ表示、オブジェクトの削除、すべてのラベルの表示オフ、グラフィックの表示のボタンです。
- 次の2つのツールバーボタンは、ヒストグラムダイアログ、統計ダイアログの表示をします。
- 右端の2つのボタンは、ライブビデオと固定ビデオの切り替え、また、カラーとグレースケール画像の切り替えを行います。
- 右端のツールバーボタンは、フローチャートの表示のオン、オフを行います。



Vision Guideツールバーには、各ツールバーボタンに、下の例のようなツールチップが備わっています。



Vision Guide 7.0(およびEPSON RC+ 7.0)の各ツールバーボタンの用途がわからないときは、そのツールバーボタンの上にしばらくマウスポインターを置いてください。約2秒後にツールバーボタンのツールチップが現われます。

次に、Vision Guide 7.0の各ツールバーボタンについて簡単な説明を示します。

#### ボタン ツールチップ: 簡単な説明



**新規シーケンス:** ビジョンシーケンスを作成します。

ダイアログが表示され、新規シーケンス名を入力するように要求します。



**シーケンス削除:** 現在のプロジェクトのビジョンシーケンスを削除します。

現在のプロジェクトにビジョンシーケンスがない場合、ボタンがグレースアウト表示(選択不可)になります。



**新規ステップ:** このボタンを押すと、ステップウィザードを表示します。

ステップウィザードを使用することで、オブジェクトを追加することができます。

メニューバーからの追加と異なり、ステップウィザードからの追加では、追加する前にオブジェクト名の設定とシーケンス内でのオブジェクトの位置を設定することができます。



**オブジェクト削除:** 現在アクティブのビジョンオブジェクトを削除します。

ビジョンオブジェクトを削除するには、削除したいビジョンオブジェクトを選択し、このボタンをクリックしてください。

現在のプロジェクトにビジョンシーケンスがない場合、または現在のシーケンスにビジョンオブジェクトが定義されていない場合、ボタンがグレースアウト表示(選択不可)になります。



**新規キャリブレーション:** キャリブレーションを作成するキャリブレーションダイアログを開きます。現在のプロジェクトにビジョンシーケンスがない場合、ボタンは無効になります。



**キャリブレーション削除:** キャリブレーションを削除するキャリブレーション削除ダイアログを開きます。

現在のプロジェクトにビジョンキャリブレーションがない場合、ボタンは無効になります。

## ボタン ツールチップ: 簡単な説明



**カレントオブジェクト表示:** このボタンをクリックすると、現在選択されているオブジェクトのみ表示されます。1つのオブジェクトのみで作業したい場合に便利です。



**全ラベル非表示:** このボタンを押すと、ビジョンオブジェクトのラベルの表示がオフになります。

画面上に複数のオブジェクトが存在する場合、ラベルが表示されていると、かえってオブジェクトの判別が難しくなることがあります。その場合、この機能が役に立ちます。現在のプロジェクトにビジョンシーケンスがない場合、ボタンがグレースアウト表示(選択不可)になります。



**全グラフィック表示:** このボタンをクリックすると、ビジョンオブジェクトのすべてのグラフィック (サーチウィンドウ, モデル原点, モデルウィンドウ, 直線, ラベル) が表示されます。

このボタンは、個々のビジョンオブジェクトのグラフィックプロパティの設定を無視するので、個々のビジョンオブジェクトのグラフィックプロパティを修正するよりも速く、すべてのビジョンオブジェクトを表示したいときに便利です。



**ヒストグラム:** このボタンをクリックすると、ヒストグラムダイアログが表示されます。現在のプロジェクトにビジョンシーケンスがない場合、または、現在のシーケンスにビジョンオブジェクトがない場合、ボタンがグレースアウト表示(選択不可)になります。



**統計:** このボタンをクリックすると、統計ダイアログが表示されます。現在のプロジェクトにビジョンシーケンスがない場合、または、現在のシーケンスにビジョンオブジェクトがない場合、ボタンがグレースアウト表示(選択不可)になります。



**画像入力停止:** ライブ画像と固定画像を切り替えます。



**カラー/白黒:** カラーとグレースケール表示を切り替えます。



**フローチャート:** フローチャートの表示のオン、オフを切り替えます。



**検出:** 検出カテゴリのオブジェクト (Geometric, Correlation, Blob, Edge, Polar, Arc Finder, Line Finder, Box Finder, Corner Finder, Frame, Line, Point, Contour, Coordinates)を選択できます。



**カウント:** カウントカテゴリのオブジェクト (Blob, Correlation, Geometric)を選択できます。



**検査:** 検査カテゴリのオブジェクト (Blob, Defect Finder, Line, Line Inspector, Arc Inspector, Color Match)を選択できます。



**読み取り:** 読み取りカテゴリのオブジェクト (Code Reader, OCR)を選択できます。



**画像:** 画像カテゴリのオブジェクト (ImageOp, Text)を選択できます。



**実行:** 実行カテゴリのオブジェクト(Decision)を選択できます。



**全ツール:** すべてのオブジェクトを選択できます。

## ツールバーのカテゴリから呼び出されるボタン

## ボタン ツールチップ: 簡単な説明

**ImageOp(画像操作):** ImageOpオブジェクトを作成します。

現在のプロジェクトにビジョンシーケンスがない場合、ボタンがグレースアウト表示(選択不可)になります。

**Geometric(幾何学的形状):** Geometricオブジェクトを作成します。

現在のプロジェクトにビジョンシーケンスがない場合、ボタンがグレースアウト表示(選択不可)になります。

**Correlation(相関サーチ):** Correlationオブジェクトを作成します。

現在のプロジェクトにビジョンシーケンスがない場合、ボタンがグレースアウト表示(選択不可)になります。

**Blob(2値検査):** Blobオブジェクトを作成します。

現在のプロジェクトにビジョンシーケンスがない場合、ボタンがグレースアウト表示(選択不可)になります。

**Edge(エッジ検出):** Edgeオブジェクトを作成します。

現在のプロジェクトにビジョンシーケンスがない場合、ボタンがグレースアウト表示(選択不可)になります。

**Polar(ポーラサーチ):** Polarオブジェクトを作成します。

現在のプロジェクトにビジョンシーケンスがない場合、ボタンがグレースアウト表示(選択不可)になります。

**OCR:** OCRオブジェクトを作成します。

現在のプロジェクトにビジョンシーケンスがない場合、ボタンがグレースアウト表示(選択不可)になります。

OCRのライセンスがない場合、ボタンがグレースアウト表示(選択不可)になります。

**Code Reader:** CodeReaderオブジェクトを作成します。

現在のプロジェクトにビジョンシーケンスがない場合、ボタンがグレースアウト表示(選択不可)になります。

**Color Match:** ColorMatchオブジェクトを作成します。

現在のプロジェクトにビジョンシーケンスがない場合、ボタンがグレースアウト表示(選択不可)になります。

**Box Finder(長方形検出):** BoxFinderオブジェクトを作成します。

現在のプロジェクトにビジョンシーケンスがない場合、ボタンがグレースアウト表示(選択不可)になります。

**Corner Finder(コーナー検出):** CornerFinderオブジェクトを作成します。

現在のプロジェクトにビジョンシーケンスがない場合、ボタンがグレースアウト表示(選択不可)になります。

**Line Finder(直線検出):** LineFinderオブジェクトを作成します。

現在のプロジェクトにビジョンシーケンスがない場合、ボタンがグレースアウト表示(選択不可)になります。

**Line Inspector(直線検査):** LineInspectorオブジェクトを作成します。

現在のプロジェクトにビジョンシーケンスがない場合、ボタンがグレースアウト表示(選択不可)になります。

##### ボタン ツールチップ: 簡単な説明

---



**ArcFinder(円弧/楕円弧検出):** ArcFinderオブジェクトを作成します。  
現在のプロジェクトにビジョンシーケンスがない場合、ボタンがグレースアウト表示(選択不可)になります。



**Arc Inspector(円弧/楕円弧検査):** ArcInspectorオブジェクトを作成します。  
現在のプロジェクトにビジョンシーケンスがない場合、ボタンがグレースアウト表示(選択不可)になります。



**Defect Finder(差分検査):** DefectFinderオブジェクトを作成します。  
現在のプロジェクトにビジョンシーケンスがない場合、ボタンがグレースアウト表示(選択不可)になります。



**Frame(フレーム):** Frameオブジェクトを作成します。  
現在のプロジェクトにビジョンシーケンスがない場合、ボタンがグレースアウト表示(選択不可)になります。



**Line(ライン):** Lineオブジェクトを作成します。  
現在のプロジェクトにビジョンシーケンスがない場合、ボタンがグレースアウト表示(選択不可)になります。



**Point(ポイント):** Pointオブジェクトを作成します。  
現在のプロジェクトにビジョンシーケンスがない場合、ボタンがグレースアウト表示(選択不可)になります。



**Contour(輪郭抽出):** Contourオブジェクトを作成します。  
現在のプロジェクトにビジョンシーケンスがない場合、ボタンがグレースアウト表示(選択不可)になります。



**Text:** Textオブジェクトを作成します。  
現在のプロジェクトにビジョンシーケンスがない場合、ボタンがグレースアウト表示(選択不可)になります。



**Decision:** Decisionオブジェクトを作成します。  
現在のプロジェクトにビジョンシーケンスがない場合、または、シーケンスのオブジェクトがない場合、ボタンがグレースアウト表示(選択不可)になります。



**Coordinates:** Coordinatesオブジェクトを作成します。  
現在のプロジェクトにビジョンシーケンスがない場合、ボタンがグレースアウト表示(選択不可)になります。

### 4.5.3 画像イメージ表示部

画像イメージ表示部は、Vision Guideウィンドウ中の、Vision Guideツールバーの下にある、カメラで見たイメージを表示する領域です。

また、画像イメージ表示部は、イメージ処理に使われる各々のグラフィックを表示します。例えば、画像イメージ表示部では、種々の位置に表示される複数のボックスや十字線が見えます。これらのグラフィックによって、ビジョンオブジェクトツールを使った開発作業が楽になります。

Vision Guideウィンドウと画像イメージ表示部は、4.5 Vision Guideウィンドウの構成 に記載しています。

Vision Guide GUIのウィンドウサイズを大きくすると、画像イメージ表示も合わせて大きくなります。Runダイアログの画像表示も同様に、ウィンドウサイズに合わせて画像イメージ表示のサイズが変わります。画像イメージ表示部とタブグループの間にはスプリッターバーがあり、画像表示のサイズを調整できます。画像表示のサイズを変更しても、画像のアスペクト比は保たれます。

### 4.5.4 Vision Guideウィンドウタブ

タブは、関連データやグループになっているデータへのアクセスを迅速に行うための機能です。タブは、メニュー項目や複数ウィンドウに比べ、はるかに簡単にデータへアクセスできます。

Vision Guideウィンドウでは、タブを使用することによって、システムが簡単に学習、記憶できるような、シングルウィンドウの開発環境となります。

Vision Guideウィンドウの右端には2つのタブセットがあり、それぞれ「ジョグ」、「ロボット」というラベルがついています。これらのタブの位置は、シーケンスツリーの隣に固定されていて、画像イメージ表示部の外でタブを使うことができます。

これら2つのタブは、共通してVision Guide 7.0開発環境で使われ、また、Vision Guide 7.0についての知識や使用方法に関するヘルプに簡単にアクセスすることができます。各タブの詳細は、本章で説明します。

### 4.5.5 実行パネル

実行パネルは、画像イメージ表示部の下にあります。実行パネルの目的は、Vision Guide 7.0開発環境からシーケンスを実行することと、デバッグすることです。実行パネルは次のように表示されます。



#### シーケンスの実行

ビジョンシーケンスを作成し、ビジョンオブジェクトを追加したら、実行パネルの左側にある、<Run>ボタンをクリックし、ビジョンシーケンスを実行します。

#### サイクル数

<Run>ボタン右側の▼ボタンをクリックし、<サイクル数設定>からサイクル数を指定できます。ここで設定したサイクル数の回数だけシーケンスが実行されます。

### 中断

ビジョンシーケンスを中止したいときは、実行パネルの<中断>ボタンをクリックします。<中断>ボタンは、シーケンスが実行中のときだけ有効です。<中断>ボタンは、ストロボトリガーを待つシーケンスに対しても使用できます。

### ステップ実行

<ステップ実行>ボタンをクリックすると、ビジョンシーケンスの1つのステップを実行できます。各ステップの実行で、1つのビジョンオブジェクトを実行します。

次のステップ実行を行うには、そのたびに<ステップ実行>ボタンをクリックしてください。<ステップ実行>ボタンを初めてクリックするとき、ビジョンオブジェクトは青い点線に囲まれてステップモードがアクティブでない状態です。

<ステップ実行>ボタンを次にクリックするときは、シーケンスの最初のオブジェクトが実行され、それから2番目以降のオブジェクトが最後まで実行されます。ステップ実行を始めに戻すには、実行パネル左側の<Run>ボタンをクリックして、ビジョンシーケンス全体を実行してください。

ビジョンシーケンスの実行中に、特定のオブジェクトのリザルト値を確認することができます。

フローチャート、またはシーケンスツリーを使用し、確認したいビジョンオブジェクトを選択してください。ビジョンオブジェクトの選択後、実行パネル左側にある<Run>ボタンをクリックしてください。ビジョンシーケンスが実行を開始します。

フローチャート、またはシーケンスツリーからオブジェクトを選択したときには、選択した特定のビジョンオブジェクトのリザルト値を見ることができます。

### NOTE



実行パネル左側にある<Run>ボタンを使用すると、SPEL+言語からの実行を必要とせず、ビジョンシーケンスの全体を迅速にテスト、デバッグすることができます。

また、ビジョンシーケンスを変更してテストし、そのリザルトが必要でないときは、前回に保存したビジョンシーケンスに戻ることができます。

このために、実行パネル左側にある<Run>ボタンによって、ビジョンシーケンスの実行のたびにビジョンシーケンスが自動的に保存されるようなことはありません。しかし、Runウィンドウからビジョンシーケンスを実行したときには、自動保存が有効になっていれば、ビジョンシーケンスはプロジェクトの残りの項目と一緒に自動的に保存されます。

### オブジェクト実行

ビジョンシーケンスを作成し、ビジョンオブジェクトを追加したら、実行パネルの右から2番目にある、<Run>ボタンをクリックし、ビジョンオブジェクトを実行します。

### ティーチ

作成したビジョンオブジェクトがティーチを必要とする場合、<ティーチ>ボタンをクリックし、オブジェクトのティーチウィンドウ部分のティーチを行います。

ティーチを必要としないオブジェクトの場合は<ティーチ>ボタンはクリックできません。

### キャリブレーション

シーケンスツリーからキャリブレーションプロジェクトを選択した場合、実行パネルは<ポイントティーチング>ボタンと<キャリブレーション実行>ボタンのみになります。

<ポイントティーチング>ボタンをクリックするとポイントティーチウィンドウが表示されティーチを実行することができます。

ポイントティーチが完了した状態で<キャリブレーション実行>ボタンをクリックすることでキャリブレーションを実行します。ポイントティーチを行っていない状態では<キャリブレーション実行>ボタンはクリックできません。

#### 4.5.6 フローチャート

フローチャートは、画像イメージ表示部の右側に表示されています。フローチャートは、選択しているシーケンスのオブジェクトの処理の流れを表しています。フローチャートは、先頭のフローが現在選択されているシーケンスを表し、2番目からはそのシーケンスに含まれるオブジェクトを表しています。オブジェクトの順番は、実行される順番になっています。

フローチャートは、通常、フローの枠が青色で表示されています。選択したいオブジェクトのフローをクリックすると、そのオブジェクトのフローのみピンク色に変わります。同時に画像イメージ表示部のオブジェクトウィンドウもピンク色に変わります。ただし、シーケンスフローはクリックしてもピンク色に変わりません。

シーケンス、オブジェクトを選択すると、フローチャート右側のプロパティ、リザルトリストに選択したシーケンス、またはオブジェクトのプロパティとリザルトが表示されます。また、シーケンスツリーの対応するノード部分の背景が灰色になります。

シーケンスフローは、シーケンスに含まれる全てのオブジェクトが実行に成功した場合、枠が緑色になり、1つでも失敗したオブジェクトがあれば赤色になります。

オブジェクトフローは、実行に成功すれば、緑色に、失敗すれば赤色になります。

フローチャートでは、フローを右クリックすることで様々な操作を呼び出すことができます。

シーケンスフローを右クリックした場合、次の操作ができます。

<b>新規シーケンス</b>	ビジョンシーケンスを作成します。
<b>シーケンス削除</b>	現在のプロジェクトのビジョンシーケンスを削除します。
<b>ステップ順序変更...</b>	シーケンスステップ順序を変更します。
<b>シーケンス実行</b>	ビジョンシーケンスの全体が実行されます。
<b>新規ステップ追加</b>	ステップウィザードを表示します。ステップウィザードを使用することで、オブジェクトを追加することができます。

オブジェクトフローを右クリックした場合、現在アクティブなビジョンオブジェクトに次の操作ができます。

<b>オブジェクト実行</b>	ビジョンオブジェクトを実行します。 Enabled プロパティが“False”に設定されている場合、クリックできません。
<b>コピー</b>	ビジョンオブジェクトをコピーします。
<b>切り取り</b>	ビジョンオブジェクトを切り取ります。
<b>削除</b>	ビジョンオブジェクトを削除します。

モデルティーチ	ビジョンオブジェクトがティーチを必要とする場合、オブジェクトのモデルウィンドウ部分のティーチを行います。
モデル表示	ティーチ済みの教示モデルを表示します。
ウィンドウ 編集	サーチウィンドウに検出マスクを設定できるウィンドウ編集モードをアクティブにします。
オブジェクト無効化	Enabled プロパティを “False” に変更し、ビジョンオブジェクトが実行されないように指定します。
オブジェクト有効化	Enabled プロパティを “True” に変更し、ビジョンオブジェクトが実行するように指定します。

### 4.5.7 シーケンス, キャリブレーションツリー

シーケンス, キャリブレーションツリーは、フローチャート右側上部に表示されています。ツリーには、すべてのシーケンスとキャリブレーションプロジェクトが表示されています。シーケンスツリーでのシーケンスノードやオブジェクトノードはフローチャートでのフローの操作と同じ操作ができます。

キャリブレーションノードをクリックしたとき、実行パネルの表示がキャリブレーション用の実行パネルの表示に変わります。



## 4.6 Vision Guideウィンドウ

### 4.6.1 プロパティリスト ツールバーとボタン

ツールバーとボタンは、プロパティリストの上部に表示されます。プロパティリスト/リザルトリストの、表示モードを選択できます。(デフォルト: ベーシックモード)

表示モードの種類は、次のとおりです。

#### ボタン 説明



#### ベーシックモード:

選択されたビジョンシーケンス, ビジョンオブジェクト, キャリブレーションで、一般的に使用される、プロパティとリザルトが、リストに表示されます。ビジョンガイドをはじめて使用する場合や、プロパティ設定値の変更を最小限にして、ビジョンオブジェクトをテストしたいときに、選択します。

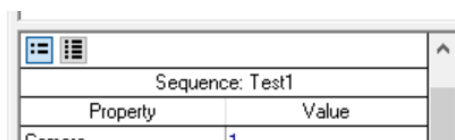


#### アドバンスドモード:

選択されたビジョンシーケンス, ビジョンオブジェクト, キャリブレーションで、利用できるすべてのプロパティとリザルトが、リストに表示されます。すべてのプロパティとリザルトを使用して、最適なビジョンシーケンスへと調整したいときに、選択します。

ここで選択したモードは、シーケンス, オブジェクト, キャリブレーションに反映され、EPSON RC+7.0を再度開いたときにも、同じ設定が引き継がれます。

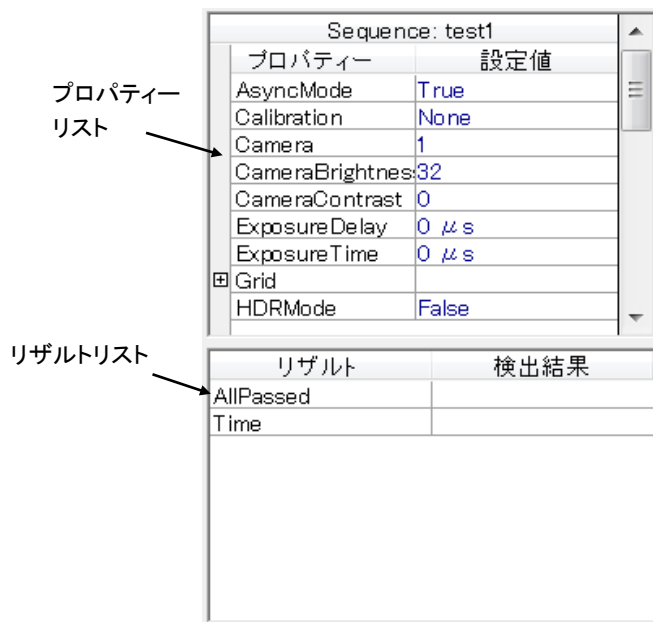
例えば、アドバンスドモードを選択すると、選択されたシーケンス, オブジェクト, キャリブレーションでは、利用できるすべてのプロパティとリザルトが表示されます。EPSON RC+7.0を再度開くと、最後に選択していたアドバンスドモードで、表示されます。



## 4.6.2 シーケンスウィンドウ

シーケンスウィンドウはフローチャートからビジョンシーケンスをクリックするか、シーケンスツリーからクリックすることで画面右側に表示されます。シーケンスウィンドウの用途は、次のとおりです。

- ビジョンシーケンスのプロパティの設定
- ビジョンシーケンス全体の実行結果のテスト



## シーケンスウィンドウ: プロパティリスト

ビジョンシーケンスのプロパティは、小型のスプレッドシートのようなプロパティリストに表示されます。プロパティ名は左側に、プロパティの設定値は右側に表示されます。

ビジョンプロパティを設定するには、まず特定のプロパティについて設定値フィールドをクリックして、値を入力するか、または表示されているプロパティリストから設定値を選択します。

ビジョンシーケンスのプロパティの例を次に示します。

Camera	ビジョンシーケンスで使うカメラの指定
RuntimeAcquire	ビジョンシーケンスで使う取り込み方法の定義 (例: strobed, none, stationary)
Calibration	ビジョンシーケンスで使うキャリブレーションの定義
ImageBuffer	ビジョンシーケンスで使うバッファ番号を指定 0: 各カメラが持つ専用のバッファ番号 (デフォルト) 1~10: ビジョンシーケンス内で共有されるバッファ
ImageSize	ビジョンシーケンスで使うカメラの解像度を指定 デフォルト: カメラの解像度

ビジョンシーケンスのビジョンプロパティリスト全体一覧については、「ビジョンシーケンス」、またはVision Guide 7.0 プロパティ & リザルトリファレンスを参照してください。

### シーケンスウィンドウ: リザルトリスト

ビジョンシーケンスのリザルトは、プロパティリストの下にあるリザルトリストに表示されます。ここでは、ビジョンシーケンスの実行後に、リザルト値のフィールドに値を表示するだけです。実行前のリザルトフィールドには、なにも表示されていません。

シーケンスリザルトのフィールドに表示されるのは、次の2つのリザルトだけです。

**AllPassed**                      すべてのビジョンオブジェクトがPassしたかどうかを表示

**Time**                              ビジョンシーケンスの処理に要した時間を表示

### 4.6.3 オブジェクトウィンドウ

オブジェクトウィンドウはフローチャートからビジョンオブジェクトをクリックするか、シーケンスツリーからクリックすることで画面右側に表示されます。オブジェクトウィンドウの用途は、次のとおりです。

- ビジョンオブジェクトのプロパティの設定
- ビジョンオブジェクト、またはビジョンシーケンス実行後のリザルトの表示

プロパティリスト →

Step 2: Geom01	
Property	Value
AbortSeqOnFail	False
Accept	700
AngleEnable	True
AngleOffset	0.000
AngleRange	180.000
AngleStart	0.000
Caption	
CenterPointObject	Screen
Confusion	800

リザルトリスト →

Result 1 of 1	Value
Angle	0.708 deg
CameraX	(no cal) mm
CameraY	(no cal) mm
Found	True
FoundOnEdge	False
NumberFound	1
Overlapped	False
Passed	True

### オブジェクトウィンドウ: プロパティリスト

ビジョンオブジェクトのプロパティは、小型のスプレッドシートのようなプロパティリストに表示されます。プロパティ名は左側に、プロパティの設定値は右側に表示されます。

ビジョンオブジェクトのプロパティを設定するには、まず特定のプロパティについて設定値フィールドをクリックして、値を入力するか、または表示されているプロパティリストから値を選択します。

ビジョンオブジェクトのプロパティは、ビジョンオブジェクトのタイプによって異なりますが、一般的なビジョンオブジェクトプロパティの例を、次に示します。

<b>AbortSeqOnFail</b>	Trueに設定されているときは、シーケンス実行中にこのオブジェクトが良(Pass)判定されなかった場合、シーケンスの全体を中断します。
<b>Accept</b>	Scoreリザルトとの比較に用いるしきい値 ScoreがAcceptプロパティの設定より大きいとき、オブジェクトは検出されたものと判断されます。
<b>Frame</b>	オブジェクトの位置決めに、どのビジョンフレームを適用するかを定義
<b>PassType</b>	オブジェクトの検出結果の良否条件を定義

ビジョンオブジェクトのビジョンプロパティリストの全体一覧については、「6. ビジョンオブジェクト」、またはVision Guide 7.0 プロパティ&リザルトリファレンスを参照してください。

### オブジェクトウィンドウ: リザルトリスト

ビジョンオブジェクトのリザルトは、プロパティリストの下にあるリザルトリストに表示されます。ここでは、このビジョンオブジェクトまたは全体のビジョンシーケンスの実行後に、リザルト値フィールドに値を表示します。実行前のリザルト値フィールドには、なにも表示されていないか、または直前に実行したビジョンオブジェクトのリザルトが表示されます。

ビジョンシーケンス全体を実行するときは、オブジェクトウィンドウを表示すると便利な場合がよくあります。シーケンスを実行する度に、その特定のオブジェクトのリザルト見ることができます。

Vision Guide 7.0により、各ビジョンオブジェクトのリザルトを見るために、ユーザーはビジョンシーケンスを実行して、ビジョンオブジェクトを切り替えることができます。必ずビジョンオブジェクトドロップダウンリストを使って、ビジョンオブジェクトを選択してください。画像イメージ表示部のオブジェクトをクリックして、ビジョンオブジェクトの1つを選択した場合、そのオブジェクトのリザルトは消去されます。

Vision Guide 7.0は、ビジョンオブジェクトを切り替えてもリザルトを保存します。

リザルトは、ビジョンオブジェクトのタイプによって異なりますが、共通して用いられるリザルトの例を、次に示します。

<b>Found</b>	ビジョンオブジェクトが検出されたかどうかを表示
<b>Passed</b>	オブジェクトの検出結果の良否判定結果を表示
<b>Time</b>	ビジョンオブジェクトの実行にかかった時間を表示

リザルトリストを右クリックした場合、現在アクティブなビジョンオブジェクトに対して、次の操作ができます。

<b>全てのリザルトのコピー</b>	リザルトすべてを、クリップボードにコピーします。
<b>全てのリザルトの エクスポート</b>	リザルトすべてを、CSVファイルにエクスポートします。 表形式で保存できるため、複数リザルトを返すビジョンオブジェクトを使用するときに便利です。

ビジョンオブジェクトの登録の詳細は、「6. ビジョンオブジェクト」を参照してください。

実行パネルのオブジェクトの<Run>ボタンは、現在表示されているビジョンオブジェクトを実行するのに使います。オブジェクトの<Run>ボタンを操作することによって、現在のビジョンオブジェクトは、プロパティが適切に設定されているかどうかをセルフテストします。

ビジョンオブジェクトが他のビジョンオブジェクトと関連している場合(Correlationオブジェクトのリザルトによって中心位置が定義されるPolarオブジェクトなど)は、定義する側のビジョンオブジェクトを最初に実行し、次に現在のビジョンオブジェクトを実行します。

例えば、PolarオブジェクトのCenterPointObjectプロパティがCorrelationオブジェクトのリザルトで定義される場合に、Polarオブジェクトを実行したいと仮定します。この場合、フローチャートからPolarオブジェクトを現在のオブジェクトとして選択します。<オブジェクト実行>ボタンをクリックすると、Correlationオブジェクトの実行が最初に行われ、続いてすぐに、Polarオブジェクトの実行が行なわれます。

## NOTE



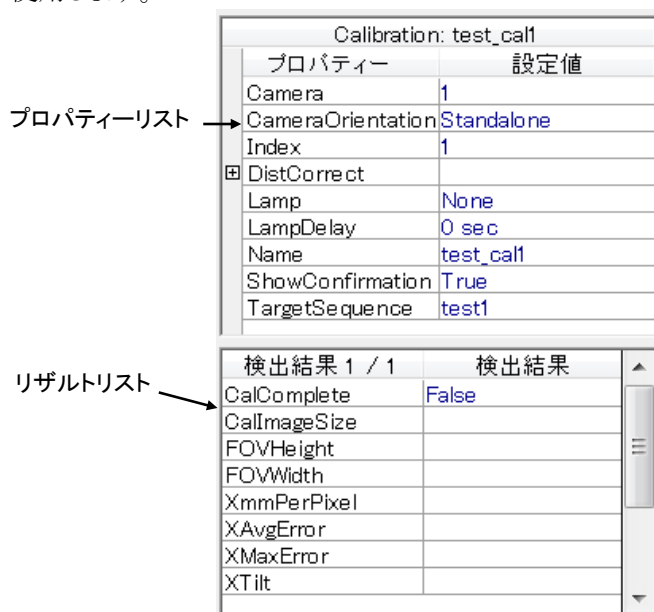
ビジョンオブジェクトの実行とビジョンシーケンスの実行の違いを、充分ご理解ください。ビジョンオブジェクトを実行するには、実行パネルのオブジェクトの<Run>ボタンをクリックするだけです。

実行する前に、フローチャート、またはシーケンスツリーから、正しいビジョンオブジェクトが選択されていることを確認してください。ビジョンオブジェクトの実行は、選択されているビジョンオブジェクトと、そのオブジェクトの実行に必要とされるビジョンオブジェクト(例えば、LineオブジェクトのStartingPointプロパティを定義するオブジェクトや、PolarオブジェクトのCenterPointObjectプロパティを定義するオブジェクトなど)を実行するだけです。ビジョンシーケンスを実行するときは、シーケンス中のすべてのビジョンオブジェクトが実行されます。

#### 4.6.4 キャリブレーションウィンドウ

キャリブレーションウィンドウは、シーケンスツリーからクリックすると画面右側に表示されます。

キャリブレーションウィンドウは、キャリブレーションの設定とキャリブレーション結果の確認に使用します。



##### キャリブレーションウィンドウ: プロパティリスト

小さいスプレッドシートがビジョンキャリブレーションのプロパティリストです。プロパティの名前は左に、値は右に示されます。まず、特定のプロパティの値フィールドをクリックすると、プロパティが設定されます。値を入力したり、表示されているリストから選択しても、プロパティが設定されます。

キャリブレーションプロパティについての詳細は、「4.9 ビジョンキャリブレーション」、または Vision Guide 7.0 プロパティ&リザルトリファレンスを参照してください。

##### キャリブレーションウィンドウ: リザルトリスト

ビジョンキャリブレーションの結果は、プロパティリストのすぐ下にある、リザルトリストに表示されます。

カメラキャリブレーションの詳細は、「4.9 キャリブレーション」を参照してください。

### 4.6.5 ジョグタブ

ジョグタブはキャリブレーション中、またはライブイメージ画像を表示中にロボットをジョグするとき 사용됩니다。



ジョグタブはシーケンス、キャリブレーションツリーの右側に表示されている<Jog>ボタンをクリックすると表示されます。また、ジョグタブはフライアウトでき、任意の場所にタブを設置することができます。

[ジョグ]タブを選択すると、コントローラーと通信を開始します。通信できなかった場合、ひとつ前に選択していたタブが表示されます。

ジョグを行うとき、ロボットのモーターはオンの必要があります。モーターはロボットマネージャ、またはコマンドウィンドウからオンにできます。

#### ジョグタブ: ジョグモードの設定

[モード]ボックスでジョグモードを設定します。World, Tool, Local, Joint, ECP (ECPが有効の場合)があります。現在のTool, Local, ECPをロボットマネージャで設定できます。

#### ジョグタブ: ジョグ速度の設定

[Speed]ボックスでジョグ速度を設定します。

#### ジョグタブ: ジョグ移動距離の設定

[ジョグ移動距離]グループボックスのオプションボタンでジョグ移動距離を設定します。“大”、“中”、“小”を選択する場合、テキストボックスに値を入力すると移動距離を変更できます。

#### ジョグタブ: ロボットをジョグする

ジョグモード、ジョグ速度、ジョグ移動距離を設定した後、ジョグボタンをクリックすると、ロボットをジョグできます。ジョグ移動距離を“連続”に設定すると、ジョグボタンを離すまでロボットを連続ジョグできます。他のジョグ移動距離を設定すると、設定したジョグ移動距離を1ステップそして動作します。ジョグボタンをクリックし続けると、ロボットはジョグ動作をし続けます。

### 4.6.6 ロボットタブ

ロボットタブはシーケンス、キャリブレーションツリーの右側、ジョグタブの下に表示されている<Robot>ボタンをクリックすることで表示されます。また、ロボットタブはフライアウトで任意の場所にタブを設置することができます。

ロボットタブは、ロボットをジョグするときに必要なさまざまな機能の設定に使用します。以下の操作が可能です。

カレントロボットの選択

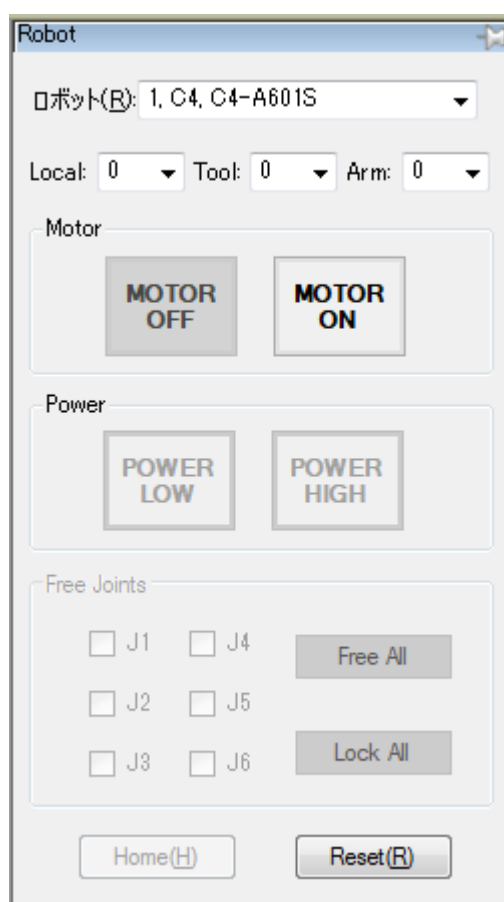
モーターパワーのON/OFF

パワーの変更

関節の設定

Home, MCalの実行

Resetの実行





## 4.7 ビジョンシーケンスとオブジェクト

### 4.7.1 ビジョンシーケンスの概要

ビジョンシーケンスの基本的な知識と使用方法について説明します。

本項では、以下について説明します。

ビジョンシーケンスの定義と一般的な知識

シーケンスウィンドウ

ビジョンシーケンスの作成と削除

ビジョンシーケンスに固有のプロパティとリザルト

Vision Guideウィンドウからのビジョンシーケンスの実行

ビジョンシーケンスのテストとデバッグ

SPEL+言語からビジョンシーケンスを実行する方法

画像イメージの取り込み (ビジョンシーケンスから画像イメージを取り込む方法)

### 4.7.2 ビジョンシーケンスの定義と一般的な知識

#### ビジョンシーケンスとは何か

ビジョンシーケンスは、ビジョンオブジェクトを特定の順番でグルーピングしたもので、Vision Guideウィンドウ、またはSPEL+言語から実行することができます。

ビジョンシーケンスは、ビジョンオブジェクトと同様に、ビジョンシーケンスの実行を設定するための特定のプロパティを備えています。例えば、**Camera**プロパティは、どのカメラを用いてビジョンシーケンスに画像イメージを取り込むかを定義し、**RuntimeAcquire**プロパティは、カメラからどのように画像を取り込むかを定義します。

ビジョンオブジェクトは、個々に実行し、テストすることが可能ですが、ほとんどのビジョンアプリケーションでは、最終結果を計算するため、一連のビジョンオブジェクトを連続的に実行する必要があります。

ビジョンオブジェクトを特定の順番で実行するために、ビジョンシーケンスを使います。個々のビジョンオブジェクトの作成とテストが完了した後、ビジョンシーケンスのテストを行います。

#### ビジョンシーケンスはどのように動作するか

ビジョンシーケンスは、1つあるいはそれ以上のビジョンオブジェクトから構成されます。ビジョンシーケンスの基本的な目的は、これらのビジョンオブジェクトを、定義された順番で連続的に実行することです。

次のステップでは、ビジョンシーケンスの実行によってどのような結果が生じるのかを示します。

1. ビジョンシーケンスの実行の設定は、そのプロパティを設定することで行います。これらのプロパティの設定には、画像イメージを取り込むカメラの設定や、画像イメージ取り込み方法の決定(**RuntimeAcquire**設定)などがあります。
2. 画像イメージは、取り込まれると、ビジョンオブジェクトで使用できるようにフレームバッファに蓄えられます。(RuntimeAcquireプロパティが“None”に設定されている場合は、事前にフレームバッファに蓄えられていた画像イメージを使ってシーケンスが実行されます。これによって、同一の画像イメージを複数のシーケンスで使用することができます。)
3. ビジョンオブジェクトは、上述のステップ2によって取り込まれた画像イメージを使って、連続的な順番で実行されます。すべてのビジョンオブジェクトの実行が終わると、シーケンスリザルトとオブジェクトリザルトは、シーケンスウィンドウとオブジェクトウィンドウに表示されるようになるとともに、SPEL+言語から利用できるようになります。

### ビジョンシーケンスとEPSON RC+ 7.0プロジェクト

EPSON RC+ 7.0は、プロジェクトの概念に基づいて考えられています。

各々のEPSON RC+ 7.0プロジェクトには、特定のアプリケーションに必要なプログラムや、教示ポイント、その他のロボット構成パラメーターが含まれます。

Vision Guide 7.0がEPSON RC+ 7.0プロジェクト内で実行される場合には、そのプロジェクト内で作成されたすべてのビジョンシーケンスが、そのプロジェクトに保存されます。

これは、特定のプロジェクトを次回開くときに、そのプロジェクト内で以前に作成されていたすべてのビジョンシーケンスを利用、修正できるようにするためです。

#### 4.7.3 ビジョンオブジェクトの概要

ビジョンオブジェクトは、すべての作業の基本となる核であり、Vision Guide 7.0のメインの部分形成しています。

Vision Guide 7.0を使用する上で必要とされる基本的な知識と詳細を説明するとともに、Vision Guide 7.0を使ってビジョンアプリケーションを処理する方法について説明します。

- ビジョンオブジェクトの定義と機能
- さまざまなタイプのビジョンオブジェクトの形状の説明
- ビジョンオブジェクトの位置とサイズの設定方法
- ビジョンオブジェクトの用途についての説明
- ビジョンオブジェクトの使用方法
- ビジョンオブジェクトのプロパティとリザルトについて
- ビジョンオブジェクトの具体的な説明
- ビジョンオブジェクトに使うユーティリティについて  
(ヒストグラム、統計、ビジョンオブジェクトグラフィック操作)

#### 4.7.4 ビジョンオブジェクトの定義

##### ビジョンオブジェクトとは何か

ビジョンオブジェクトとは、画像イメージに適用できるビジョンツールです。

ビジョン開発者の多くは、ビジョンオブジェクトを一目見て、Vision Guide 7.0をサポートするアルゴリズムの別名だと考えます。しかし、Vision Guide 7.0を使用いただくと、ビジョンアルゴリズムはビジョンオブジェクトを定義するほんの一部でしかない、ということにお気づきになるでしょう。

ビジョンオブジェクトは、特定のビジョンアルゴリズムの使用に必要な情報を持ったコンテナのようなものです。しかし、データを簡単に操作しテストする独自の方法(プロパティやリザルト)によって、他のビジョンシステムに比べてはるかに迅速に新規ビジョンアプリケーションを開発できる、非常にパワフルなツールです。

ビジョンオブジェクトがサポートするツールには、相関サーチ、2値検査、ポーラサーチ、エッジ検出、ライン、ポイント、フレームなどがあります。すべてのビジョンオブジェクトは実行後、ビジョンオブジェクトの実行に要した時間、位置情報、角度情報、ビジョンオブジェクトが検出できたかどうか、などのリザルトを返します。

詳細は、各ビジョンオブジェクトについての説明を参照してください。

## 4.8 プロパティとリザルトとは

ビジョンシーケンスは、ビジョンオブジェクトと同様にプロパティとリザルトを備えています。主要な違いは、ビジョンオブジェクトプロパティの設定値が、1つのビジョンオブジェクトにしか適用されないのに対して、ビジョンシーケンスのプロパティとリザルトは、ビジョンシーケンス全体に適用される、という点です。

### 4.8.1 ビジョンシーケンスのプロパティ

ビジョンシーケンスのプロパティは、通常、ビジョンシーケンスの実行の設定をするのに使用され、ビジョンシーケンス内のすべてのビジョンオブジェクトが適切に動作するような値を、画像イメージ取込み値に設定します。

ビジョンシーケンスのプロパティのすべては、ビジョンシーケンスの全体に適用されます。例えば、**Calibration**プロパティは、ビジョンオブジェクトのレベルではなく、むしろビジョンシーケンスのレベルで設定されます。

このようにして、ビジョンシーケンス内のすべてのビジョンオブジェクトは、**Calibration**プロパティで規定されたキャリブレーションを使用し、すべての実行結果が同一のキャリブレーションに関して保証されることとなります。

下の一覧は、ビジョンシーケンスプロパティの概略です。各プロパティの詳細は、「Vision Guide 7.0 プロパティ & リザルトリファレンス」を参照してください。

プロパティ	説明
<b>Calibration</b>	ビジョンシーケンスで使用するカメラキャリブレーションを指定 ロボット座標系リザルトやカメラ座標系リザルトを返すため、このプロパティは指定されていなければなりません。 デフォルト: none
<b>Camera</b>	ビジョンシーケンスで使用するカメラを指定 デフォルト: 1
<b>CameraBrightness</b>	輝度を指定 デフォルト: 128
<b>CameraContrast</b>	コントラストを指定 デフォルト: 128
<b>Description</b>	ユーザー指定の説明を設定 デフォルト: 空欄
<b>ExposureDelay</b>	露光までの時間を指定 (単位: マイクロ秒) デフォルト: 0
<b>ExposureTime</b>	非同期リセットモードでの露光時間を指定 (単位: マイクロ秒) デフォルト: 0
<b>ImageBuffer</b>	取得した画像を保存するバッファを指定 デフォルト: 0
<b>ImageColor</b>	取得する色相を指定 デフォルト: 1 - All

プロパティ	説明
ImageFile	作業したいディスクに画像イメージを含むファイルの名前を指定 デフォルト: None (画像イメージファイルなし)
ImageFileScale	画像イメージファイルの倍率を指定 デフォルト: 0 (ImageSizeに合わせる)
ImageSize	取得した画像イメージの解像度を指定 デフォルト: 使用中のカメラの最大解像度
ImageSource	画像イメージソースを指定 デフォルト: 1 - Camera
Index	シーケンスのインデックスを表示
HDRMode	撮像画像をHDR画像で表示 デフォルト: False
Name	ビジョンシーケンス名
RuntimeAcquire	ビジョンシーケンスで使用する画像イメージの取り込み方法を定義 デフォルト: 1 - Stationary
RuntimeFreeze	この設定値によって、ビジョンシーケンスは画像イメージの表示を一時停止 デフォルト: True
SaveImage	現在の画像イメージをディスクに保存するダイアログを表示
ShowProcessing	画像イメージ処理を表示するかどうかを指定 デフォルト: True
StrobeBlackVideo	シーケンス開始後、トリガーを受けて画像イメージを取得する前に画像イメージを黒に戻すかを指定 デフォルト: True (画像イメージを黒に戻す)
StrobeDelay	ストロボ発光までの時間を指定(単位: マイクロ秒) デフォルト: 0
StrobeTime	ストロボ発光時間を指定(単位: マイクロ秒) デフォルト: 0
TriggerMode	電子シャッターのトリガー論理を指定します。 デフォルト: Leading Edge (立ち上がりエッジ)
GridColor	画像イメージ表示部に表示するグリッドの色を指定
GridPitchX	画像イメージ表示部に表示するグリッドのXピッチを指定
GridPitchY	画像イメージ表示部に表示するグリッドのYピッチを指定
GridShow	画像イメージ表示部にグリッド表示するかどうかを指定 デフォルト: False
GridType	画像イメージ表示部に表示するグリッドタイプを指定 1 - 十字 2 - 格子

プロパティ	説明
GridUnits	画像イメージ表示部に表示するグリッドのピッチ単位を指定 1 - 画素数 2 - mm 2 - mm選択時、キャリブレーション未実施の場合には、グリッドは表示されません。

### 4.8.2 ビジョンシーケンスのリザルト

ビジョンシーケンスのリザルトは、ビジョンシーケンス全体に適用されます。

これらのリザルトは、ビジョンシーケンス全体についての情報を表示するので非常に便利です。例えば、AllPassedリザルトは、ビジョンシーケンス内のビジョンオブジェクトのすべてが検出されたかどうかの結果を返します。

ビジョンシーケンスで使われるリザルトの一覧を次に示し、簡単に説明します。各リザルトの詳細は、「Vision Guide 7.0 プロパティ & リザルトリファレンス」を参照してください。

リザルト	説明
AllPassed	ビジョンシーケンス内のビジョンオブジェクトのすべてがPassしたかどうかの結果を返す
Time	ビジョンシーケンスの実行に要した合計時間を返す この合計時間には、ビジョンオブジェクトすべての実行に要した累積時間と、画像イメージの取り込みに要した時間が含まれています。

## 4.9 キャリブレーション

### 4.9.1 キャリブレーションについて

キャリブレーションは、Vision Guide 7.0システムで重要な部分を担っています。

ロボット座標系で、パーツの整列や、物理的な測定を行うには、まずシステムのキャリブレーションを実施してください。

Vision Guide 7.0では、可動カメラ、上向き固定カメラ、下向き固定カメラ、スタンドアローンカメラのキャリブレーションをサポートしています。

### 4.9.2 キャリブレーションの定義

Vision Guide 7.0では、各プロジェクトに複数のキャリブレーションをサポートしています。どのキャリブレーションも、同一プロジェクト内の複数のビジョンシーケンスで使用できます。

各キャリブレーションは、次のデータで構成されています。

キャリブレーション名	ビジョンシーケンスが基準とするキャリブレーション名 (アルファベットで半角16文字まで)
カメラ番号	キャリブレーションが行われるカメラの番号
カメラ方向	カメラの取り付け方法
キャリブレーションターゲットシーケンス名	カメラのキャリブレーションに使用されるビジョンシーケンス名 (カレントプロジェクトにあるどのシーケンスでも可能)
ロボットのキャリブレーション速度と加速度 (必要に応じて)	キャリブレーション中に適用される速度と加速度
キャリブレーション中に使用される、ロボットアーム番号とツール番号	キャリブレーション中にアームやツールを使用することができます。これらのアームやツールは、ポイントの教示やキャリブレーションをする前に定義してください。
基準点やカメラ位置となるロボットポイント	これらのポイントは、キャリブレーションごとに保存されます。
ライト出力	キャリブレーションをする間、照明のコントロールに使用するロボットコントローラーの出力オプションです。
撮像歪み補正有無	撮像歪み(カメラのチルト、およびレンズ歪み)の補正を実施するかどうかを指定
撮像歪み補正ターゲットシーケンス名	撮像歪みの補正に使用されるビジョンシーケンス名 (カレントプロジェクトにあるどのシーケンスでも可能)

それぞれのキャリブレーションでは、キャリブレーションするカメラの視野をカバーする9つの点を使用しています。

視野全体をカバーすることにより、システムのカメラ傾きの補正やレンズ精度の補正に最高の結果を得ることができます。

視野全体で距離測定を実行する場合、撮像歪み補正を実施してください。撮像歪み補正を実施することで、カメラのチルト (作業平面とカメラ光軸との角度ずれ) とレンズによる歪みを補正することができます。

### どのようなときにキャリブレーションが必要か

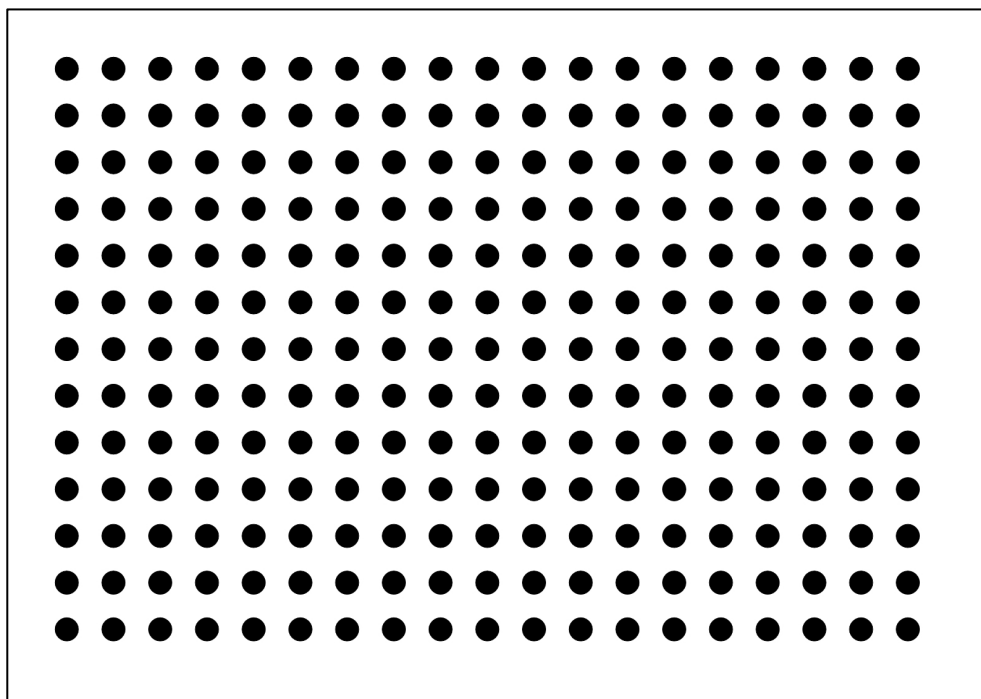
物理的な測定を行うとき、あるいはロボット座標系の位置を特定するときには、必ずキャリブレーションを行う必要があります。

ビジョンオブジェクトに返されるリザルトの中には、キャリブレーションを実行しておく必要のあるリザルトもいくつかあります。例えば、CameraX, CameraY, RobotX, RobotY, RobotUのリザルトを得るには、キャリブレーションの実行が必要です。

#### 4.9.3 撮像歪み補正用の撮像パターン

撮像歪みの補正に使用できる画像イメージ例を以下に示します。

撮像歪みの補正を行うためには、グリッドが縦横同一ピッチで、かつカメラ視野内に100個以上のグリッドが検出できるようにする必要があります。



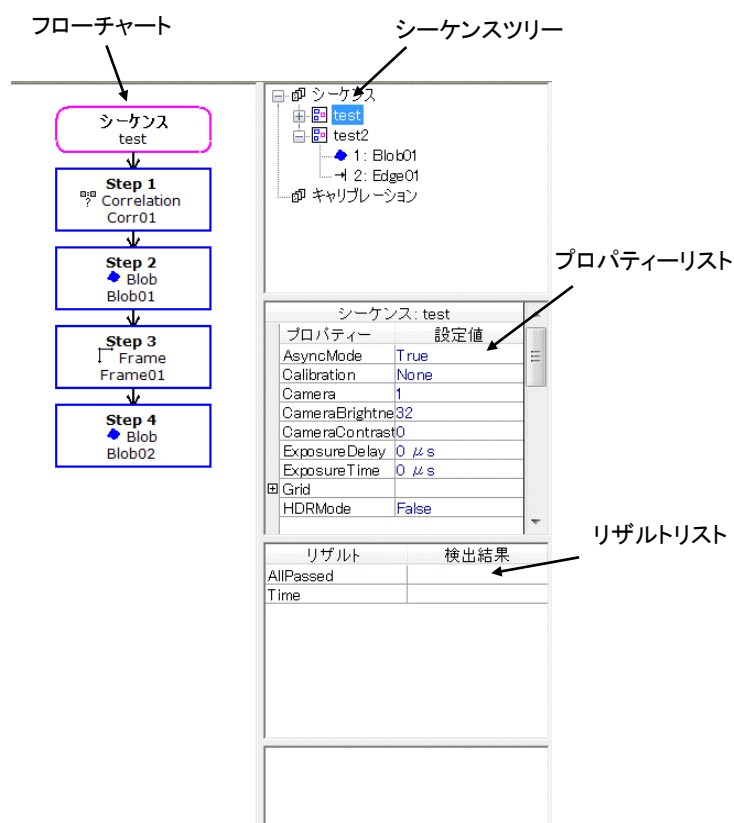


## 5. ビジョンシーケンス

### 5.1 ビジョンシーケンスウィンドウ

シーケンスウィンドウは、下記の用途に使用します。

- ビジョンシーケンスのプロパティの設定
- ビジョンシーケンス全体の実行結果の確認



#### 5.1.1 ビジョンシーケンスの選択

シーケンスツリー左の“+”キーをクリックすると、このプロジェクトの全ビジョンシーケンスのリストが表示されます。このビジョンシーケンスリストから、作業したいシーケンスをクリックして選択することができます。

#### 5.1.2 シーケンスウィンドウのプロパティリスト

ビジョンシーケンスのプロパティは、小さなスプレッドシートに似たプロパティリストに表示されます。プロパティ名は左側に、その設定値は右側に表示されます。

ビジョンプロパティは、特定プロパティの設定値フィールドをクリックし、そのフィールドに値を入力するか、あるいは表示されているリストから選択することによって、値を設定することができます。

ビジョンシーケンスプロパティについては、「4.8.1 ビジョンシーケンスのプロパティ」を参照してください。

### 5.1.3 シーケンスウィンドウのリザルトリスト

ビジョンシーケンスのリザルトは、プロパティリストのちょうど真下にあるリザルトリストに表示されます。

リザルトリストでは、ビジョンシーケンスが実行されない限り、リザルト値のフィールドに値が表示されません。

ビジョンシーケンスリザルトについては、「4.8.2 ビジョンシーケンスのリザルト」を参照してください。

### 5.1.4 シーケンスのフローチャート


画像表示部右側には、シーケンスのフローチャートが表示されます。このフローチャートには、このビジョンシーケンスに定義されているビジョンオブジェクトのすべてを表示します。

フローチャートに表示される順番は、ビジョンオブジェクトの実行の順番になっています。

詳細は、「5.5 シーケンス順序の変更」を参照してください。


## 5.2 ビジョンシーケンスの新規作成

### 5.2.1 概要

ビジョンシーケンスを作成します。Vision Guideツールバー -  <新規シーケンス>ボタンをクリックしてください。または、フローチャートのシーケンスフロー、もしくはシーケンスツリーのシーケンスノードを右クリックし、「新規シーケンス」の項目を選択してください。すると、ダイアログが表示されて、新規シーケンス名を入力するように要求しますので、アルファベットの半角16文字以内でシーケンス名を入力し、<OK>ボタンをクリックしてください。

新しいシーケンスを作成するとき、存在するシーケンスドロップダウンリストからコピーして、シーケンスを作成することもできます。

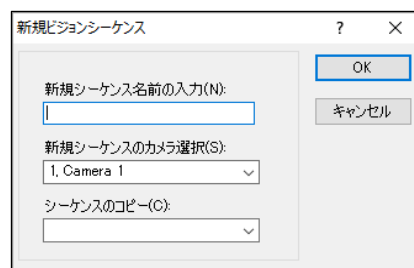
### 5.2.2 ショートカット

ツールバー : 

キー : None


### 5.2.3 ダイアログボックスのオプション

オプション	説明
OK	画面入力したシーケンス名で新規ビジョンシーケンスを作成
Cancel	新規ビジョンシーケンスの作成をキャンセル
Help	新規ビジョンシーケンスコマンドのヘルプを表示



## 5.3 ビジョンシーケンスの削除

### 5.3.1 概要


Vision Guideツールバー -  <シーケンス削除>ボタンをクリックするか、フローチャートのシーケンスフロー、もしくはシーケンスツリーのシーケンスノードを右クリックし、「シーケンス削除」の項目を選択してください。すると、ダイアログが開いて、現在のプロジェクトに作成済みのビジョンシーケンスがすべてリスト表示されます。

マウスあるいは矢印キーを使って、削除したいビジョンシーケンスを選択してください。

削除したいビジョンシーケンス名が強調表示されたら、ビジョンシーケンスダイアログの<削除>ボタンをクリックしてください。確認のメッセージが表示されます。

いったんビジョンシーケンスを削除して、カレントプロジェクトを保存してしまうと、修復することはできません。誤ってシーケンスを削除した場合は、[ファイル]-[復元]を実行すれば、ビジョンプロジェクト全体を最後に保存した状態に復元できます。後日そのシーケンスを使用する機会があると思われる場合には、保存するように気をつけてください。プロジェクトにシーケンスを保存しても、プログラムでそのシーケンスを使用しなければならないということはありません。

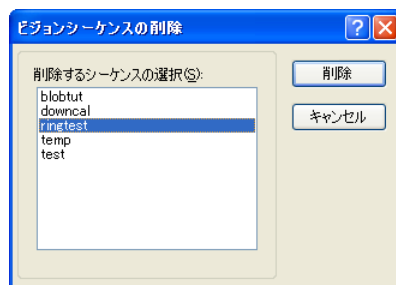
### 5.3.2 ショートカット

ツールバー : 

キー : None


### 5.3.3 ダイアログボックスのオプション

オプション	説明
選択リスト	削除対象となるシーケンスを選択
削除	強調表示されたビジョンシーケンスを削除
キャンセル	ビジョンシーケンスの削除をキャンセル
ヘルプ	シーケンスの削除コマンドのヘルプを表示



## 5.4 ビジョンオブジェクトの削除

ビジョンオブジェクトを削除するには、まず削除したいビジョンオブジェクトを選択します。

Vision Guideツールバー -  <オブジェクト削除>ボタン をクリックします。確認ダイアログが表示されますので、<はい>ボタンをクリックしてオブジェクトを削除します。

<オブジェクト削除>ボタンは、カレントプロジェクトのビジョンシーケンスがない場合やオブジェクトがない場合はグレースアウト表示になっています。

## 5.5 シーケンスステップ順序の変更

シーケンスステップ順序を変更するには、フローチャートのシーケンスフロー、またはシーケンスツリーのシーケンスノードを右クリックし、<ステップ順序変更...>をクリックすることで[シーケンス順序の変更]ダイアログが表示されます。オブジェクトのステップ番号を変更するには、リストにあるオブジェクトを選択し、上下の矢印ボタンをクリックします。<OK>ボタンをクリックすると変更されます。



ビジョンオブジェクトの実行順序を変更するとき、他のオブジェクトに基づいているビジョンオブジェクトに特に注意してください。“Corr01”のStartingPointプロパティと“Blob01”のEndPointプロパティを持つLineオブジェクトがその一例です。Lineオブジェクトのステップを“Corr01”オブジェクトの上に移動しようとしても、“Blob01”オブジェクトの下に残ります。そうすると、「警告」で、LineオブジェクトのStartingPointプロパティを“Screen”に変更するよう表示します。他のオブジェクトに基づいているビジョンオブジェクトの多数の組み合わせがありますので、適切に実行するには、他のオブジェクトを必要とするオブジェクトには注意してください。

### 5.6 ビジョンシーケンスの実行

#### 5.6.1 ビジョンシーケンスの選択

ビジョンシーケンスを実行する前に、実行したいビジョンシーケンスを、シーケンスツリーから選択する必要があります。このシーケンスツリーによって、プロジェクト内のビジョンシーケンスが簡単に変更できます。

#### 5.6.2 ビジョンシーケンスパラメーター (プロパティ) の設定

実行すべきビジョンシーケンスを選択したら、そのシーケンスに何らかのパラメーターを設定する必要があります。

ビジョンシーケンスパラメーターは、シーケンスウィンドウ上のプロパティリストで設定することができます。

プロパティは、ビジョンシーケンスを実行する前に、適切に設定しておかなければなりません。例えば、カメラ2から取り込まれた画像イメージを使用するシーケンスを実行したい場合には、そのシーケンスのCameraプロパティの値を2に設定しなければなりません。

初めてご使用になる方に共通して最も多い間違いは、Cameraプロパティに間違った値を設定してビジョンシーケンスを実行することです。

##### 複数カメラを使用するときの重要注意:

カメラプロパティは、特定のビジョンシーケンスに使うカメラを選択するのに使われます。カメラは、開始時にシステムが自動的に見つけます。Vision Guide 7.0がカメラのビデオを表示しているときに、そのカメラの電源を入れたら、シーケンスCameraプロパティからカメラを再選択しなければ、ビデオを見ることができません。

#### 5.6.3 ビジョンシーケンスのリザルト

ビジョンシーケンスを実行すると、シーケンスのリザルトを確認することができます。ビジョンシーケンスのリザルトは、シーケンスウィンドウ上のリザルトリストに表示されます。

#### 5.6.4 シーケンスのフローチャート

画像表示部右側には、シーケンスのステップ実行リストが表示されます。このフローチャートには、現在選択されているシーケンスに定義されている、ビジョンオブジェクトのすべてを表示します。

リストに表示される順番は、ビジョンオブジェクトの実行の順番になっています。

#### 5.6.5 ビジョンシーケンスの実行

Vision Guideウィンドウの下方にある実行パネル左側に、シーケンスの<Run>ボタンがあります。このボタンをクリックすると、ビジョンシーケンスの全体が実行されます。

特定のビジョンシーケンスで使用するビジョンオブジェクトの作成とテストが完了したら、実行パネル左側の<Run>ボタンをクリックして、ビジョンシーケンスの全体をテストしてください。

### 5.6.6 シーケンスの複数回数実行

実行パネル左側のシーケンスの<Run>ボタン右側の▼ボタンをクリックすることで使用できるようになる サイクルテキストボックスに注意してください。

このボックス内にある数値は、ビジョンシーケンスを何回実行するかを指定します。シーケンスを複数回実行することにより、ロボットで実行するための命令コードを書く前に、ビジョンシーケンスの信頼性をテストできるので、非常に便利です。

シーケンスを複数回実行し、Vision Guide 7.0の統計機能を使用すると、シーケンス中の各ビジョンオブジェクトの平均、標準偏差、範囲、最小値、最大値を確認することができるため、非常に役立ちます。この機能によって、ビジョンシーケンスが適切に実行できているかを全体的に見ることができます。

複数回実行のために数字をサイクルテキストボックスに入力すると、サイクルテキストボックスが白色から黄色に変わります。(1より大きい数字はすべて複数回数とみなし、色が黄色に変わります。) <Run>ボタンをクリックして、シーケンスを複数回実行するときは注意してください。


### 5.6.7 ビジョンシーケンス実行中の画像イメージ表示部

ビジョンシーケンスを実行すると、画像イメージ表示部には、どのビジョンオブジェクトが検出され、どのビジョンオブジェクトが検出されなかったかが、色別表示されます。

検出されたビジョンオブジェクトは、検出位置とともに、サーチウィンドウのアウトラインが緑色で表示されます。

検出されなかったビジョンオブジェクトは、サーチウィンドウのアウトラインが赤色で表示されます。検出されなかったビジョンオブジェクトについては、検出位置は表示されません。

Graphicsプロパティーが“None”に設定されているビジョンオブジェクトには、いかなるグラフィックも表示されません。すべてのビジョンオブジェクトについて、確実にグラフィックを

表示したい場合には、Vision Guideツールバー -  <全グラフィック表示> ボタンをクリックしてください。この操作をすることによって、個々のGraphicsプロパティーの設定と関係なく、すべてのグラフィックが表示されるようになります。

### 5.6.8 ビジョンシーケンスの中断

<中断>ボタンをクリックすると、ビジョンシーケンスは直ちに実行を中断します。

### 5.7 ビジョンシーケンスのテストとデバッグ

ビジョンシーケンス中のすべてが正常に機能しているかどうかを知るには、ビジョンシーケンスのAllPassedリザルトの状態を調べます。

ビジョンオブジェクトのいずれかの検出結果が不良であった場合、AllPassedビジョンリザルトが、Falseに設定されています。一方で、すべてのビジョンオブジェクトが正しく設定されていないとAllPassedリザルトもまちがえて良品と返す場合があります。最後に、ビジョンシーケンスのテストをして、ビジョンシーケンス中で適切に実行されないものがある場合には、デバッグをする必要があります。

ここでは、Vision Guideシーケンスで使われるデバッグの機能と方法のいくつかについて説明します。


#### 5.7.1 ビジョンオブジェクトの色の確認

問題の原因を解明するために、まず確認すべきことのひとつとして、画像イメージ表示部のビジョンオブジェクトにおかれている、良/不良の色の確認があります。

検出結果-良とされたビジョンオブジェクトは、検出位置とともに、サーチウィンドウが緑色(PassColorプロパティーで変更可能)で表示されます。

不良判定されたビジョンオブジェクトは、サーチウィンドウが赤色(FailColorプロパティーで変更可能)で表示されます。このような場合は、これについてさらに詳細に検討する必要があることを示唆しています。

Graphicsプロパティーが“None”に設定されているビジョンオブジェクトには、いかなるグラフィックも表示されません。すべてのビジョンオブジェクトについて、確実にグラフィックを

表示したい場合には、Vision Guideツールバー -  <全グラフィック表示> ボタンをクリックしてください。この操作をすることによって、個々のGraphicsプロパティーの設定と関係なく、すべてのグラフィックが表示されるようになります。

#### 5.7.2 個々のビジョンオブジェクトリザルトの確認

ビジョンシーケンスを実行すると、シーケンスウィンドウ上にビジョンシーケンスのリザルトを表示したり、オブジェクトウィンドウ上にビジョンオブジェクトのリザルトを表示したりすることができます。

あるビジョンオブジェクトの実行について問題があると推測される場合、また、ビジョンシーケンスの実行中にそのビジョンオブジェクトを詳細にモニターしたい場合などは、数回クリックすれば、表示できます。

- (1) 対象となるビジョンオブジェクトをフローチャート、またはシーケンスツリーから選択します。
- (2) そのビジョンオブジェクトのリザルトを確認します。

ビジョンシーケンスを実行している最中でも、ビジョンオブジェクトの切り替えや、オブジェクトウィンドウ表示とシーケンスウィンドウ表示の切り替えができます。

このような機能によって、1サイクルあるいはそれ以上のビジョンシーケンスの実行を開始することができ、ビジョンシーケンスを中止することなく、ビジョンシーケンス内のさまざまな多数のビジョンオブジェクトについてリザルトを確認することができます。



### 5.7.3 ビジョンシーケンスでの逐次ステップ実行

Vision Guide 7.0開発環境のその他の優れたデバッグ機能のひとつに、逐次ステップ実行機能があります。

ビジョンシーケンスの実行パネルには、<ステップ実行>ボタンがあります。このボタンをクリックすると、一度に1つのビジョンオブジェクトを逐次ステップ実行します。

また、逐次ステップの実行中に、実行前のビジョンオブジェクトについてプロパティを変更することができます。

<ステップ実行>ボタンをクリックすると、すべてのビジョンオブジェクトを静止状態にします。オブジェクトは青色になります。もう一度<ステップ実行>ボタンをクリックすると、シーケンスの最初のビジョンオブジェクトを実行します。オブジェクトが検出されたら緑色に、検出されなければ赤色になります。

### 5.7.4 ビジョン統計機能の使用

もうひとつのツールとして、ビジョン統計機能があります。

ビジョンシーケンスの信頼性をテストする場合は、まず、「サイクル数」の値を高い数値に設定し、ビジョンシーケンスをしばらくの間実行してください。

次に、統計ダイアログで、すべてのビジョンオブジェクトの統計結果をチェックします。信頼性の高い結果が得られたのは、どのビジョンオブジェクトか、また結果にばらつきが多いのは、どのビジョンオブジェクトかを確認することができます。

ビジョン統計機能の詳細は、「9. 統計ツール」を参照してください。

### 5.7.5 RUNウィンドウとVision Guideウィンドウの切り替え

何度かテストを実行することにより、RUNウィンドウから最終テストを実行する準備ができたと判断できる場合や、まだ調整する必要があると判断できる場合があります。このような場合に備えて、RUNウィンドウの実行中に、Vision Guide 7.0開発環境ウィンドウへのアクセスができるようになっています。

例をあげて説明してみましょう。

今、RUNウィンドウからVision Guide 7.0アプリケーションを実行しているとして、前回実行時に問題のあったCorrelationオブジェクト“pin1”について、Scoreリザルトを確認すると仮定してください。

まず、EPSON RC+ 7.0ツールバー -  <ビジョン>ボタンをクリックします。この操作によって、アプリケーションを実行しながら、Vision Guide 7.0開発環境ウィンドウを表示させることができます。

フローチャートから“pin1”オブジェクトを選択してください。これで、プログラムの実行を実際に中止することなく“pin1”オブジェクトのリザルトを表示することができます。

RUNウィンドウに戻りたい場合は、EPSON RC+ 7.0ツールバー - <RUNウィンドウ>ボタンをクリックしてください。

### 5.8 SPEL+からビジョンシーケンスを実行する方法

Vision Guide 7.0の基本的な設計目標は、Vision Guide 7.0開発環境からポイントクリック操作で作成されたものは、SPEL+言語から全面的にアクセス可能にすることです。つまり、Vision Guide 7.0開発環境をプロトタイピング環境として使用し、それを書き直すことなくSPEL+言語においても使用できるようにすることです。

したがって、Vision Guide 7.0開発環境から作成されたビジョンシーケンスは、SPEL+のVRunコマンドで実行できるようになっています。

VRunのシンタックスは“VRun seqname”です。このときの“seqname”は、実行したいビジョンシーケンス名を表します。下記に示す命令コードの一部は、それぞれ異なるビジョンシーケンスを実行する、2つのVRunステートメントを示しています。

“VRun findpin”は、ロボットがピンをピックアップするために、ピンの中心位置を検出するシーケンスを実行します。

“VRun housing”は、ハウジング中のピンを置く位置を検出するシーケンスを実行します。

```
VRun findpin
VGet findpin.pincntr.RobotXYU, found, X, Y, U
.
.
VRun housing
VGet housing.setpos.RobotXYU, found, X, Y, U
```

以上は、VRunコマンドの使用例のごく一部でしかありません。Vision GuideシーケンスやロボットでSPEL+言語を使用する方法の詳細は、「11. SPEL+でVision Guideを使う方法」を参照してください。

### 5.9 画像イメージの取り込み

#### 5.9.1 画像イメージはいつ取り込まれるか

多くのビジョンシステムでは、処理すべき画像イメージを取り込むための特定のコマンドやステップを実行しなければなりませんが、Vision Guide 7.0では、ビジョンシーケンスの実行開始時に画像イメージが取り込まれるので、このような余分な手順を省略することができます。

必要な手順は、RuntimeAcquireプロパティに適切な値を設定することだけです。ただし、RuntimeAcquireプロパティは、“Stationary”にデフォルト設定されているため、ほとんどのアプリケーションで、このプロパティを設定する必要はありません。デフォルト設定では、画像イメージはビジョンシーケンスの実行開始時に取り込まれるようになっています。他に、画像取得について設定するシーケンスプロパティがあります。

### 5.9.2 同一の画像イメージを2つ以上の異なるビジョンシーケンスで使用方法

同一の画像イメージを、2つ以上のビジョンシーケンスで使用したい場合は、1番目のビジョンシーケンスのRuntimeAcquireプロパティを“Stationary”に設定し、他のビジョンシーケンスのRuntimeAcquireプロパティを“None”に設定するだけです。他のビジョンシーケンスのRuntimeAcquireを“None”に設定することにより、このシーケンスは他の画像イメージを取り込むことをせず、1番目のビジョンシーケンスで取り込まれた画像イメージに基づいて、すべての処理を行います。

### 5.9.3 イメージバッファの使用法

各カメラにひとつのイメージバッファ (バッファ 0)があり、シーケンス内で共有できる10個のイメージバッファ (バッファ 1~10)があります。

ImageBufferシーケンスプロパティで、使用するバッファを指定します。デフォルトは、バッファ 0です。イメージバッファを使用すると画像をメモリーに保存しておき、後で処理することができます。

例えば、オブジェクトのないシーケンスを作成して複数のバッファに画像を取り込み、他のシーケンスで取り込んだ画像を処理することができます。

```
' seq1 はビジョンオブジェクトなし - 複数のバッファに画像を取得するためのシーケンス
VSet seq1.RuntimeAcquire, VISION_ACQUIRE_STATIONARY
VSet seq1.ImageBuffer, 1
VRun seq1 ' バッファ1 へ画像を取り込む
Go Image2Pos
VSet seq1.ImageBuffer, 2
VRun seq1 ' バッファ2 へ画像を取り込む
Go Image3Pos
VSet seq1.ImageBuffer, 3
VRun seq1 ' バッファ3 へ画像を取り込む
...
' すでに取得している画像の処理を行う
VSet seq2.RuntimeAcquire, VISION_ACQUIRE_NONE
VSet seq2.ImageBuffer, 1
VRun seq2 ' バッファ1 の画像処理を行う
VGet seq2.AllPassed, allPassed
...
```

画像を取得しない(処理のみを行う)シーケンスでイメージバッファ 1~10を使用すると、VRunの実行中は、Cameraプロパティの値が無視されます。

## 5.9.4 外部トリガーでのイメージ取り込みの使用方法

Vision Guide 7.0は、外部信号によって、ビジョンシーケンスに画像イメージを取得させ、処理させるトリガー入力機能をサポートしています。トリガー入力の使用手順は、次のとおりです。

- (1) トリガー信号をカメラのコネクターへ配線します。ストロボ照明を使用するときは、カメラのストロボ出力信号へ配線することができます。
- (2) トリガー入力を使用するビジョンシーケンスのRuntimeAcquireプロパティを“Strobed”に設定してください。
- (3) SPEL+プログラムで通常どおりVRunを実行して、AcquireStateプロパティが、画像取り込み完了を示す値“3”に変わるまで待ってください。次の例では、外部装置からトリガー信号を送っています。

```
Integer state
Boolean found

#define PICTURE_DONE 3
TmrReset 0
VRun seq1
Do
    Wait 0.01
    VGet seq1.AcquireState, state
    If Tmr(0) > 10 Then
        Error ER_STROBE_OT ' User defined overtime error
    EndIf
Loop Until state = PICTURE_DONE
VGet seq1.obj1.Found, found
```

- AcquireStateで画像の取り込み完了を待たなかった場合、自動的にタスクの次のビジョンコマンドは画像が取得されるまで待つから実行されます。この場合、処理を継続する前に画像を取得しなければなりません、取得できない場合はタスクを中止しなければなりません。このため、トリガーが入力されず、画像が取得できなくてもプログラムを継続できるようにAcquireStateを確認することを推奨します。

- Vision Guide 7.0 GUIからトリガー入力を使用しているシーケンスを実行するとき、システムはトリガーが入力されるまで待ちます。＜中断＞ボタンをクリックすると、トリガー入力待ちを中断できます。



注意

- ビジョンシーケンスの実行結果は、外らん光や外部機器ノイズにより影響を受けることがあります。  
外らん光や外部機器ノイズにより、影響を受けると、取り込み画像が想定外の結果となり、検出位置結果がサーチエリア (検出エリア)内の不特定位置になる可能性があります。  
サーチエリアは、極力小さく、また誤検出に配慮した画像処理シーケンスの作成を行ってください。

## 5.9.5 カラー画像の使用方法

カラーカメラを設定すると、カラー画像を取得したりディスクから読み込むことができます。

ビジョンツールには、カラー画像を処理できるものとグレースケール画像のみに対応しているものがあります。下表を参照してください。

ビジョンツール	カラー処理	グレースケール処理
Blob		×
Correlation		×
Geometric		×
Edge		×
Polar		×
Code Reader		×
OCR		×
ImageOp	×	×
ColorMatch	×	×
LineFinder		×
LineInspector		×
ArcFinder		×
ArcInspector		×
DefectFinder		×
BoxFinder		×
CornerFinder		×
Contour		×

ImageOpツールには、カラー画像を処理するColorFilter操作とColorStretch操作がありますが、その他のImageOp操作はグレースケール画像を使用します。

ColorMatchツールは通常、カラー画像を処理するために使用されますが、グレースケール画像にも使用できます(グレイに濃淡があります)。

カラー画像が取得されると、グレースケール画像が必要なツールで使用するため、内部にグレースケール画像も作成されます。シーケンスが実行されると、カラー処理を実行するオブジェクトがカラー画像を使用し、グレースケール処理を実行するオブジェクトがグレースケール画像を使用します。

カラー画像には、赤、緑、青の色相があります。ImageColorシーケンスプロパティを使って取得する相を選択します。デフォルトでは、Allが選択され、3つの色相を使用したフルカラー画像が取得できます。

また、赤、緑、青、グレースケールを選択することもできます。赤、緑、青を選択すると、グレースケール画像は単色カラー画像になります。グレースケール処理ツールを使用して、ひとつの色相を検索することができます。

Vision Guideウィンドウの<Run Object>ボタンをクリックしたとき、使用しているオブジェクトがグレースケール処理ツールなら、ツールが処理中に使用するグレースケールでビデオ画像が表示されます。

## 6. ビジョンオブジェクト

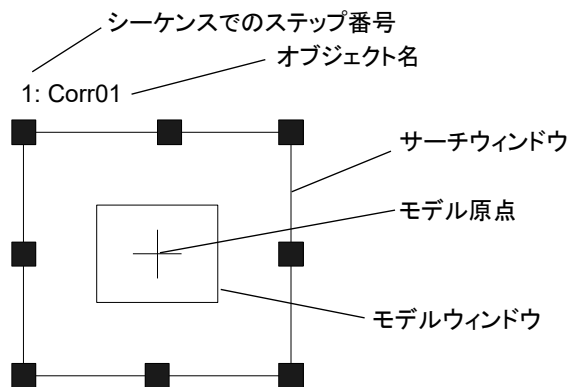
### 6.1 ビジョンオブジェクトの基本事項

ビジョンオブジェクトの詳細を説明する前に、ビジョンオブジェクトのレイアウトとビジョンオブジェクトを操作するときの位置決め サイズ設定方法について、基本的事項を簡単に説明します。

新規ビジョンオブジェクトが作成されて画像イメージ表示部に表示されると、特定のビジュアル特性が付与されますので、ぜひ覚えておいてください。

例えば、Correlationオブジェクトは、サーチウィンドウ、モデルウィンドウ、モデル原点とともに表示されます。これらは、マウスをドラッグすることによって、簡単に表示位置を変更することができます。

以下に、基本的なビジョンオブジェクトとその操作方法について説明します。



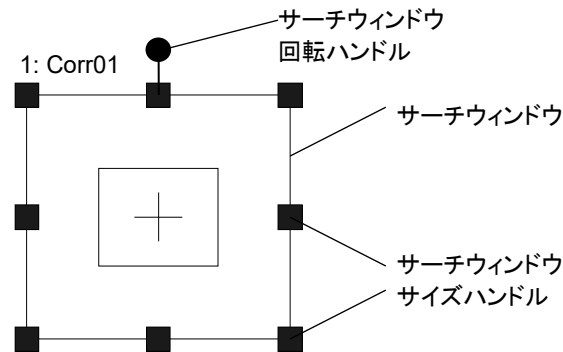
ビジョンオブジェクトのレイアウト例

#### 6.1.1 サーチウィンドウ

ほとんどのビジョンオブジェクトには、次図に示すようなサーチウィンドウがあります。サーチウィンドウは、上図の外側のボックスです。アクティブ(現在作業しているビジョンオブジェクト)のときは、マゼンタ色(明るい紫色)になり、インアクティブ(シーケンスにあるけれども今のところ作業はしていない、別のビジョンオブジェクト)のときは、青色になります。実行時には、オブジェクトが検出されたかどうかによって、緑色あるいは赤色に変わります。

サーチウィンドウは、対象領域(あるいはパーツあるいはパターンを検索する領域)を定義するのに使います。

サーチウィンドウの移動は、ビジョンオブジェクト名(またはサーチウィンドウ)をクリックし、画像イメージ表示部の希望の位置にドラッグすることによって簡単に実行できます。(図中の“Corr01”がビジョンオブジェクトの名称です。)



サーチウィンドウのサイズは、ビジョンオブジェクトをコンフィギュレーションするときに重要なポイントとなります。サーチウィンドウを大きくすると、検索に要する時間が長くなります。サーチウィンドウは、なるべく小さく、しかも、パーツの様々な位置を取りあつかうことができる程度に十分な大きさのサイズに設定します。



SearchWinTypeプロパティを“RotatedRectangle”や“Circle”に設定することで、サーチウィンドウに角度を持たせたり、円形のサーチエリアを設定したりすることも可能です。

「6.2.10 Frameオブジェクト」を使用すると、サーチウィンドウのサイズを小さくするのに便利な場合があります。

### サーチウィンドウの位置決め

サーチウィンドウの位置決めは、ビジョンオブジェクト名をクリックし、移動したい位置にビジョンオブジェクトをドラッグして行います。

ビジョンオブジェクトを移動するときは、左上コーナーが適切な位置にくるように注意してください。

### サーチウィンドウのサイズ調整

サイズは、サーチウィンドウサイズハンドルを使って調整します。サーチウィンドウサイズハンドルは、ビジョンオブジェクトの初期作成時、またはビジョンオブジェクトをクリックして選択したとき表示されます。

サーチウィンドウサイズハンドルは、サーチウィンドウの辺とコーナーに表示される小さなボックスであり、簡単に動かしてサーチウィンドウのサイズを調整することができます。

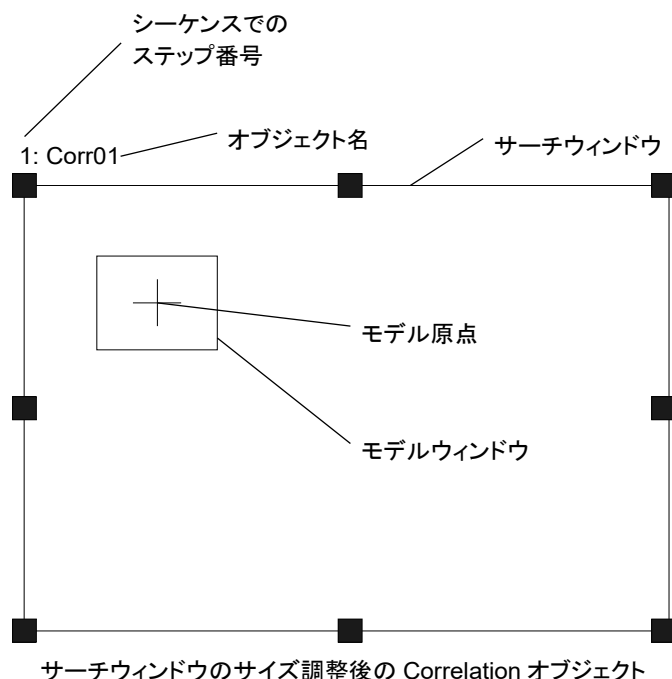
サーチウィンドウサイズハンドルの上でマウスポインターを動かすと、マウスポインターは両方向矢印に変わります。このときマウスをクリックすると、そのハンドルが選択されます。

どのサーチウィンドウサイズハンドルもクリックすることができますので、辺やコーナーをドラッグして、サーチウィンドウのサイズを調整してください。

サーチウィンドウの両側の辺にあるサーチウィンドウサイズハンドルをクリックすると、オブジェクトを水平方向にサイズ調整することができます。

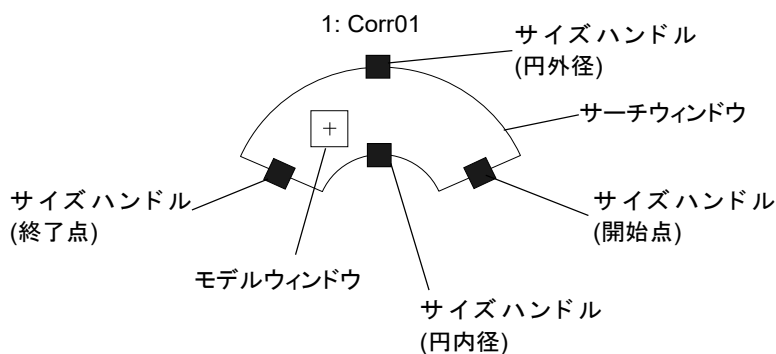
サーチウィンドウの上下にあるサーチウィンドウサイズハンドルをクリックすると、オブジェクトを垂直方向にサイズ調整することができます。

サーチウィンドウのコーナーにあるサーチウィンドウサイズハンドルをクリックすると、サーチウィンドウの水平、垂直の両方向のサイズ調整を一度に行うことができます。



サイズ調整後のサーチウィンドウ

サーチウィンドウの形は、SearchWinTypeプロパティで設定できます。  
 “Rectangle”、“RotatedRectangle”、“Circle”の他に、“Arc”、“Polygon”が設定できます。  
 “Arc”、“Polygon”は、サイズハンドルで調整できるプロパティや、使用できるプロパティ  
 が、他の3タイプとは異なります。  
 “Arc”に設定することで、円弧型のサーチエリアを設定することが可能です。



### サーチウィンドウの位置決め

サーチウィンドウの位置決めは、ビジョンオブジェクト名をクリックし、移動したい位置にビジョンオブジェクトをドラッグして行います。

ビジョンオブジェクトを移動するときは、中心点が適切な位置にくるように注意してください。



### サーチウィンドウのサイズ調整

サイズは、サーチウィンドウサイズハンドルを使って調整します。“Arc”型のサーチウィンドウでは、サーチウィンドウサイズハンドルは、サーチウィンドウ円弧の円外径と円内径の中心、円弧の開始点と終了点に表示される小さなボックスであり、簡単に動かしてサーチウィンドウのサイズを調整することができます。

サーチウィンドウサイズハンドルの上でマウスポインターを動かすと、マウスポインターは両方向矢印に変わります。このときマウスをクリックすると、そのハンドルが選択されます。

円弧の開始点と終了点にあるサーチウィンドウサイズハンドルをクリックすると、サーチウィンドウの円弧の長さをサイズ調整することができます。

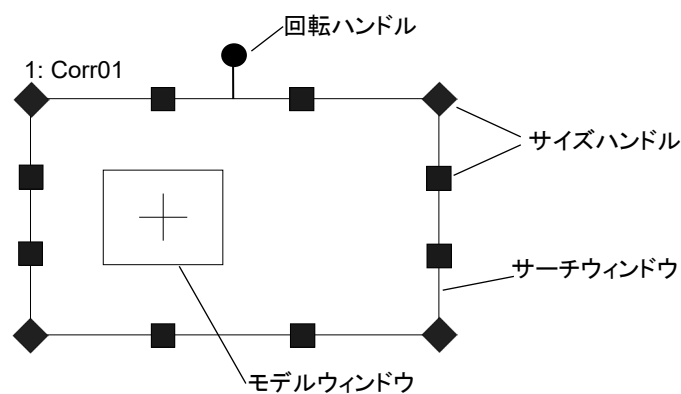
円外径と円内径の中心にあるサーチウィンドウサイズハンドルをクリックすると、サーチウィンドウの厚さをサイズ調整することができます。

#### NOTE



“Arc”型のサーチウィンドウは、“Rectangle”、“RotatedRectangle”、“Circle”型のサーチウィンドウの形状で使用できた、SearchWinLeftプロパティ、SearchWinTopプロパティ、SearchWinHeightプロパティ、SearchWinWidthプロパティが、使用できなくなります。代わりにSearchWinAngleStartプロパティ、SearchWinAngleEndプロパティ、SearchWinRadiusInnerプロパティ、SearchWinRadiusOuterプロパティが使用できるようになります。SearchWinCenterXプロパティ、SearchWinCenterYプロパティも使用できるので、合わせて6つのプロパティを調整することで、円弧の位置、サイズを表現できます。

SearchWinTypeを“Polygon”に設定することで12角形型のサーチエリアを設定できます。他の4タイプと比べ多くのプロパティを持ちますが、自由度の高い形状を表現できます。



### サーチウィンドウの位置決め

サーチウィンドウの位置決めは、ビジョンオブジェクト名をクリックし、移動したい位置にビジョンオブジェクトをドラッグして行います。

ビジョンオブジェクトを移動するときは、中心点が適切な位置にくるように注意してください。

サーチウィンドウのサイズ調整

サイズは、サーチウィンドウサイズハンドルを使って調整します。“Polygon”型のサーチウィンドウでは、サーチウィンドウサイズハンドルは、12角形の各頂点に表示される小さなボックスであり、簡単に動かしてサーチウィンドウのサイズ、形状を調整することができます。

サーチウィンドウサイズハンドルの上でマウスポインターを動かすと、マウスポインターは両方向矢印に変わります。このときマウスをクリックすると、そのハンドルが選択されます。

“Rectangle”や、“RotatedRectangle”型のサーチウィンドウでは、サイズハンドルはサーチウィンドウの幅、高さ、もしくはその両方を調整するためのハンドルです。よって選択したハンドルを移動させると、操作に応じて他のサイズハンドルも一緒に移動します。しかし、“Polygon”型のサーチウィンドウでは、選択したハンドルのみが移動し、他のハンドルは移動しません。これにより自由度の高いサーチエリアの形状を調整できます。



“Polygon”型のサーチウィンドウは、“Rectangle”, “RotatedRectangle”, “Circle”型のサーチウィンドウの形状で使用できた、SearchWinLeftプロパティ、SearchWinTopプロパティが使用できなくなります。

代わりに各頂点の座標を表現するSearchWinPolygonPointX1プロパティ、SearchWinPolygonPointY1プロパティ～SearchWinPolygonPointX12プロパティ、SearchWinPolygonPointY12プロパティが使用できるようになります。SearchWinAngleプロパティ、SearchWinCenterXプロパティ、SearchWinCenterYプロパティ、SearchWinHeightプロパティ、SearchWinWidthプロパティも使用できるので、合わせて29のプロパティを調整することで、サーチエリアの位置、サイズ、形状を表現できます。



■ ビジョンシーケンスの実行結果は、外らん光や外部機器ノイズにより影響を受けることがあります。

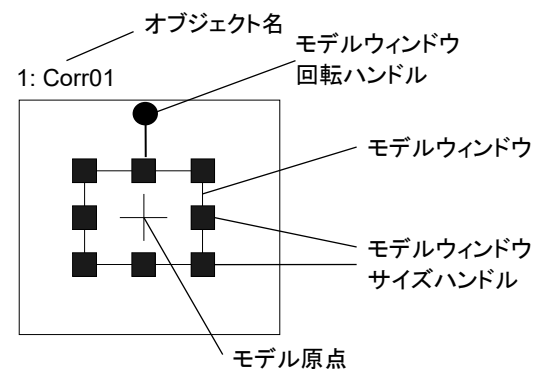
外らん光や外部機器ノイズにより、影響を受けると、取り込み画像が想定外の結果となり、検出位置結果がサーチエリア (検出エリア)内の不特定位置になる可能性があります。

サーチエリアは、極力小さく、また誤検出に配慮した画像処理シーケンスの作成を行ってください。

### 6.1.2 モデルウィンドウ

Correlationオブジェクト, Geometricオブジェクト, Polarオブジェクトには、モデルウィンドウとサーチウィンドウがあります。この解説ではCorrelationオブジェクトのモデルウィンドウを使います。

Correlationオブジェクトには、図に示すようなモデルウィンドウが備わっています。モデルウィンドウは、図に示すように、内側のボックスでマゼンタ色をしています。(オブジェクトを検出したときに緑色に変わる実行時はのぞきます。) モデルウィンドウは、モデルの境界を定義するのに使います。(モデルとは、対象物の理想的表現であり、一般的には検索対象のことです。)



#### モデルウィンドウ

##### モデルウィンドウの位置決め

新規モデルを教示するには、モデルのサイズと位置の設定が非常に重要なので、モデルウィンドウの位置を設定してサイズ調整をする方法を正確に理解する必要があります。

モデルウィンドウを移動させるときは、モデルウィンドウを構成する4本の辺の1本をクリックしてドラッグします。まず、左上コーナーが適切な位置にくるよう移動させ、次に、モデルウィンドウサイズハンドルを使ってサイズを調整するようにしてください。

サーチウィンドウと同様、モデルウィンドウのサイズは、ビジョンオブジェクトをコンフィギュレーションするとき重要なポイントになります。モデルウィンドウを大きくすると、検索に要する時間が長くなります。モデルウィンドウは、なるべく小さくしながら、しかも、検索するパーツやパターンを見分けられる程度に十分な大きさのサイズに設定することをお勧めします。同時に、サーチウィンドウも同じように小さくしてください。

#### ヒント

ModelWinTypeプロパティを“RotatedRectangle”や“Circle”に設定することで、モデルウィンドウに角度を持たせたり、円形のモデルウィンドウを設定することも可能です。

### モデルウィンドウのサイズ調整

モデルウィンドウサイズハンドルは、モデルウィンドウを選択したときに表示されます。

ただし、サーチウィンドウが選択されていると、モデルウィンドウサイズハンドルは表示されません。モデルウィンドウにハンドルが表示されないときは、サーチウィンドウがアクティブウィンドウになっている場合がありますので、モデルウィンドウの辺をクリックしてモデルウィンドウに切り換えてください。モデルウィンドウサイズハンドルが表示されます。

モデルウィンドウサイズハンドルは、モデルウィンドウの辺とコーナーに表示される小さなボックスであり、簡単に動かしてモデルウィンドウのサイズを調整することができます。モデルウィンドウサイズハンドルの上でマウスポインターを動かすと、マウスポインターは両方向矢印に変わります。このときマウスをクリックすると、そのハンドルが選択されます。

どのモデルウィンドウサイズハンドルもクリックすることができますので、辺やコーナーをドラッグして、モデルウィンドウのサイズを調整してください。

モデルウィンドウの両側の辺にあるモデルウィンドウサイズハンドルをクリックすると、オブジェクトを水平方向にサイズ調整することができます。

モデルウィンドウの上下にあるモデルウィンドウサイズハンドルをクリックすると、オブジェクトを垂直方向にサイズ調整することができます。

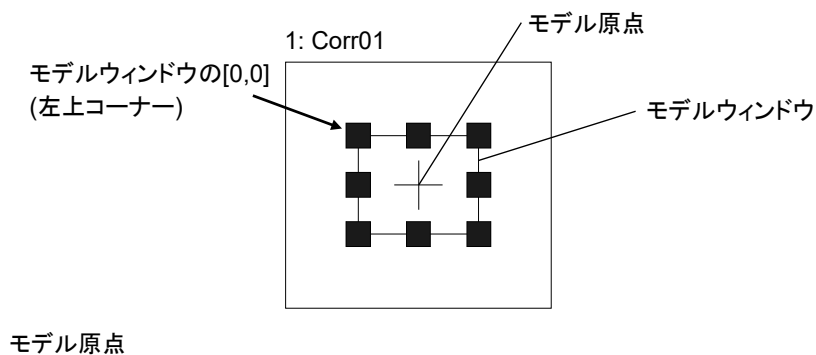
モデルウィンドウのコーナーにあるモデルウィンドウサイズハンドルをクリックすると、モデルウィンドウの水平、垂直の両方向のサイズ調整を一度に行うことができます。

#### 6.1.3 モデル原点

すべてのモデルにはモデル原点があります。モデル原点とは、画像イメージ中のモデルの位置を示す固定基準点です。

モデル原点の座標位置は、モデルウィンドウの左上の位置を基準点としますので注意してください。

モデルウィンドウの左上の位置は、[0,0]から始まります。



### モデル原点の位置決め

モデル原点の位置は、自動的に、あるいは手動で設定することができます。

モデル原点の位置を自動的に設定するには、対象となるオブジェクトの **ModelOrgAutoCenter** プロパティを “True” に設定してください。この場合、モデル原点の位置はモデルウィンドウの中央に自動的に設定されます。

NOTE



**ModelOrgAutoCenter** プロパティのデフォルト設定は、“True” になっています。

モデル原点を手動で移動するには、**ModelOrgAutoCenter** プロパティを “False” に設定してください。特定のオブジェクトのオブジェクトウィンドウから、プロパティリスト内で設定することができます。プロパティリスト内で設定する場合、設定値としてサブピクセルで設定することができます。

NOTE



サブピクセルでモデル原点を設定した場合でも、画面に表示されるモデル原点はピクセル単位で表示されます。

モデル原点を移動するには、対象となるオブジェクトのモデルウィンドウがアクティブになっていなければなりません。サーチウィンドウがアクティブになっている場合は、モデル原点の移動ができませんので、モデルウィンドウの枠をクリックし、モデルウィンドウにモデルウィンドウサイズハンドルを表示させてください。

**ModelOrgAutoCenter** プロパティが “False” に設定されていると、モデル原点はマウスで移動することができます。モデル原点を移動するには、モデル原点の十字線をクリックして、移動したい位置へドラッグしてください。

NOTE



モデル原点をマウスで移動した場合、設定されるモデル原点はピクセル単位となります。サブピクセルで設定する場合は、プロパティリスト内で設定してください。

モデル原点は、作業対象となるモデル上の有意性のある位置に設定すると便利です。例えば、ロボットがパーツをつかみ、ピックアップする位置に、モデル原点の位置を設定します。

## 6.2 ビジョンオブジェクトの使い方

ここでは、さまざまなビジョンオブジェクトのレイアウトや操作方法の違いについて説明します。



注意

- ビジョンシーケンスの実行結果は、外らん光や外部機器ノイズにより影響を受けることがあります。  
外らん光や外部機器ノイズにより、影響を受けると、取り込み画像が想定外の結果となり、検出位置結果がサーチエリア (検出エリア)内の不特定位置になる可能性があります。  
サーチエリアは、極力小さく、また誤検出に配慮した画像処理シーケンスの作成を行ってください。

### 6.2.1 ImageOp オブジェクト (画像操作)

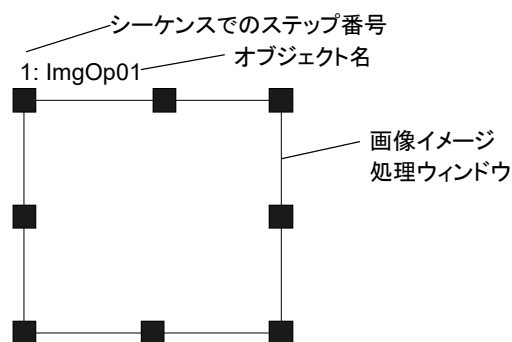
#### ImageOp オブジェクトの概要

ImageOpオブジェクトは、指定された領域について、形態を変える (広げる, ふさぐ, 縮小, 膨張), コンボリューション (鮮明にする, 滑らかにする), ひっくり返す, 2値化する, 回転することができます。

ImageOp領域にある他のビジョンオブジェクトは、ImageOpのアウトプット上でその操作を実行します。例えば、ImageOpツールを使って、ビデオ画像イメージ全体に縮小操作を実行して、それからBlobオブジェクトをImageOpサーチウィンドウ内に置いて、縮小された画像イメージを検索することができます。

#### ImageOp オブジェクトのレイアウト

ImageOpオブジェクトには、下に示すように画像イメージ処理ウィンドウがあります。



ImageOp オブジェクトのレイアウト

## ImageOpオブジェクトのプロパティー

ImageOpオブジェクトで使われるプロパティーの一覧を次に示し、簡単に説明します。各プロパティーの詳細は、Vision Guide 7.0 プロパティー & リザルトリファレンスを参照してください。

プロパティー	説明
AbortSeqOnFail	オブジェクトの実行に失敗した (不良判定になった) とき、直ちにシーケンス全体を中断し、それ以降のオブジェクトを処理しないように指定
AngleObject	どのオブジェクトから自動的に回転するかを定義 画像イメージは、指定されたオブジェクトのAngleリザルトを使って、反時計回りに回転します。Operationプロパティーは、Rotationに設定してください。設定しないと、このプロパティーは無視されます。 デフォルト: None
AngleObjectResult	AngleObject プロパティーが使用するリザルトを指定
Caption	見出しをImageOpオブジェクトに割りあてる デフォルト: 空欄
ColorMode	カラー処理に使用する色空間(RGB/HSV)を設定 デフォルト: RGB
CurrentModel	実行時のみ Operation が ColorFilter に設定されている場合、ModelColor プロパティーや VTeach で、使用するモデルを指定 CurrentModel = 0 の場合、背景は設定されています。 デフォルト: 0
Description	ユーザー指定の説明を設定 デフォルト: 空欄
EditWindow	被検索領域の検出マスクを定義
Enabled	このオブジェクトを実行するかを指定 デフォルト: True
FailColor	オブジェクトの検出不良時の色を選択
Frame	どの位置に置いたフレームを使用するかを指定 デフォルト: None
FillHoles	2値画像の穴を埋めるかを指定 OperationをBinarizeに設定したときのみ表示されます。 デフォルト: False
FrameResult	Frameの何番目のリザルトを使用するかを指定 デフォルト: 1
Graphics	表示するグラフィックを指定 デフォルト: 1 – All

プロパティ	説明
ImageBuffer1	OperationプロパティをSubtractAbsに設定したときの、第1イメージをバッファ番号で指定  SubtractAbsは、(第1イメージ - 第2イメージ)の絶対値画像を算出します。
ImageBuffer1File	ImageBuffer1を“File”に設定した時の、バッファに格納する画像ファイルのパスを指定  デフォルト: None
ImageBuffer2	OperationプロパティをSubtractAbsに設定したときの、第2イメージをバッファ番号で指定
ImageBuffer2File	ImageBuffer2を“File”に設定した時の、バッファに格納する画像ファイルのパスを指定  デフォルト: None
Iterations	指定された操作を実行する回数を定義 デフォルト: 1
KeepRGBRatio	処理にRGB比率を維持するかを指定 デフォルト: True
LabelBackColor	オブジェクトラベルの背景の色を設定 デフォルト: Transparent (透明)
MaxRGB	ColorStretch処理で使用する色の最大値を定義 デフォルト: 255, 255, 255
MinRGB	ColorStretch処理で使用する色の最小値を定義 デフォルト: 0, 0, 0
ModelColor	実行時のみ OperationプロパティがColorFilterに設定されている場合、実行時にRGB色の値を直接入力し、モデルや背景色を教示 デフォルト: RGB (0, 0, 0)
ModelColorTol	実行時のみ モデル色の許容差を設定 画素色が許容差の範囲内であれば、画素は変化しません。 デフォルト: 10 (ColorMode = RGB), 0,0,50 (ColorMode = HSV)
ModelName	実行時のみ 現在使用しているモデルの名前を設定
ModelWin	実行時のみ 1コールでモデルウィンドウの左端, 上端, 高さ, 幅のパラメーターを設定、または返す
ModelWinHeight	モデルウィンドウの高さを定義
ModelWinLeft	モデルウィンドウの最左端の位置を定義
ModelWinTop	モデルウィンドウの最上端の位置を定義
ModelWinWidth	モデルウィンドウの幅を定義



プロパティ	説明
Name	ImageOpオブジェクトに独自のオブジェクト名を割りあてる デフォルト: ImgOp01
NumberOfModels	実行時のみ 使用しているカラーモデル数 実行中、このプロパティを設定すると、CurrentModel, VTeach を 使用し、カラーモデルを教示できます。 デフォルト: 1
Operation	画像処理の種類を設定  Operationプロパティの詳細は、Vision Guide 7.0 プロパティ& リザルトリファレンスを参照してください。 デフォルト: Open
PassColor	オブジェクトの検出良時の色を選択 デフォルト: LightGreen
PassType	オブジェクト検出時の良否判定の条件を定義
Polarity	オブジェクトと背景の差 (明るい背景に暗いオブジェクトか、暗い背 景に明るいオブジェクトか)を定義
RotationAngle	操作プロパティをRotationに設定したとき、画像イメージを回転す る角度を定義 デフォルト: 0
RotationDirection	回転操作時の、回転方向を指定
SearchWin	実行時のみ 1コールでサーチウィンドウの左端, 上端, 高さ, 幅のパラメーターを 設定、または返す
SearchWinAngle	被検索領域の角度を定義
SearchWinCenterX	被検索領域の中心のX座標値を定義
SearchWinCenterY	被検索領域の中心のY座標値を定義
SearchWinHeight	被検索領域の高さを定義 (単位: ピクセル) デフォルト: 100
SearchWinLeft	被検索領域の最左端の位置を定義 (単位: ピクセル)
SearchWinTop	被検索領域の最上端の位置を定義 (単位: ピクセル)
SearchWinType	被検査領域のタイプ (Rectangle, RotatedRectangle, Circle)を定義
SearchWinWidth	被検索領域の幅を定義 (単位: ピクセル) デフォルト: 100
ShiftObject	Shift処理を行うオブジェクトを設定
ShiftX	X方向のShift量を設定
ShiftY	Y方向のShift量を設定
ThresholdAuto	対象物 (オブジェクト)、背景、画像イメージのエッジを表すグレイレ ベルのしきい値を自動設定するか否かを設定

プロパティ	説明
ThresholdBlockSizeOperation	プロパティをBinarizeAdaptiveに設定したときに使用する、しきい値を決定するために近傍領域を参照する範囲を定義 デフォルト: 1/16ROI
ThresholdColor	しきい値の範囲内にあるピクセルの色を定義 デフォルト: 黒色
ThresholdHigh	OperationプロパティをBinarizeに設定したときに使用する、しきい値を高く維持する設定を定義 デフォルト: 128
ThresholdLevel	OperationプロパティをBinarizeAdaptiveに設定したときに使用する、近傍領域との輝度差の割合を定義 デフォルト: 15%
ThresholdLow	OperationプロパティをBinarizeに設定したときに使用する、しきい値を低く維持する設定を定義 デフォルト: 0
ThresholdMethod	2値化の処理方法を設定
ZoomFactor	OperationプロパティをZoomに設定したときに使用する、ズーム値を設定 デフォルト: 1

### ImageOpオブジェクトリザルト


下の一覧は、ImageOpオブジェクトリザルトを簡単に説明した概略です。各リザルトの詳細は、Vision Guide 7.0プロパティ&リザルトリファレンスを参照してください。

リザルト	説明
Passed	検出結果が良か否かを返す
Time	オブジェクト処理に要した時間を返す (単位: ミリ秒)
FocusValue	相対的なフォーカスレベルを返す 最小値になったときに最適なフォーカスです。 OperationプロパティをDetectFocusに設定したときのみ表示されます。


### ImageOpオブジェクトの使用方法

これまでVision Guide ImageOpオブジェクトの使い方を理解するための基礎を説明してきました。このセクションで説明するImageOpオブジェクトを使うのに必要な手順は、次のとおりです。

- 新規ImageOpオブジェクトの作成方法
- サーチウィンドウの位置とサイズの設定
- ImageOpオブジェクトに関連するプロパティの設定
- ImageOpオブジェクトのテストとリザルトの確認
- プロパティの調整と再テスト

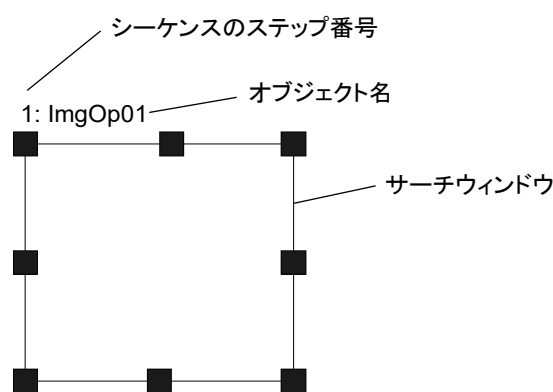
次の手順を始める前に、新規ビジョンシーケンスを作成するか、または使用するビジョンシーケンスを選択しておいてください。作業するビジョンシーケンスがなければ、 <新規シーケンス>ボタンをクリックして、新しいビジョンシーケンスを作成することができます。Vision Guideウィンドウの[シーケンス]タブをクリックしてから、[シーケンス]タブの上部にあるドロップダウンリストボックスをクリックして、以前に作成したシーケンスを選択することもできます。新規ビジョンシーケンスを作成する、または以前に定義したシーケンスを選択する方法の詳細は、「5.2 ビジョンシーケンスの新規作成」を参照してください。

### Step 1: 新規ImageOpオブジェクトの作成

- (1) Vision Guideツールバー - <全ツール> -  <ImageOp (画像操作)>ボタンをクリックします。
- (2) マウスを画像イメージ表示部に動かしてください。マウスポインターがImageOpオブジェクトアイコンに変わります。
- (3) 画像イメージ表示部の目的の位置にアイコンが来るまで、マウスを動かし続けてください。それから、マウスを左クリックして、オブジェクトを作成します。
- (4) オブジェクト名は自動的に作成されます。この例では、このシーケンスで最初に作成されたImageOpオブジェクトなので、“ImgOp01”となります。(名前の変更方法は、後述します。)

### Step 2: サーチウィンドウの位置とサイズの設定

次に示すようなImageOpオブジェクトがあります。



新規ImageOpオブジェクトのレイアウト

- (1) ImageOpオブジェクトの名前ラベルをクリックして、ボタンを押さえたまま、ImageOpオブジェクトをサーチウィンドウの左上に来る位置にドラッグしてください。
- (2) サーチウィンドウサイズハンドルを使って、ImageOpオブジェクトサーチウィンドウのサイズを変えてください。マウスポインターをサイズハンドルに動かして、それからマウスの左側ボタンを押さえながら、ハンドルをドラッグして、ウィンドウの大きさを変えます。

### Step 3: ImageOpオブジェクトプロパティの設定

ImageOpオブジェクトプロパティの設定を行います。よく使われるImageOpオブジェクトに特有のプロパティを、いくつか下に説明します。別のビジョンオブジェクトで使われる、Graphicsなど他のプロパティの説明は、「Vision Guide 7.0 プロパティ&リザルトリファレンス」、またはImageOpプロパティ一覧にあります。

**Name  
プロパティ** 新規に作成されたImageOpオブジェクトに、デフォルトとして与えられるオブジェクト名は“ImgOp\*\*”です。\*\*は、1つのビジョンシーケンス内で複数のImageOpオブジェクトを用いるとき、それらのImageOpオブジェクトを区別するための番号です。

ビジョンシーケンスで1番目に用いられるImageOpオブジェクトには、“ImgOp01”というオブジェクト名がデフォルトで設定されます。

オブジェクト名を変更したい場合は、Nameプロパティの設定値フィールドをクリックして、変更後の名称を入力し、リターンキーを押してください。画面上でImageOpオブジェクト名を表示する場所は、すべて変更後の名称に変わります。

**Operation  
プロパティ** どの画像イメージ操作を実行するかを決定  
ImageOpオブジェクトにもっとも重要なプロパティです。

**Iterations  
プロパティ** 反復する回数を決定

**Polarity  
プロパティ** 暗いオブジェクト、それとも明るいオブジェクトで操作を実行するかを決定  
デフォルト設定は暗いオブジェクトです。変更する場合は、PolarityプロパティのValueフィールドをクリックしてください。選択肢が2つ(“DarkOnLight”と“LightOnDark”)のドロップダウンリストがあります。使用する選択肢をクリックしてください。

ImageOpオブジェクトをテストして、後で必要な他のプロパティを設定することもできます。

#### Step 4: ImageOp オブジェクトカラーの教示

Operation プロパティが “ColorFilter” に設定されている場合、背景とともにフィルターにかける色をひとつ以上ティーチしてください。Vision Guide ウィンドウの<ティーチ>ボタンが有効になり、ImageOp メインウィンドウに長方形のモデルウィンドウが表れます。

<ティーチ>ボタンをクリックすると、次のような画面が表示されます。最初は、モデルがない状態になります。



<追加>ボタンをクリックすると、新しいフィルターカラーがデフォルト色(黒)で追加されます。背景色、または、フィルターカラーをひとつ選択し、<ティーチ>ボタンをクリックすると、モデルウィンドウにある画素の平均色が使用されます。または、RGB 値を直接入力することもできます。

モデルウィンドウを開いている間、ウィンドウのサイズと位置を変更することができます。色を追加したり、サイズや位置を変更した後、画面を閉じずに<ティーチ>ボタンをクリックすると、新しい色を教示します。

ColorMode プロパティは Teach ダイアログが表示されていても変更できます。

ColorMode が RGB の場合、各色の Tolerance の値は 1 つです。ColorMode が HSV の場合、各色の Tolerance の値は 3 つです。(hTol, sTol, vTol)

<削除>ボタンは、フィルターカラーを削除します。背景色は削除できません。

#### Step 5: ImageOp オブジェクトのテスト

ImageOpオブジェクトを実行するには、[オブジェクト]タブの左下にある<オブジェクト実行>ボタンをクリックしてください。画像イメージに与えるImageOpツールの効果がわかります。

### Step 6: プロパティの調整と再テスト

ImageOpオブジェクトを何回か実行したあと、問題を見つけたり、プロパティ設定を微調整する必要があるかもしれません。一般的な微調整の方法について次に説明します。

アプリケーションによっては、ImageOpオブジェクトの微調整が必要となる場合があります。ImageOpオブジェクトの微調整に関する主要なプロパティについて、次に説明します。

Iterations	目的の画像イメージ操作の実行回数を決定
ThresholdColor, ThresholdLow, ThresholdHigh, ThresholdAuto	Binarize操作のパラメーターを調整 「Vision Guide 7.0 プロパティ&リザルトリファレンス」の「ThresholdColor プロパティ」、「ThresholdLow プロパティ」、「ThresholdHigh プロパティ」、「ThresholdAuto プロパティ」の説明を参照してください。

プロパティの調整が完了し、ImageOpのテストを行って満足なリザルトを得ることができたら、ビジョンオブジェクトの作成は完了です。他のビジョンオブジェクトを作成したり、ビジョンシーケンス全体の設定やテストを行う次のステップに進むことができます。

### 6.2.2 Geometricオブジェクト (幾何学的形状)

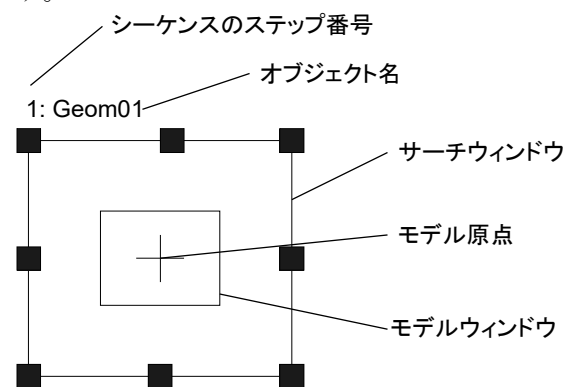
#### Geometricオブジェクトの概要

Geometricオブジェクトは、幾何学的形状対象物に基づいて、モデルを検出します。このオブジェクトは、画素間相関の代わりに、エッジを基準にした幾何学的形状対象物でモデルを検出します。Geometricオブジェクトは、スケールや角度の変化量に加えて、照明変化量(鏡面反射を含む)、モデルに対する許容誤差が大きいなど、相関パターンマッチングより役に立つ点があります。

通常、Geometricオブジェクトは、オブジェクトの対象物を見つけて、位置と方向を決定するのに使われます。これは、ロボットをピックアップ位置や配置位置に動かすために、パーツ位置を検出するのによく使われます。

#### Geometricオブジェクトのレイアウト

Geometricオブジェクトには、下に示すようにサーチウィンドウ、モデルウィンドウがあります。



## Geometricオブジェクトのプロパティー

下の一覧は、Geometricオブジェクトのプロパティーを簡単に説明した概略です。各プロパティーの詳細は、「Vision Guide 7.0 プロパティー&リザルトリファレンス」を参照してください。

プロパティー	説明
AbortSeqOnFail	オブジェクトの実行に失敗した (不良判定になった)とき、直ちにシーケンス全体を中断し、それ以降のオブジェクトを処理しないように指定 デフォルト: False
Accept	対象物が検出されたと判断するための形状スコアの値を指定  対象物のスコアが設定値を超えているものを検出します。設定値を低く設定すると、誤検出の原因となります。 デフォルト: 700
AngleEnable	Geometricサーチが角度検出付き (回転)サーチを実行するかどうかを指定  Geometricオブジェクトのモデル登録の前に指定されます。
AngleOffset	回転のオフセット値を設定 デフォルト: 0.000
AngleRange	一連の回転モデルの登録範囲を指定
AngleStart	角度サーチの中心を指定
CalRobotPlacePos	設計時とプログラム実行時に RobotPlacePos をキャリブレーションする
Caption	見出しをGeometricオブジェクトに割りあてる
CenterPointObject	オブジェクトの中心点として用いられる位置を指定  このプロパティーを“Screen”に設定すると画面上の任意の位置にオブジェクトを配置できますが、他のビジョンオブジェクトを指定した場合、そのオブジェクトの PixelX, PixelYリザルトに中心点が設定されます。
CenterPntObjResult	CenterPointObjectプロパティーからどのリザルトを使うのかを指定  Allを指定すると、指定されたビジョンオブジェクトリザルトのすべて (NumberFound)にGeometricオブジェクトが適用されます。 デフォルト: 1
CenterPntOffsetX	サーチウィンドウの中心をCenterPointObjectプロパティーで位置決めしたあとに、Xオフセットを設定、または返す



プロパティ	説明
CenterPntOffsetY	サーチウィンドウの中心をCenterPointObjectプロパティで位置決めしたあとに、Yオフセットを設定、または返す
CenterPntRotOffset	中心点の XY オフセット値 (CenterPntOffsetX, CenterPntOffsetY)を、CenterPointObject の Angle リザルトにしたがって回転するか否かを指定  SearchWinType が RotatedRectangle に設定されている場合、サーチウィンドウは Angle リザルトにしたがって回転します。  デフォルト: False
CheckClearanceFor	クリアランス (隙間)を確認するオブジェクトを設定
ClearanceCondition	クリアランス (隙間)の判断方法を指定
Confusion	検出する画像イメージ中に予想される複雑さの度合い量を示す  検索している実際の対象物ではない対象物で得られる形状スコアの最大値です。
CoordObject	リザルトをコピーするCoordinatesオブジェクトを指定 コピー処理はオブジェクトが実行される時に行われ、Decisionの分岐機能などで実行されなかった場合、コピー処理は行われません  デフォルト: None
CurrentResult	オブジェクトウィンドウのリザルトリストに表示するリザルト、または、1つのサーチウィンドウ内にある対象物などの複数のリザルトを、システムが検出する設定のときにデータを返すリザルトを定義
Description	ユーザー指定の説明を設定  デフォルト: 空欄
DetailLevel	検索中にエッジが検出されたとみなされるレベルを選択
EditWindow	被検索領域の検出マスクを定義
Enabled	このオブジェクトを実行するかを指定  デフォルト: True
FailColor	オブジェクトの検出不良時の色を選択
Frame	指定されたフレームに関して、作業中のオブジェクトの検索位置を規定  (Geometricオブジェクトをフレームに関係して位置決めすることができます。)
FrameResult	Frameの何番目のリザルトを使用するかを指定  デフォルト: 1
Graphics	表示するグラフィックを指定

プロパティ	説明
LabelBackColor	オブジェクトラベルの背景の色を設定
ModelObject	検索に使用するモデルを決定
ModelOrgAutoCenter	モデルは固定基準点を持っていて、モデルウィンドウではモデルの位置はその基準点で表わされます。この固定基準点は、モデル原点と呼ばれています。 <b>ModelOrgAutoCenter</b> プロパティによって、モデル原点は、モデルウィンドウの中心に設定できます。
ModelOrgFindCenter	モデル原点を、登録済みモデルのエッジ中心に設定します。
ModelOrgX	モデル原点のX座標値 (サブピクセルで設定可)
ModelOrgY	モデル原点のY座標値 (サブピクセルで設定可)
ModelWin	実行時のみ  1コールでモデルウィンドウの左端, 上端, 高さ, 幅のパラメーターを設定、または返す
ModelWinAngle	モデルウィンドウの角度を定義
ModelWinCenterX	モデルウィンドウの中心のX座標値を定義
ModelWinCenterY	モデルウィンドウの中心のY座標値を定義
ModelWinHeight	モデルウィンドウの高さを定義
ModelWinLeft	モデルウィンドウの最左端の位置を定義
ModelWinTop	モデルウィンドウの最上端の位置を定義
ModelWinType	モデルウィンドウのタイプを定義
ModelWinWidth	モデルウィンドウの幅を定義
Name	Geometricオブジェクトに独自のオブジェクト名を割りあてる デフォルト: Geom01
NumberToFind	作業中のサーチウィンドウで検出するオブジェクトの数を定義  (Geometricオブジェクトは一度に複数のオブジェクトを検出できます。)
PassColor	オブジェクト検出で良判定時の色を選択
PassType	オブジェクト検出時の良否判定の条件を定義 デフォルト: SomeFound
RejectOnEdge	サーチウィンドウのエッジにかかって検出されたパーツを除外するかどうかを決定  通常は、Trueで使用してください。誤検出が発生した結果、サーチウィンドウに接する検出が除外でき、誤検出リスクを軽減できます。
SaveTeachImage	モデルをティーチするときに、画像ファイルに保存するかを設定
ScaleEnable	スケーリングを有効にする

プロパティ	説明
ScaleFactorMax	ScaleTarget値に適用する最大倍率を設定、または返す
ScaleFactorMin	ScaleTarget値に適用する最小倍率を設定、または返す
ScaleTarget	検索するモデルの予想スケールを設定、または返す
ScaleTargetPriority	オブジェクト検出時に、ScaleTarget近傍を優先的に検出するか否かを設定、または返す
ScoreMode	Fail時のリザルトを表示させるための閾値を設定、または返す
SearchReducedImage	検索時に縮小画像を使用するか否かを設定、または返す
SearchPolarity	検索の極性を設定、または返す
SearchReversed	オブジェクト検出時に極性が反転しているワークを検出するか否かを設定、または返す
SearchWin	実行時のみ  1コールでサーチウィンドウの左端, 上端, 高さ, 幅、または、中心のX座標, 中心のY座標, 円内周の半径サイズ, 円外周の半径サイズのパラメーターを設定、または返す
SearchWinAngle	被検索領域の角度を定義
SearchWinAngleEnd	被検索領域の終了角度を定義
SearchWinAngleStart	被検索領域の開始角度を定義
SearchWinCenterX	被検索領域の中心のX座標値を定義
SearchWinCenterY	被検索領域の中心のY座標値を定義
SearchWinHeight	被検索領域の高さを定義 (単位: ピクセル)
SearchWinLeft	被検索領域の最左端を定義 (単位: ピクセル)
SearchWinPolygonPointX1	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第1頂点のX座標値を定義
SearchWinPolygonPointY1	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第1頂点のY座標値を定義
SearchWinPolygonPointX2	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第2頂点のX座標値を定義
SearchWinPolygonPointY2	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第2頂点のY座標値を定義
SearchWinPolygonPointX3	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第3頂点のX座標値を定義
SearchWinPolygonPointY3	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第3頂点のY座標値を定義
SearchWinPolygonPointX4	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第4頂点のX座標値を定義

プロパティ	説明
SearchWinPolygonPointY4	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第4頂点のY座標値を定義
SearchWinPolygonPointX5	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第5頂点のX座標値を定義
SearchWinPolygonPointY5	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第5頂点のY座標値を定義
SearchWinPolygonPointX6	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第6頂点のX座標値を定義
SearchWinPolygonPointY6	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第6頂点のY座標値を定義
SearchWinPolygonPointX7	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第7頂点のX座標値を定義
SearchWinPolygonPointY7	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第7頂点のY座標値を定義
SearchWinPolygonPointX8	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第8頂点のX座標値を定義
SearchWinPolygonPointY8	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第8頂点のY座標値を定義
SearchWinPolygonPointX9	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第9頂点のX座標値を定義
SearchWinPolygonPointY9	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第9頂点のY座標値を定義
SearchWinPolygonPointX10	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第10頂点のX座標値を定義
SearchWinPolygonPointY10	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第10頂点のY座標値を定義
SearchWinPolygonPointX11	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第11頂点のX座標値を定義
SearchWinPolygonPointY11	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第11頂点のY座標値を定義
SearchWinPolygonPointX12	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第12頂点のX座標値を定義
SearchWinPolygonPointY12	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第12頂点のY座標値を定義
SearchWinRadiusInner	被検索領域の円内径を定義
SearchWinRadiusOuter	被検索領域の円外径を定義
SearchWinTop	被検索領域の最上端を定義 (単位: ピクセル)
SearchWinType	被検査領域のタイプ (Rectangle, RotatedRectangle, Circle, Arc, Polygon)を定義
SearchWinWidth	被検索領域の幅を定義 (単位: ピクセル)

プロパティ	説明
SeparationAngle	検出されたオブジェクト間の許容最小角度を設定、または返す
SeparationMinX	検出されたオブジェクト間のX軸方向の許容最短距離を設定、または返す
SeparationMinY	検出されたオブジェクト間のY軸方向の許容最短距離を設定、または返す
SeparationScale	検出されたオブジェクト間の許容最小スケール差を設定、または返す
SharedEdges	検出されたオブジェクト間のエッジ共有を許容するかどうかを設定、または返す
ShowModel	教示モデルをさまざまなズーム設定によって表示 モデルの原点変更や検出マスクを設定
SkewFitEnable	モデルに対して歪み変形を適応するかを設定 デフォルト: False
Smoothness	幾何学的形状エッジ抽出フィルターの平滑化レベルを設定、または返す
Sort	オブジェクトの結果を並び替える順番を選択
Timeout	Geometricオブジェクトの最長サーチ時間を設定、または返す

## Geometricオブジェクトのリザルト

下の一覧は、Geometricオブジェクトのリザルトを簡単に説明した概略です。各リザルトの詳細は、Vision Guide 7.0プロパティ&リザルトリファレンスを参照してください。

リザルト	説明
Angle	検出されたパーツに関係する回転量を返す (最初に教示された方向に対して、パーツが持っている回転量を定義)
CameraX	カメラ座標系における検出パーツ位置の (モデル原点を基準にした)X座標を返す
CameraY	カメラ座標系における検出パーツ位置の (モデル原点を基準にした)Y座標を返す
CameraXYU	実行時のみ カメラ座標系における検出パーツ位置のCameraX座標, CameraY座標, CameraU座標を返す
ClearanceOK	クリアランス (隙間)の判断結果を返す
Found	オブジェクトが検出されたかどうか (対象物、またはパーツがAcceptプロパティの現在の設定を超える形状スコア値を持っているかどうか)の値を返す
FoundOnEdge	Geometricオブジェクトがサーチウィンドウのエッジに密接して検出されたときに、Trueを返す FoundOnEdgeがTrueのとき、FoundリザルトはFalseに設定されます。
NumberFound	検出されたオブジェクト数を返す (最低0個から、最高はNumberToFindプロパティで設定した検出個数になります。)
Overlapped	検出されたオブジェクト同士に重なりがある場合にTrueを返す
Passed	検出結果が良か否かを返す
PixelX	検出パーツ位置の (モデル原点を基準にした)X座標をピクセルで返す
PixelY	検出パーツ位置の (モデル原点を基準にした)Y座標をピクセルで返す
PixelXYU	実行時のみ 検出パーツ位置のPixelX座標, PixelY座標, PixelU座標をピクセルで返す
Reversed	検出されたオブジェクトがモデルと極性反転している場合にTrueを返す
RobotX	ロボット座標系における検出パーツの (モデル原点を基準にした)X座標位置を返す
RobotY	ロボット座標系における検出パーツの (モデル原点を基準にした)Y座標位置を返す
RobotU	ロボット座標系における検出パーツのU座標位置を返す

リザルト	説明
RobotXYU	実行時のみ ロボット座標系における、検出パーツ位置のRobotX座標, RobotY座標, RobotU座標を返す
Scale	倍率を返す このリザルトを有効にするには、ScaleEnabledを“True”に設定してください。
Score	実行時に検出された対象物が、Geometricオブジェクトが検索するモデルに一致するレベルを表わすINTEGER数値を返す
ShowAllResults	特定のビジョンオブジェクトのリザルトすべてを、表形式でダイアログに表示 リザルトの比較が簡単にできます。
SkewDirection	実行時に検出された対象物の歪み方向を返す
SkewRatio	実行時に検出された対象物の歪み率を返す
Time	オブジェクト処理に要した時間を返す (単位: ミリ秒)
TimedOut	オブジェクト実行がタイムアウトで終了したか否かを返す

### Geometricサーチの基本的知識

Geometricオブジェクトの目的は、サーチウィンドウで、事前に登録されたモデルを位置決めすることです。Geometricオブジェクトは、パーツの検出、対象物の有無の検出、パーツの欠陥の検出など、さまざまなものの検出に利用することができます。

ここでは、Geometricオブジェクトの基本的知識について説明します。説明する項目は次のとおりです。

- Geometricオブジェクトのモデルと対象物
- 検索の基本的概念
- AcceptプロパティとConfusionプロパティの設定
- 追加された検索パラメーター
- 複数リザルトのダイアログを使って、検索上の問題をデバッグする方法
- Geometricオブジェクトと回転
- Geometricオブジェクトとスケール
- 角度検索のためのモデル登録
- 検索の反復性と精度
- カメラと対象物間距離のキャリブレーション

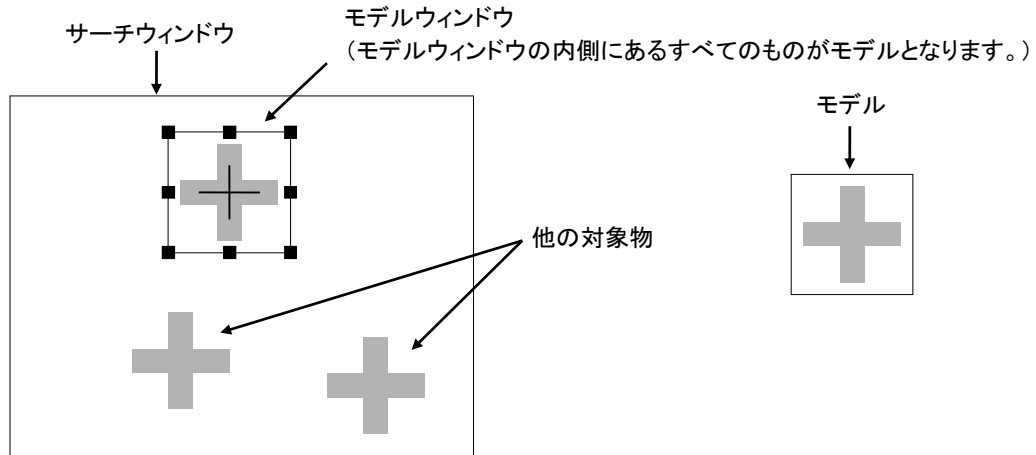
### Geometricオブジェクトのモデルと対象物

Geometricオブジェクトを使うときは、対象物とモデルの違いをよくご理解いただくことが重要になります。対象物とは、サーチウィンドウの領域中のエッジの特定パターンのことです。対象物には、数個のピクセルの単純なエッジから、数万個のピクセルの複雑なパターンまであります。Geometricオペレーションでは、領域中の対象物が、事前にその対象物を登録されたモデルに、どの程度一致するかを測定します。対象物の理想的な表現であるモデルあるいはテンプレートに対して、対象物とは、実際のサーチウィンドウ内に現実存在するもののことです。

## 6. ビジョンオブジェクト

一般的な検索では、1つのサーチウィンドウで登録された代表モデルを使って、そのサーチウィンドウ内にある代表モデルの類似対象物を検索します。下図は、対象物(十字線)を含むサーチウィンドウを示します。十字線のモデルを登録するには、モデルウィンドウを定義し、実行パネルの<ティーチ>ボタンをクリックしてください。(モデルの登録については、「Geometricオブジェクトの使用方法」を参照してください。)

その結果、モデルは下図の右側の図に示すとおり作成され、サーチウィンドウ内の他の十字線を検索するのに使用することができます。



対象となる数個の対象物を含むサーチウィンドウ(左側)と画像イメージから登録されたモデル(右側)

幾何学的形状対象物に基づいてモデルを検索する方法は、安定した信頼できる技法ですが、最適のモデルを選べるようにモデルの位置を決めるときに、気をつけなければならない点がいくつかあります。

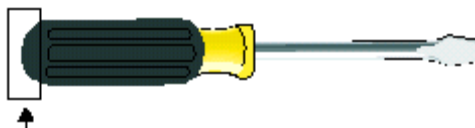
### 画像イメージに十分なコントラストをつけてください

コントラストは、サブピクセル精度で、モデル元のエッジとターゲット画像イメージを識別するのに必要不可欠です。幾何学的形状のエッジを基準にした検索アルゴリズムをGeometricサーチツールが使うので、コントラストの弱い画像イメージは使わないでください。コントラストが弱ければ弱いほど、検索に使うエッジを基準にした情報の量は少なく、精度は低くなります。このため、グレイスケール値のグラデーションが緩やかなモデルを使用しないでください。背景と画像イメージのエッジの間に、少なくとも10グレイスケール値の差を維持するようにしてください。

### 不適切な幾何学的形状モデルを使用しないでください

不適切な幾何学的形状モデルには、明確に定義された幾何学的形状対象物がない、または、他の画像イメージ対象物とはっきり区別できない幾何学的形状対象物であるという欠点があります。このモデルは不確かなリザルトを出すことがあります。

#### 不適切な幾何学的形状モデル



単純な曲線は特徴となる対象物がなく、誤って一致したとすることがあります。

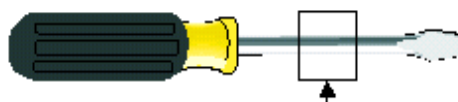


曖昧なモデルを使用しないでください

幾何学的形状モデルのあるタイプのものは、位置、角度、スケールについて、曖昧なリザルトを出します。

位置が曖昧なモデルは、通常、1組以上の平行線だけで構成されています。例えば、平行線だけでできているモデルは、平行線の正確な位置を確立することが不可能なので、使用しないでください。どの直線にも線分の実際の数値は、理論上無限にあるので、一致が無限に検出されます。線分は必ず、他の画像イメージの細部と区別できる特徴になる形状を含むようにしてください。

#### 位置の曖昧なモデル

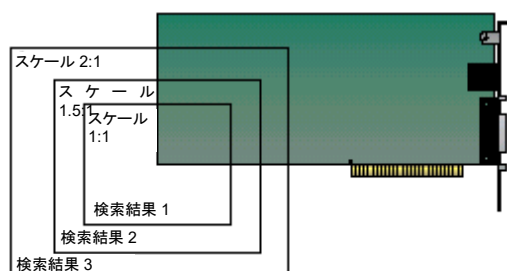


特徴となる対象物がない、平行線で構成されたモデルは使用しないでください。

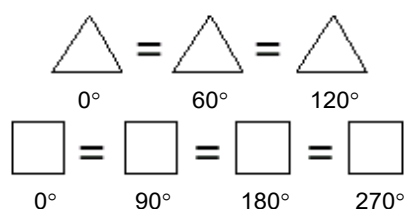
オブジェクトの一部分で構成されるモデルは、スケールが曖昧でないことを確認するために、テストを行ってください。例えば、孤立した角でできたモデルは、スケールの点で、曖昧です。



これは、スケールの範囲の中で検索するのに曖昧になる例です。区別の目安となる幾何学的特徴がないので、不適切な結果を出します。



対象物が類似しているために、対称モデルはよく角度が曖昧になります。例えば、円形は、角度の点において、完全に曖昧になります。四角形や三角形のような単純な対称モデルは、一定の角度が曖昧になります。



### ほとんど曖昧なモデル

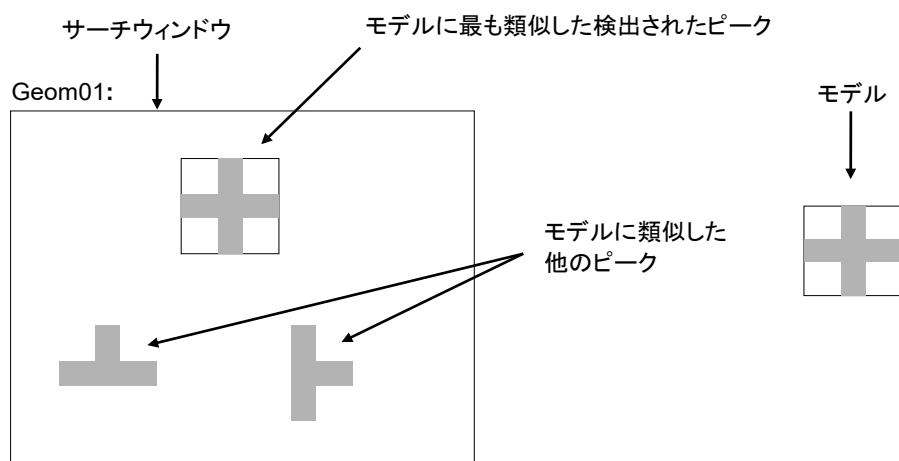
モデルの大部分に曖昧な対象物がある場合は、曖昧な対象物に含まれる検索結果のエッジの割合が一致とみなされるほど高いために、誤って一致したとすることがあります。これを避けるには、他のターゲット画像イメージの対象物と十分に区別できるような明瞭な対象物をモデルが持つことを確認してください。これにより、確実に正しい一致だけが検索結果として返されます。例えば、下記のモデルは、特徴となる曲線よりも、平行な直線でモデルの有効なエッジの大部分が構成されているために、誤って一致したとすることがあります。

### ほとんど曖昧なモデル



### 検索の基本的概念

検索では、サーチウィンドウ内でモデルが最も類似している領域を検出することによって、対象物を位置づけます。下図は、モデルとサーチウィンドウ、およびサーチウィンドウ内でモデルに最も類似している領域を示します。このようなモデルは、例えばプリント基板上の基準マークのような対象物を検索するのに使えます。ロボットは、サーチファンクションから戻された位置データを使って、ボード上の部品取付位置、あるいは、ボード自体の位置決め場所を検出できます。



### Acceptプロパティーのしきい値とConfusionプロパティーのしきい値の設定方法

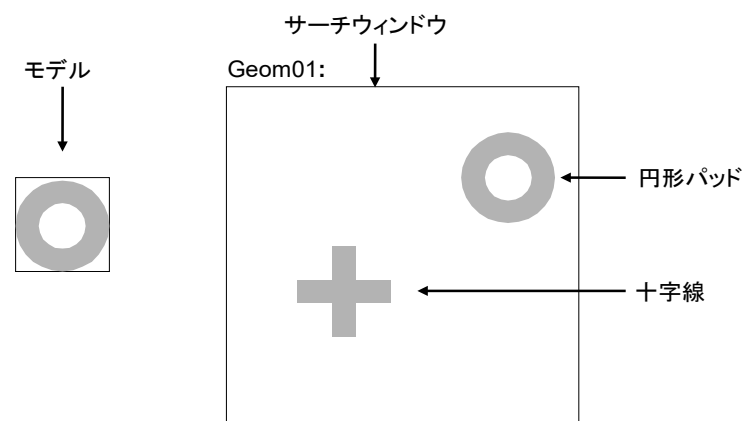
AcceptプロパティーとConfusionプロパティーは、Geometricオブジェクトの重要な検索パラメーターです。Acceptプロパティーは、どのような場合にシーンの一定領域での検索を続行するかヒントを与えることによって、検索速度に影響を与えます。Acceptプロパティーが高い値に設定された場合は、対象物はモデルに非常に類似したものとなるので、多くの領域が大まかな確認によって排除され、それ以上の検索が続行されるようなことはありません。Acceptプロパティーが低い値に設定された場合は、モデルにわずかでも類似している対象物はAcceptプロパティーのしきい値を超えてしまうので、シーン中のより多くの領域について詳細な確認が必要となります。このように、Acceptプロパティーの値が高ければ、Geometricオブジェクトの実行速度、つまり検索速度は速くなります。

**Confusion**プロパティは、検索速度に影響すると予想されるリザルトの数と相互作用します。**Confusion**プロパティと予想されるリザルトの数は一緒になって、画像イメージの検索可能領域の全検索が完了する前に、検索を中止させることができます。

**Accept**プロパティを設定して、許容可能な「最悪ケース劣化」の対象物例をシステムによって検索できるようにしてください。劣化は、欠陥、スケール、回転、ビデオノイズなどによって発生することがあります。**Vision Guide 7.0**では、**Accept**プロパティに“700”のデフォルト値を設定しています。この値は、ほとんどのアプリケーションにとって適切な開始点です。ただし、試行や訂正を重ねることによって、状況に合わせて最適な値に設定することもできます。なお、アプリケーションが適切に機能するためにスコアが“1000”、あるいはそれに近い値である必要はありませんので注意してください。対象物のゆがみなどにもよりますが、アプリケーションによっては形状スコア“200”でも良好な位置情報が得られる場合もあります。

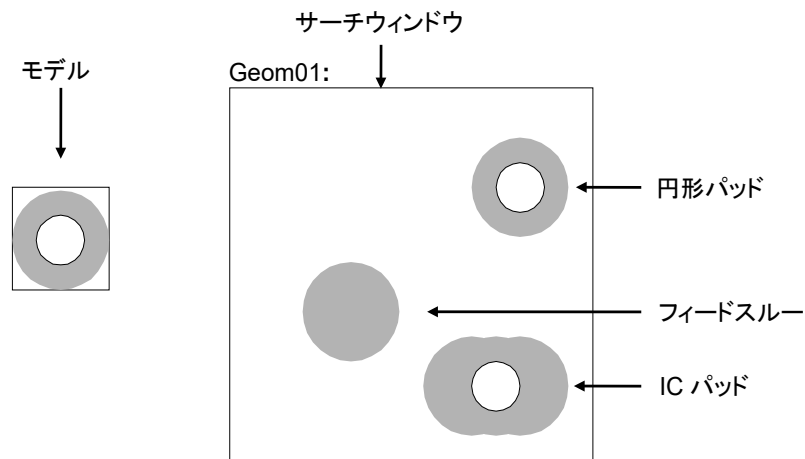
「誤差対象物」を獲得すると予想される最高値(+誤差余裕)を基にして、**Confusion**プロパティを設定してください。**Confusion**プロパティのしきい値は、**Accept**プロパティのしきい値に等しいかそれ以上の値にします。高い値に設定すると、検索時間が長くなりますが、正しい対象物を確実に検出するためにはやむをえません。**Confusion**プロパティのデフォルト値は“800”ですが、アプリケーション要件にしたがって調整してください。

下図は、複雑度のほとんどないシーンを示しています。円形のパッドは、十字線とほとんど類似性はありません。したがって、**Confusion**プロパティは、かなり低い値(700程度)に設定することができます。許容可能な劣化量によりますが、**Accept**プロパティは、通常、**Confusion**プロパティに等しいかそれ以下の値に設定します。このシーンでは劣化がほとんどないので、“920”の形状スコアが適切と予想されます。



複雑度がほとんどないシーン

下図は、複雑度の多いシーンを示しています。フィードスルーとICパッドは、両方とも、円形パッドに類似しています。したがって、Confusionプロパティは、かなり高い値 (820程度) に設定しなければなりません。



複雑度の度合いが高いシーン

グレイ定数の領域を持つサーチウィンドウは、その領域で常に0の幾何学的形状値を獲得します。シーンが基本的に均一の背景を持つ場合 (例: 1枚の白い紙) は、ほとんどの場所で幾何学的形状がないので、Geometricオブジェクトが何かを検出した場合、それは検出したい対象物であるはずであり、Confusionプロパティは低い値に設定することができます。

AcceptプロパティとConfusionプロパティの設定の仕方によって、より迅速に対象物の位置を把握させることができる、と考えることもできます。一般的に、これらのプロパティには慎重な値を設定すべきであって、正確に設定する必要はありません。最も慎重な設定方法はAcceptプロパティを低く、Confusionプロパティを高く設定する方法です。検索するシーンについてわかっている情報が少ししかない場合は、より慎重な設定方法をとってください。検索は注意深く実行されますが、検索速度は遅くなります。(これは、Geometricプロパティ位置リザルトを使ってロボットに動作先を示す場合、特に重要となります。)

検索するシーンについてわかっている情報が多い場合は、より大まかな設定方法を実施してください。例えば、ある1つの対象物を探していて、残りのシーンに何もない場合は、より大まかな値を設定してください。注意深く検索する必要がないので、検索速度は速くなります。

## 追加されたGeometricサーチパラメーター

DetailLevel	<p>DetailLevelは、検索中にエッジとみなすものを決めます。</p> <p>エッジは隣接するピクセルとの間のグレイスケール値の変化により定義されます。デフォルト設定の“Medium”では、コントラストに変化がある、ノイズがある、照明が一定でないなどの状況で、画像イメージから有効なエッジを安定して検出します。画像イメージ中のコントラストの強い領域に比べて、目的のオブジェクトがとても弱いコントラストの場合などは、コントラストの弱いエッジが検出されないことがあります。画像イメージ中にコントラストの弱いオブジェクトがある場合、DetailLevelの設定を高くして、画像イメージ中の重要なエッジはすべて検出できるようにしてください。VeryHighの設定では、コントラストが弱いエッジもすべて抽出します。ただし、このモードはノイズにとっても影響されやすいことに注意してください。</p>
ScaleTargetPriority	<p>オブジェクト検出時に、ScaleTarget近傍を優先的に検出するように設定します。工業製品など検出オブジェクトにサイズ変動がない場合、本プロパティを“True”に設定してください。</p> <p>デフォルト: True</p>
SearchReducedImage	<p>オブジェクト検出の粗検出時に、入力画像を縮小して検出するようになります。入力画像に特徴点(エッジなど)が非常に多い場合など、検索時間を短縮できることがあります。</p> <p>デフォルト: False</p>
Smoothness	<p>Smoothnessプロパティにより、エッジ抽出フィルターの平滑化レベルをコントロールすることができます。平滑化オペレーションは、粗いエッジを滑らかにして、ノイズを除去します。このコントロール範囲は、0 (滑らかでない) ~ 100 (非常に滑らか)までです。</p> <p>デフォルト: 50</p>
SharedEdges	<p>SharedEdgesを“True”に設定して、複数リザルトにエッジを共有させることができます。これを選択しない場合は、複数のリザルトの一部になるエッジは最大スコアを持つリザルトの一部だとみなします。</p>
TimeOut	<p>時間重視のアプリケーションでは、Geometricオブジェクトが指定モデルの検索結果を検出するのに時間制限をミリ秒単位で設定できます。時間制限中に必要とされる数の検索結果が検出されない場合は、検索は停止します。検出された検索結果のリザルトは返されます。ただし、制限時間になる前に、どの検索結果が検出されるのかはわかりません。</p> <p>デフォルト: 2000ミリ秒</p>

### 複数リザルトのダイアログを使って、検索上の問題をデバッグする方法

たとえ同一生産ロット内でも、ときには、作業しているパーツがかなり変化したり、1つのパーツ上に類似対象物が2つ以上あったりする場合があります。このような場合、Acceptプロパティを適切な値に設定した場合でも、その値に該当する他のパーツをまちがえて検索してしまう場合があります。これらのケースでは、何が起こるか予想するのが非常に困難です。

[Show All Results]ダイアログは、上記の問題をはじめとするさまざまな問題を解決するために作成されました。1つのパーツ上に1つの対象物だけを検出したい場合は、複数検出を行って、Vision Guide 7.0が時々二次的な対象物を本来検出したい対象物として結果を返す理由を調べることができます。このような問題が発生するには、一般的に、いくつか異なった原因があります。

1. サーチウィンドウ中の2つ以上の対象物が非常に類似しているため、非常に近い値のScoreリザルトを持つ場合
2. Confusionプロパティ、またはAcceptプロパティが十分高く設定されていないために、本来検出したい対象物よりも低いスコアを持つ対象物がAcceptプロパティの設定値を満たしてしまう場合

Vision Guide 7.0を初めてご使用になる方にとって、上記の2つの状況は、サーチウィンドウ内で1つの対象物を検索する場合に極めて混同しやすいので注意してください。

本来検出したい対象物が検出されたり、別の対象物が検出されたりするようなときには、リザルトダイアログを使って、問題を特定してください。問題の原因を参照するための手順を、次に示します。

- (1) NumberToFindプロパティを“3”以上に設定してください。
- (2) Vision Guide 7.0開発環境からビジョンオブジェクトを実行してください。
- (3) <ShowAllResults>プロパティボタンをクリックして、リザルトダイアログを表示してください。
- (4) 検出された上位3つあるいはそれ以上の対象物のスコアを確認してください。

このように、検出された上位3つあるいはそれ以上の対象物のスコアを確認すると、問題の原因が明確になります。このような問題は、ほとんど次に示すどちらかの場合に発生します。

1. 検出された各々の対象物が、Acceptプロパティの設定値より大きいスコアを持っている場合。  
このような場合には、Confusionプロパティをより高く設定し、適切な対象物だけを常に強制的に検出するようにして、Acceptプロパティのしきい値に一致するようなその他の対象物は返されないようにしてください。また、Acceptプロパティの値を変更することもできます。
2. 各々の対象物のスコアが非常に近接している場合。  
このような場合は、次の処理を行って、本来検出したい対象物が区別できるようにしてください。
  - サーチウィンドウを再調整し、検出対象物としてランダムに返される対象物がサーチウィンドウ内部に入らないようにします。
  - 本来検出したい対象物をモデルに再登録します。

- アプリケーションの照明条件を調整して、本来検出したい対象物が、まちがって検出してしまふ対象物のスコアより高いスコアを獲得するようにしてください。

複数リザルトの使用方法的詳細は、「6.2.24 1つのオブジェクトから複数のリザルトを返す」の項目を参照してください。

### Geometricオブジェクトと分離

検索結果がはっきり区別できる (一致)とみなされるのに必要な、同一モデルの他の検索結果との分離の最小量を指定できます。本質的には、これは同一モデルの検索結果がオーバーラップできる量を定義します。

X位置, Y位置, 角度, スケールの4基準の最小分離を設定できます。検索結果が他と区別できるとみなされるためには、最小分離条件の1つを満たすことが必要です。例えば、角度の最小分離を満たせば、位置やスケールに関係なく、検索結果は区別できるとみなされます。しかし、これら分離基準は、有効な検索結果を決定するときに考慮されないように、無効になることがあります。

位置の最小分離プロパティのSeparationMinXとSeparationMinYは、同一モデルの2つの検索結果の検出位置がどれくらい離れていなければならないかを決定します。この分離は、公称スケール (ScaleTarget)でのモデルサイズの割合として指定されます。

デフォルト値は10%です。0%の値はプロパティを無効にします。例えば、ScaleTargetで、モデルが幅100ピクセルの場合、SeparationMinXを10%に設定すると、検索結果が区別できて、分離しているとみなされるためには、検索結果は最低10ピクセルX方向に離れていなければなりません。

角度の最小分離 (SeparationAngle)は、検索結果間の角度の最小差異を決定します。この値は絶対角度値として指定されます。デフォルト値は10.0°です。0°の値はプロパティを無効にします。

スケールの最小分離 (SeparationScale)は、検索結果間のスケールの最小差異を倍率として定義します。デフォルト値は1.1です。1.0の値はプロパティを無効にします。

### Geometricオブジェクトとスケール

モデルのスケールは、ターゲット画像イメージ中で検出されると予想されるモデルのサイズを規定します。予想される検索結果がモデルよりも小さい、あるいは大きい場合は、サポートされた倍率に応じて、スケールを設定できます。予想スケールはScaleTargetプロパティ (範囲: 0.5 ~ 2.0)を使って設定されます。

デフォルトで、スケール範囲の検索は無効になっています。必要なら、ScaleEnableをTrueに設定して、スケール範囲の検索を有効にすることもできます。これにより、指定したScaleTargetから、小さいサイズでも大きいサイズでも、異なるサイズ範囲のターゲット画像イメージ中でモデルを検出することができます。スケール範囲を設定するには、ScaleMinFactorとScaleMaxFactorを使ってください。ScaleMinFactor (0.5 ~ 1.0)とScaleMaxFactor (1.0 ~ 2.0)は、ともに、公称ScaleTargetからスケール範囲を決定します。

これらの最大倍率と最小倍率は、次のようにScaleTarget設定に用いられます。

$$\text{max scale} = \text{ScaleTarget} \times \text{ScaleMaxFactor}$$
$$\text{min scale} = \text{ScaleTarget} \times \text{ScaleMinFactor}$$

予想スケール(ScaleTarget)を変更する場合に、範囲を変更する必要がないように、範囲を倍率として定義するように注意してください。スケール範囲の検索は、平行に行われます。つまり、検索結果の実際のスケールは、最初に検出される検索結果には影響しません。しかし、スケール範囲が大きくなればなるほど、検索は遅くなります。

### Geometricオブジェクトと回転

Geometricオブジェクトは回転したパーツを検出するのに理想的です。エッジの幾何学的形状パターンを検索しているので、回転したパーツは、一般的に他のビジョンツールよりも、より確かに検出されます。

Geometricオブジェクトの角度検出付きサーチを行う場合は、Geometricオブジェクトのモデルを登録するときに、AngleEnableプロパティの設定を“True”としておかなければなりません。AngleStartプロパティとAngleRangeプロパティも設定してください。

### 角度検出のためのモデル登録

角度検出付きサーチを行うには、まず、角度検出のためのGeometricサーチオブジェクトを設定してください。そのための手順としては、AngleEnableプロパティを“True”に設定し、登録されるモデルの角度範囲をAngleRangeプロパティを使って指定します。AngleStartプロパティを設定することにより、角度検出の中心を変更することもできます。これはAngleRangeが基づく角度です。例えば、AngleStartが45、AngleRangeが10の場合、35°～55°までのモデルを検索します。

角度付でモデルを登録するときは、モデルがサーチウィンドウの外側にはみ出すことなく回転できるように、サーチウィンドウは充分大きくなければなりません。

### 検索の反復性と精度

検索の反復性と精度とは、モデルのサイズと詳細 (形状、対象物の粗さ、形状の対称性) と、サーチウィンドウの中の対象物の劣化(ノイズ、欠陥、回転やスケールによる影響)により影響を受けます。

ノイズによる位置への影響を測定するには、劣化のない対象物を含む特定のサーチウィンドウで検索を実行し、この検索と完全に同じ検索をオブジェクトの位置を変更せずに再実行し(2番目の画像イメージをフレームバッファに取り込み)、位置についての測定結果を比較します。そのための手順を次に示します。

- (1) モデルの定義後、実行パネルのオブジェクトの<Run>ボタンを2回以上クリックします。
- (2) <統計>ツールバーボタンをクリックします。
- (3) すると、統計ダイアログが表示され、2回のオブジェクト検索で得られた位置の差異を調べることができます。

劣化のない対象物の大型モデル (30 × 30)については、1/20ピクセルの繰返し精度を得ることも可能です。しかし、ほとんどの場合、1ピクセルをわずかに下回る精度 (1/2, 1/3, 1/4 など)が現実的です。



検索精度は、劣化のない対象物を含む特定のサーチウィンドウで検索を実行し、オブジェクトを移動して、検索で得られた位置の差異と実際の差異を比較することで計測できます。劣化がなく、回転誤差やスケール誤差がなく、X方向、Y方向ともに十分なエッジがある大型モデル (30 × 30) については、検索精度は、1/4ピクセルになります。(これは、ビジョンシステムでの検索精度であって、ロボットに固有の不正確な精度は影響していませんので、注意してください。したがって、ロボットでパーツを移動するときは、ロボットメカニズムそのものの不正確な精度も考慮に入れなければなりません。)

### カメラと対象物間距離のキャリブレーション

最適な検索結果を得るためには、画像イメージ中の対象物の検索時のサイズが、モデルが登録されたときのサイズと同じでなければなりません。同じカメラと同じレンズを使用すると仮定した場合、モデルを登録した時と検索を実行する時とで、対象物とカメラとの距離が変わると、サーチウィンドウ中では対象物の見かけ上のサイズが異なって見えます。つまり、カメラが対象物に近づくと対象物は大きくなり、カメラが対象物から遠ざかると対象物は小さくなります。


対象物とカメラとの距離が変わるときは、モデルを再登録しなければなりません。

### Geometricオブジェクトの使用方法

以上で正規化Geometricと検索処理がどのように動作するのか確認し、Vision Guide 7.0 のGeometricオブジェクトの使用方法を理解するための基本的知識を得ることができました。

ここでは、Geometricオブジェクトの使用方法を、次にリストアップしたステップごとに説明します。


- 新規Geometricオブジェクトの作成方法
- サーチウィンドウの位置とサイズの設定
- モデルウィンドウの位置とサイズの設定
- モデル原点の位置決め
- Geometricオブジェクトの関連プロパティの構成
- モデルの登録
- Geometricオブジェクトのテストとリザルトの確認
- プロパティの調整と再テスト
- 単一Geometricオブジェクトからの複数検出を使用した作業

これらのステップを開始する前に、まず新規ビジョンシーケンスを作成するか、使用するビジョンシーケンスを選択してください。作業するビジョンシーケンスがない場合は、 <新規シーケンス>ボタンをクリックすると、新規ビジョンシーケンスが作成できます。

すでにビジョンシーケンスを作成済みの場合は、Vision Guideウィンドウのシーケンスツリーをクリックして、作成済みのビジョンシーケンスを選択してください。

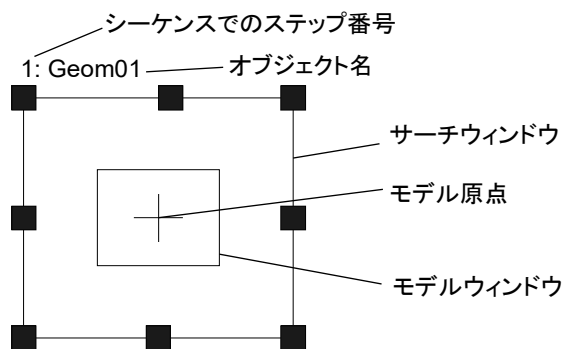
新規ビジョンシーケンスの作成方法、または作成済みビジョンシーケンスの選択方法の詳細は、「ビジョンシーケンス」を参照してください。

### Step 1: 新規Geometricオブジェクトの作成

- (1) Vision Guideツールバー - <全ツール> -  <Geometric (幾何学形状)>ボタンをクリックしてください。
- (2) <Geometric(幾何学的形状)>オブジェクトツールバーボタンの上に、Geometricアイコンが表示されます。
- (3) Geometricアイコンをクリックして、Vision Guideウィンドウの画像イメージ表示部までドラッグしてください。
- (4) オブジェクト名は自動的に作成されますので注意してください。この例では、このシーケンスで最初に作成されたGeometricオブジェクトなので、“Geom01”というオブジェクト名になります。(オブジェクト名の変更方法については後述します。)

### Step 2: サーチウィンドウの位置とサイズの設定

画面上には、下図のようなGeometricオブジェクトが表示されます。



新規Geometricオブジェクトのレイアウト

- (1) Geometric オブジェクトの名称ラベルをクリックし、ボタンを押し続けながら、Geometricオブジェクトをドラッグして、サーチウィンドウの左上コーナーを好きな位置に合わせてください。
- (2) サイズハンドルをクリックして、マウスをドラッグすると、Geometricオブジェクトサーチウィンドウのサイズが変更できますので、適当な大きさに調整してください。(サーチウィンドウ内が、被検索領域になります。)



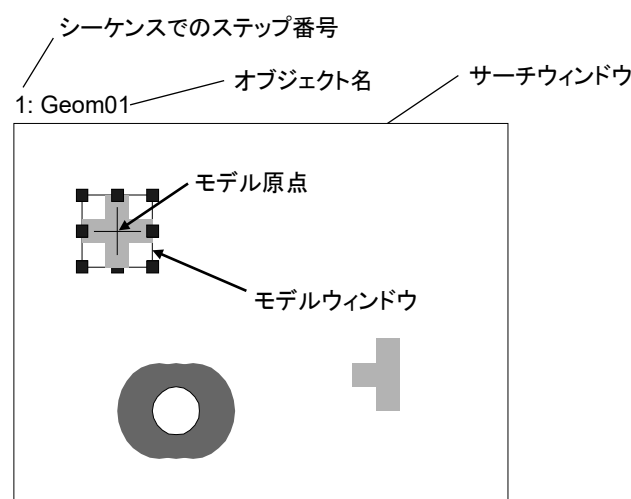
注意

- ビジョンシーケンスの実行結果は、外らん光や外部機器ノイズにより影響を受けることがあります。  
外らん光や外部機器ノイズにより、影響を受けると、取り込み画像が想定外の結果となり、検出位置結果がサーチエリア (検出エリア)内の不特定位置になる可能性があります。  
サーチエリアは、極力小さく、また誤検出に配慮した画像処理シーケンスの作成を行ってください。

### Step 3: モデルウィンドウの位置とサイズの設定

- (1) 作業を行うGeometricオブジェクトのサーチウィンドウは、サイズハンドルが強調表示されています。サイズハンドルが見えない場合は、Geometricオブジェクト名フィールドをクリックしてください。サイズハンドルが見えるなら、この手順は無視して、次の手順へ進んでください。
- (2) モデルウィンドウを形成するボックスの1辺をクリックしてください。この操作によって、モデルウィンドウが強調表示され、モデルウィンドウのサイズハンドルが現われます。
- (3) モデルウィンドウを形成するボックスの1辺をクリックして、マウスを押し続けながらモデルウィンドウをドラッグし、モデルウィンドウの左上コーナーの位置を合わせてください。
- (4) 次に、サイズハンドルをクリックしてマウスをドラッグし、モデルウィンドウのサイズを調整して、Geometricオブジェクトのモデルとして登録する対象物を囲んでください。

Geometricオブジェクトのレイアウトは、下図のように表され、サーチウィンドウは被検索領域をカバーし、モデルウィンドウは検索したい対象物を包囲しているはずです。実際のサーチウィンドウとモデルウィンドウは、この例のように表されていないかもしれませんが、画面予想図として参考にしてください。



サーチウィンドウとモデルウィンドウの位置と、サイズ調整後のGeometricオブジェクト

### モデルウィンドウの位置とサイズを適切に設定するための注意点:

Geometricオブジェクトのモデルウィンドウの位置とサイズは、これによって検索する対象物が定義されるので、適切な調整を行うことが重要です。モデルウィンドウを作成する場合には、特に次の2点に留意してください。

- (1) 被検索領域であるサーチウィンドウは、なるべく小さい方が検索時間を短縮できます。また、特にパーツの回転が大きいと予想されるときには、サーチウィンドウを小さく設定し、モデルも小さく(パーツの一部などに)設定できれば、パーツの回転による影響が少なくて済みます。
- (2) モデルウィンドウサイズを可能な限りサーチウィンドウサイズに近づけることによって、検索実行時間を短縮することができます。

また、2つのパーツが互いに接触するくらいに隣接しているような場合は、モデルウィンドウを実際の対象物より少しだけ(数ピクセル分)大きめに設定しておく、対象物を他の物体と区別しやすくなります。

モデルウィンドウの最適なサイズは、それぞれのビジョンアプリケーションによって異なりますので注意してください。

---

### Step 4: モデル原点の位置決め

モデル原点は、オブジェクトを実行するときに、対象物の位置として返されるモデル上の位置を定義します。つまり、モデル原点は、位置データが重要となる場合に、有意性のある場所に位置づけられていなければなりません。

例えば、Geometric オブジェクトを使ってパーツを検出し、ロボットにパーツのピックアップや配置をさせる場合は、モデル原点の位置は、ロボットが簡単にパーツをつかむことのできる場所であることが重要となります。その位置は、RobotX, RobotY, RobotU, RobotXYU リザルトに基づくロボットの動作先になるからです。

新規Geometricオブジェクトが作成されると、ModelOrgAutoCenter プロパティはデフォルト値の“True”に設定されます。つまり、モデル原点は、モデルウィンドウの中心に自動的に位置づけられ、手動で移動させることができません。モデル原点を移動させたい場合には、最初にModelOrgAutoCenterプロパティの値を“False”に設定する必要があります。この設定手順と、モデル原点を実際に位置決めする手順を次に示します。

- (1) Vision GuideウィンドウのフローチャートからGeometricオブジェクトをクリックしてください。オブジェクトウィンドウのプロパティリストでModelOrgAutoCenterプロパティを探し、その設定値フィールドをクリックしてください。
- (2) ドロップダウンリストに、2つの選択項目、“True”と“False”が表示されますので、“False”をクリックしてください。これで、ModelOrgAutoCenterプロパティが“False”に設定され、モデル原点はマウスで移動することができます。
- (3) モデルウィンドウをクリックして、モデルウィンドウを強調表示させてください。
- (4) モデル原点をクリックして、マウスボタンを押し続けながら、モデル原点を新規の位置にドラッグしてください。なお、モデル原点を置くことができるのは、モデルウィンドウの境界内だけですので、注意してください。

## Step 5: Geometricオブジェクトプロパティの設定

Step 4で表示したオブジェクトウィンドウから、Geometricオブジェクトプロパティの設定を行います。プロパティを設定するには、関連するプロパティの設定値フィールドをクリックし、新規の値を入力してください。あるいは、ドロップダウンリストが表示されている場合は、リスト中の該当項目をクリックしてください。

次に、Geometricオブジェクトで一般的に使われるプロパティのいくつかを示します。その他、さまざまなビジョンオブジェクトの多くで使われる、AbortSeqOnFail, Graphicsなどのプロパティの詳細は、「Geometricオブジェクトのプロパティ」を参照してください。Geometric オブジェクトをテストするときは、これらのプロパティを設定する必要はありません。ただし、実際にGeometricオブジェクトで作業するのが初めての方は、この説明を参考にしてください。



注意

- ビジョンシーケンスの実行結果は、外らん光や外部機器ノイズにより影響を受けることがあります。  
外らん光や外部機器ノイズにより、影響を受けると、取り込み画像が想定外の結果となり、検出位置結果がサーチエリア (検出エリア)内の不特定位置になる可能性があります。  
Accept, RejectOnEdgeなどのプロパティ値を適切に設定し、誤検出リスクを軽減してください。

## Nameプロパティ

新規に作成されたGeometric オブジェクトに、デフォルトとして与えられるオブジェクト名は“Geomxx”です。xxは、1つのビジョンシーケンス内で複数のGeometricオブジェクトを用いるとき、それらのGeometricオブジェクトを区別するための番号です。

ビジョンシーケンスで1番目に用いられるGeometricオブジェクトには、“Geom01”というオブジェクト名がデフォルトで設定されます。

オブジェクト名を変更したい場合は、Nameプロパティの設定値フィールドをクリックして、変更後の名称を入力し、リターンキーを押してください。

画面上でGeometricオブジェクト名を表示する場所は、すべて変更後の名称に変わります。

## Acceptプロパティ

Acceptプロパティは、対象物が一致するべき、あるいは検出されたと判定するときの基準となる形状スコアの値を設定します。

Scoreリザルトに返される値は、このAcceptプロパティに設定されている値との比較が行われます。

デフォルト: 700

## Confusionプロパティ

サーチウィンドウの中に類似する対象物が複数個ある場合、このConfusionプロパティを使うと、検索したい対象物を正確に獲得するのに役立ちます。

デフォルト: 800

ModelOrgAutoCenter プロパティ	最初にこのModelOrgAutoCenterプロパティを “False” に設定しておくでモデル原点の位置が変更できます。  デフォルト: True
Frameプロパティ	このプロパティによって、事前に定義されていたFrameオブジェクトをGeometricオブジェクトのリファレンスフレームとして選択することができます。  Frameの詳細は、「Frameオブジェクト」で説明しています。
NumberToFind プロパティ	検出したい対象物の数に合わせて、NumberToFindプロパティを1より大きい数値に設定することができます。これによって、Geometricオブジェクトは1つのサーチウィンドウ内で複数の対象物を検出することができます。
AngleEnable プロパティ	Geometricモデルを角度付で検索したい場合は、このプロパティを “True” に設定しなければなりません。複数のモデルを角度付で検索できるようにするには、Geometric オブジェクトのモデル登録の前に、このプロパティを “True” に設定しておかなければなりません。
AngleRange プロパティ	これらのプロパティは、AngleEnableプロパティと一緒に用いることにより、Geometricモデルを角度付で検索することができます。
RejectOnEdge プロパティ	このプロパティによって、サーチウィンドウのエッジにかかって検出されたパーツを除外することができます。  通常はこのプロパティを “True” に設定します。

プロパティは、まずデフォルト設定のままにして次のステップに進み、後で必要に応じて設定を変更することができます。

### Step 6: Geometricオブジェクトのモデルの登録

Geometricオブジェクトには、検索するためのモデルが必要であり、モデルの登録と呼ばれるプロセスによって実行します。すでに、Geometricオブジェクトのモデルウィンドウは、モデルとして使いたい対象物をとり囲むように位置決めされているはずです。モデルの登録は、次に示す手順で完成します。

- (1) Geometricオブジェクトが、現在表示されているオブジェクトであることを確認してください。フローチャート、またはシーケンスツリーを見ると、どのオブジェクトが現在作業中のオブジェクトになっているか確認することができます。また、画像イメージ表示部で、どのオブジェクトがマゼンタ色で強調表示されているかによって確認することもできます。
- (2) 実行パネルにある、<ティーチ>ボタンをクリックしてください。モデルが登録されます。モデルの登録に要する時間は、ほとんどの場合わずか数秒です。ただし、AngleEnableプロパティが “True” に設定されているときにモデルを登録する場合は、わずかな角度ずつオフセットした複数のモデルを登録する必要があるため、モデルの登録には、かなりの時間を要します。

**Step 7: Geometricオブジェクトのテストとリザルトの確認**

Geometricオブジェクトを実行するには、実行パネルのオブジェクトの<Run>ボタンをクリックしてください。

Geometric オブジェクトが実行されると、リザルトが表示されますので確認してください。この時点で確認すべき主要なリザルトは、次のとおりです。

<b>Foundリザルト</b>	Geometricオブジェクトが検出されたかどうかの値を返す  検索している対象物が検出されると、リザルトとして “True”の値が返ります。対象物が検出されないと、Foundリザルトは “False”を返し、赤で強調表示されます。Geometricオブジェクトが検出されなかった場合、その一般的理由については、Step 8を参照してください。
<b>FoundOnEdgeリザルト</b>	対象物がサーチウィンドウのエッジに接して検出されたときに、“True”を返す  この場合、Foundリザルトは “False”を返します。
<b>Score リザルト</b>	モデルと、モデルに最もよく似ている対象物とが、どの程度一致しているかを示す  Scoreリザルトの範囲は、0~1000で、1000はモデルと対象物が完全に一致したことを示します。検出された対象物が、検索したい対象物に一致している程度を確認するために、まず、Geometricオブジェクト実行後のScoreリザルトを確認してください。
<b>Time リザルト</b>	Geometric オブジェクトの実行に要した時間を返す  サーチウィンドウとモデルウィンドウを小さくすることで、検索速度は速くなります。
<b>NumberFoundリザルト</b>	1つ以上のGeometricオブジェクトを実行するとき、NumberFoundリザルトは、Geometricオブジェクトのモデルに一致した対象物の数を返す
<b>Angleリザルト</b>	Geometricオブジェクトの傾斜角度を返す  この角度は、モデルに設定された初期角度に基づいて計算されますが、ときには概略的な値であり、信頼性が高くないことがあります。  回転角度の決定にはPolarオブジェクトを使うことをお勧めします。特にロボットガイダンスには、Polarオブジェクトを使うようにしてください。
<b>PixelXリザルト PixelYリザルト</b>	対象物のXY座標値 (ピクセル座標系)を返す  この値は、検出された対象物のモデル原点の位置です。別の位置を返したい場合は、モデル原点の位置を変更し、モデルを再登録する必要があります。
<b>CameraXリザルト CameraYリザルト</b>	カメラ座標系における検出対象物のXY座標値を定義  これらのリザルトは、カメラキャリブレーションが完了していない場合は[nocal]を返し、値を返しません。

RobotXリザルト	ロボット座標系における検出対象物のXY座標値を定義
RobotYリザルト	ロボットは、このXY座標値を動作先とすることができます。(他の変換、処理作業は不要です。)  この値は、検出された対象物のモデル原点位置です。別の位置を返したい場合は、まずモデル原点の位置を変更しモデルを再登録する必要があります。これらのリザルトは、カメラキャリブレーションが完了していない場合は“no cal”を返し、値を返しません。
RobotUリザルト	検出対象物の角度として、ロボット座標系に変換された値を返す  RobotUリザルトは、カメラキャリブレーションが完了していない場合は“no cal”を返し、値を返しません。
ShowAllResults	複数のリザルトで作業しているとき、ShowAllResultsの設定値フィールドにあるボタンをクリックすると、ShowAllResultsダイアログが表示され、現在作業中のビジョンオブジェクトのリザルトすべてを参照することができます。

### NOTE



上記のステップ例では、キャリブレーションを実行していませんので、RobotXYU, RobotX, RobotY, RobotU, CameraX, CameraY, CameraXYUのリザルトは、“no cal”を返します。これは、キャリブレーションを実行しなかったため、ロボット座標系あるいはカメラ座標系に関して座標リザルトを計算することができなかった、という意味です。詳細は、「キャリブレーション」を参照してください。

## Step 8: プロパティの調整と再テスト

Geometric オブジェクトを数回実行することで、検出上の問題が見つかったり、プロパティ設定の微調整をする必要が出てきたりするかもしれません。ここでは、一般的に生じる問題のトラブルシューティングや、微調整の方法について説明します。

### Geometricオブジェクトのトラブルシューティング

#### GeometricオブジェクトがFalseのFoundリザルトを返す場合

- Acceptプロパティの設定を少し低め(例えば、現在のScoreリザルトより小さい値)に変更し、Geometricオブジェクトを再度実行してみてください。
- 返されたFoundOnEdgeリザルトが、“True”かどうかを確認してください。Foundリザルトが“False”を返し、FoundOnEdgeリザルトが“True”を返した場合は、対象物は検出されたものの、その一部がサーチウィンドウに接触していることを意味しています。この場合は、サーチウィンドウを大きくし、対象物がサーチウィンドウに収まるようにしてください。サーチウィンドウのサイズ変更が不可能な場合は、カメラの位置を変更するか、モデルウィンドウのサイズを再調整してください。

#### Geometricオブジェクトが誤認対象物を検出した場合

- Acceptプロパティは、充分高い値に設定されているでしょうか。Acceptプロパティの設定値が低すぎると、検索対象となる対象物のかわりに、別の対象物が検出される可能性があります。
- Confusionプロパティは、充分高い値に設定されているでしょうか。Confusionプロパティの値は、Acceptプロパティの値に等しいか、それ以上の値でなければなりません。



また、サーチウィンドウ内に、検索する対象物に類似した対象物がある場合は、**Confusion**プロパティの設定をさらに高い値に変更し、別の対象物を誤って検出しないようにしなければなりません。

- サーチウィンドウを調整して、検索する対象物に近づけ、その対象物だけを隔離するようにしてください。

### Geometricオブジェクトの微調整

オブジェクトが正常に動作するためには、通常、**Geometric**オブジェクトの微調整が必要となります。**Geometric**オブジェクトの微調整に関する主要なプロパティ、およびモデル追加について、次に説明します。

**Acceptプロパティ** **Accept**プロパティを低くすると、**Geometric**オブジェクトの実行速度は速くなります。しかし、**Accept**プロパティの値が低いと、検出したい対象物とは別の対象物が検出される可能性も強くなります。**Geometric**オブジェクトを数回実行してみて、**Score**リザルトに返される形状スコアの値の予想がつけられるようになったら、**Accept**プロパティの値を調整してみてください。数回試行を重ね、より速い実行速度で信頼性の高い対象物を得ることができるような適切な値を設定してください。

**Confusionプロパティ** サーチウィンドウ中に類似対象物が複数ある場合は、**Confusion**プロパティに相対的に高い値を設定する必要があります。これによって、検出したい対象物を、他の混同しやすい対象物と区別して検出することができます。ただし、**Confusion**プロパティの値を高く設定すると、実行速度が遅くなります。サーチウィンドウ中に類似対象物がない場合は、**Confusion**プロパティに低い値を設定して実行速度を速くすることができます。

**サンプルの追加** 現在のモデルと同じサイズのモデルウィンドウでティーチすると、「サンプルの追加」を選択できます。形や模様が少し違う、影の出方が変わる、など、モデルが少し変化する場合、変化したモデルを選択してモデルを追加すると、オブジェクト実行時のスコアが安定的になります。角度が大きくなりすぎていたり、変化が大きすぎてモデルの追加に失敗したときは、もとのモデルが保持されます。

調整が完了し、**Geometric** オブジェクトのテストを行って、満足なリザルトを得ることができたら、ビジョンオブジェクトの作成ステップは完了します。次のステップとして、他のビジョンオブジェクトを作成し、ビジョンシーケンス全体の設定やテストに進むことができます。

### Geometricオブジェクトに関するその他の便利なユーティリティ

**Vision Guide 7.0**のヒストグラム機能は、サーチウィンドウ内のグレイスケール値の分布を視覚的に表示できるため、とても便利な機能です。**Vision Guide**ヒストグラムを使用して、**Geometric**オブジェクトのイメージの状態を調べることができます。

### 6.2.3 Correlationオブジェクト (相関サーチ)

#### Correlationオブジェクトの概要

Vision Guide 7.0では一般的によく使用されるツールです。Correlationオブジェクトが登録されると、登録された対象物を、非常に迅速にかつ高精度で検出して測定することができます。Correlationオブジェクトは、次のようなタイプのアプリケーションによく用いられます。

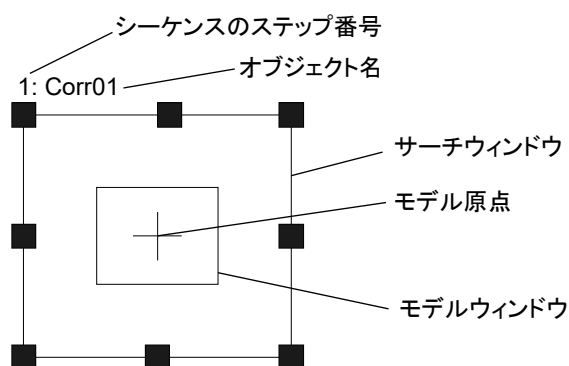
**Alignment** 既知のオブジェクトについて、オブジェクト上に対象物 (つまり、登録マーク) を位置づけることにより、位置と方向を決定します。これは、パーツ位置を検出するのに使われ、ロボットをパーツのピックアップアンドプレイス位置にガイダンスするのに役立ちます。

**Gauging** パーツ検査のために、長さ、半径、角度、その他の重要な寸法を計測します。

**Inspection** パーツの欠落、あるいは判読困難なプリント印刷などの単純な欠陥を探します。

#### Correlation オブジェクトのレイアウト

Correlationオブジェクトには、下に示すようにサーチウィンドウ、モデルウィンドウがあります。



Correlationオブジェクトのレイアウト

#### Correlationオブジェクトのプロパティー

下の一覧は、Correlationオブジェクトプロパティーの概略です。各プロパティーの詳細は、Vision Guide 7.0プロパティー&リザルトリファレンスを参照してください。

プロパティー	説明
AbortSeqOnFail	オブジェクトの実行に失敗した (不良判定になった)とき、直ちにシーケンス全体を中断し、それ以降のオブジェクトを処理しないように指定 デフォルト: False
Accept	対象物が検出されたと判定するための形状スコアの値を指定 対象物のスコアが設定値を超えているものを検出します。設定値を低く設定すると、誤検出の原因となります。 デフォルト: 700

プロパティ	説明
AngleAccuracy	角度サーチの精度を度数で指定 デフォルト: 1
AngleEnable	<p>           相関サーチが角度検出付き (回転) サーチを実行するかどうかを指定            Correlation オブジェクトのモデル登録の前に指定されます。            デフォルト: False         </p>
AngleMaxIncrement	<p>           角度検出付きサーチを実行する相関モデルの登録のための角度最大増加量            設定範囲の最大: 10            デフォルト: 10         </p>
AngleOffset	<p>           回転のオフセット値を設定            デフォルト: 0.000         </p>
AngleRange	<p>           一連の回転モデルの登録範囲を指定            登録範囲の最大値: 45            デフォルト: 10         </p>
AngleStart	<p>           角度サーチの中心を指定            デフォルト: 0         </p>
CalRobotPlacePos	設計時とプログラム実行時に RobotPlacePos をキャリブレーションする
Caption	<p>           見出しをCorrelationオブジェクトに割りあてる            デフォルト: 空欄         </p>
CenterPointObject	<p>           オブジェクトの中心点として用いられる位置を指定            このプロパティを“Screen”に設定すると画面上の任意の位置にオブジェクトを配置できますが、他のビジョンオブジェクトを指定した場合、そのオブジェクトのPixelX, PixelYリザルトに中心点が設定されます。            デフォルト: Screen         </p>
CenterPntObjResult	<p>           CenterPointObjectプロパティからどのリザルトを使うのかを指定            Allを指定すると、指定されたビジョンオブジェクトリザルトのすべて (NumberFound)にCorrelationオブジェクトが適用されます。            デフォルト: 1         </p>
CenterPntOffsetX	<p>           サーチウィンドウの中心をCenterPointObjectプロパティで位置決めしたあとに、Xオフセットを設定、または返す            デフォルト: 0         </p>
CenterPntOffsetY	<p>           サーチウィンドウの中心をCenterPointObjectプロパティで位置決めしたあとに、Yオフセットを設定、または返す            デフォルト: 0         </p>

プロパティ	説明
CenterPntRotOffset	<p>中心点の XY オフセット値 (CenterPntOffsetX, CenterPntOffsetY)を、CenterPointObject の Angle リザルトにしたがって回転するか否かを指定</p> <p>SearchWinType が“RotatedRectangle”に設定されている場合、サーチウィンドウは Angle リザルトにしたがって回転します。</p> <p>デフォルト: False</p>
CheckClearanceFor	クリアランス (隙間)を確認するオブジェクトを設定
ClearanceCondition	クリアランス (隙間)の判断方法を指定
Confusion	<p>検索する画像イメージ中に予想される複雑さの度合い量を示す</p> <p>検索している実際の対象物ではない対象物で得られる形状スコアの最大値です。</p> <p>デフォルト: 800</p>
CoordObject	<p>リザルトをコピーするCoordinatesオブジェクトを指定</p> <p>コピー処理はオブジェクトが実行される時に行われ、Decisionの分岐機能などで実行されなかった場合、コピー処理は行われません</p> <p>デフォルト: None</p>
CurrentResult	<p>オブジェクトウィンドウのリザルトリストに表示するリザルト、または、1つのサーチウィンドウ内にある対象物などの複数のリザルトを、システムが検出する設定のときにデータを返すリザルトを定義</p> <p>デフォルト: 1</p>
Description	<p>ユーザー指定の説明を設定</p> <p>デフォルト: 空欄</p>
EditWindow	被検索領域の検出マスクを定義
Enabled	<p>このオブジェクトを実行するかを指定</p> <p>デフォルト: True</p>
FailColor	<p>オブジェクトの検出不良時の色を選択</p> <p>デフォルト: Red</p>
Frame	<p>指定されたフレームに関して、作業中のオブジェクトの検索位置を規定</p> <p>Correlationオブジェクトをフレームに関係して位置決めすることができます。</p> <p>デフォルト: None</p>
FrameResult	<p>Frameの何番目のリザルトを使用するかを指定</p> <p>デフォルト: 1</p>
Graphics	<p>表示するグラフィックを指定</p> <p>デフォルト: 1 - All</p>

プロパティ	説明
LabelBackColor	オブジェクトラベルの背景の色を設定 デフォルト: Transparent (透明)
ModelObject	検索に使用するモデルを決定 デフォルト: Self
ModelOrgAutoCenter	モデルは固定基準点を持っていて、モデルウィンドウではモデルの位置はその基準点で表されます。この固定基準点は、モデル原点と呼ばれています。 <b>ModelOrgAutoCenter</b> プロパティによって、モデル原点は、モデルウィンドウの中心に位置します。 デフォルト: True
ModelOrgX	モデル原点のX座標値 (サブピクセルで設定可)
ModelOrgY	モデル原点のY座標値 (サブピクセルで設定可)
ModelWin	実行時のみ 1コールでモデルウィンドウの左端, 上端, 高さ, 幅のパラメーターを設定、または返す
ModelWinAngle	モデルウィンドウの角度を定義
ModelWinCenterX	モデルウィンドウの中心のX座標値を定義
ModelWinCenterY	モデルウィンドウの中心のY座標値を定義
ModelWinLeft	モデルウィンドウの最左端の位置を定義
ModelWinHeight	モデルウィンドウの高さを定義 デフォルト: 50
ModelWinTop	モデルウィンドウの最上端の位置を定義
ModelWinType	モデルウィンドウのタイプを定義
ModelWinWidth	モデルウィンドウの幅を定義
Name	<b>Correlation</b> オブジェクトに独自のオブジェクト名を割りあて デフォルト: Corr01
NumberToFind	作業中のサーチウィンドウで検出する対象物の数を定義 デフォルト: 1
PassColor	オブジェクトの検出良時の色を選択 デフォルト: LightGreen
PassType	オブジェクト検出時の良否判定の条件を定義 デフォルト: SomeFound
RejectOnEdge	サーチウィンドウのエッジにかかって検出されたパーツを除外するかどうか指定 通常は、Trueで使用してください。 誤検出が発生した結果、サーチウィンドウに接する検出を除外でき、誤検出リスクを軽減できます。 デフォルト: False

プロパティ	説明
SaveTeachImage	モデルをティーチするときに、画像ファイルに保存するかを設定
ScoreMode	Fail時のリザルトを表示させるための閾値を設定、または返す
SearchWin	実行時のみ 1コールでサーチウィンドウの左端, 上端, 高さ, 幅, または、中心のX座標, 中心のY座標, 円内周の半径サイズ, 円外周の半径サイズのパラメーターを設定、または返す
SearchWinAngle	被検索領域の角度を定義
SearchWinAngleEnd	被検索領域の終了角度を定義
SearchWinAngleStart	被検索領域の開始角度を定義
SearchWinCenterX	被検索領域の中心のX座標値を定義
SearchWinCenterY	被検索領域の中心のY座標値を定義
SearchWinHeight	被検索領域の高さを定義 (単位: ピクセル) デフォルト: 100
SearchWinLeft	被検索領域の最左端を定義 (単位: ピクセル)
SearchWinPolygonPointX1	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第1頂点のX座標値を定義
SearchWinPolygonPointY1	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第1頂点のY座標値を定義
SearchWinPolygonPointX2	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第2頂点のX座標値を定義
SearchWinPolygonPointY2	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第2頂点のY座標値を定義
SearchWinPolygonPointX3	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第3頂点のX座標値を定義
SearchWinPolygonPointY3	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第3頂点のY座標値を定義
SearchWinPolygonPointX4	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第4頂点のX座標値を定義
SearchWinPolygonPointY4	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第4頂点のY座標値を定義
SearchWinPolygonPointX5	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第5頂点のX座標値を定義
SearchWinPolygonPointY5	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第5頂点のY座標値を定義
SearchWinPolygonPointX6	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第6頂点のX座標値を定義
SearchWinPolygonPointY6	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第6頂点のY座標値を定義

プロパティ	説明
SearchWinPolygonPointX7	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第7頂点のX座標値を定義
SearchWinPolygonPointY7	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第7頂点のY座標値を定義
SearchWinPolygonPointX8	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第8頂点のX座標値を定義
SearchWinPolygonPointY8	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第8頂点のY座標値を定義
SearchWinPolygonPointX9	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第9頂点のX座標値を定義
SearchWinPolygonPointY9	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第9頂点のY座標値を定義
SearchWinPolygonPointX10	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第10頂点のX座標値を定義
SearchWinPolygonPointY10	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第10頂点のY座標値を定義
SearchWinPolygonPointX11	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第11頂点のX座標値を定義
SearchWinPolygonPointY11	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第11頂点のY座標値を定義
SearchWinPolygonPointX12	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第12頂点のX座標値を定義
SearchWinPolygonPointY12	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第12頂点のY座標値を定義
SearchWinRadiusInner	被検索領域の円内径を定義
SearchWinRadiusOuter	被検索領域の円外径を定義
SearchWinTop	被検索領域の最上端を定義 (単位: ピクセル)
SearchWinType	被検査領域のタイプ (Rectangle, RotatedRectangle, Circle, Arc, Polygon)を定義
SearchWinWidth	被検索領域の幅を定義 (単位: ピクセル) デフォルト: 100
ShowModel	教示モデルをさまざまなズーム設定によって表示 モデルの原点変更や検出マスクを設定
SkewFitEnable	モデルに対して歪み変形を適応するかを設定 デフォルト: False
Sort	オブジェクトのリザルトを並び替える順番を選択 デフォルト: 0 - None
Timeout	Correlationオブジェクトの最長サーチ時間を設定、または返す

## Correlationオブジェクトのリザルト

下の一覧は、Correlationオブジェクトリザルトを簡単に説明した概略です。各リザルトの詳細は、Vision Guide 7.0プロパティ&リザルトリファレンスを参照してください。

リザルト	説明
Angle	検出されたパーツの回転量を角度で返す
CameraX	カメラ座標系における検出パーツ位置の (モデル原点を基準にした)X座標を返す (単位: mm)
CameraY	カメラ座標系における検出パーツ位置の (モデル原点を基準にした)Y座標を返す (単位: mm)
CameraXYU	実行時のみ カメラ座標系における検出パーツ位置のCameraX座標, CameraY座標, CameraU座標を返す
ClearanceOK	クリアランス (隙間)の判断結果を返す
Found	オブジェクトが検出されたかどうか (つまり、対象物あるいはパーツがAcceptプロパティの現在の設定を超える形状スコア値を持っているかどうか)の値を返す
FoundOnEdge	Correlationオブジェクトがサーチウィンドウのエッジに密接して検出されたときに、“True”を返す  FoundOnEdgeが “True”のとき、Foundリザルトは “False”に設定されます。
NumberFound	検出されたオブジェクト数を返す (最小0個から、最大はNumberToFindプロパティで設定した検出個数になります。)
Passed	検出結果が良か否かを返す
PixelX	検出パーツ位置の (モデル原点を基準にした)X座標をピクセルで返す
PixelY	検出パーツ位置の (モデル原点を基準にした)Y座標をピクセルで返す
PixelXYU	実行時のみ 検出パーツ位置のPixelX座標, PixelY座標, PixelU座標をピクセルで返す
RobotX	ロボット座標系における検出パーツ位置の (モデル原点を基準にした)X座標を返す
RobotY	ロボット座標系における検出パーツ位置の (モデル原点を基準にした)Y座標を返す
RobotU	ロボット座標系における検出パーツ位置のU座標を返す
RobotXYU	実行時のみ ロボット座標系における検出パーツ位置のRobotX座標, RobotY座標, RobotU座標を返す
Scale	実行時に検出された対象物のスケーリング値を返す



リザルト	説明
Score	実行時に検出された対象物が、Correlationオブジェクトが検索するモデルに一致するレベルを表す0から1000のINTEGER数値を返す
ShowAllResults	特定のビジョンオブジェクトのリザルトすべてを、表形式でダイアログに表示  リザルトの比較が簡単にできます。
SkewDirection	実行時に検出された対象物の歪み方向を返す
SkewRatio	実行時に検出された対象物の歪み率を返す
Time	オブジェクト処理に要した時間を返す (単位: ミリ秒)
TimedOut	オブジェクト実行がタイムアウトで終了したか否かを返す

### 正規化関連の基本的知識

Correlation (相関サーチ)の目的は、サーチウィンドウで、事前に登録された1つ以上の対象物を位置決めし、測定することです。Correlationオブジェクトは、パーツの検出, 対象物の有無の検出, パーツの欠陥の検出など、さまざまなものの検出に利用することができます。

Vision Guide 7.0は、多様なビジョンオブジェクトツールを装備していますが、Correlationオブジェクトは、高速でしかも信頼性が高いため、一般的によく用いられるツールです。

例えば、多くのアプリケーションで、オブジェクトのエッジを検出するのにエッジツールを使用しますが、同一の領域内にEdgeオブジェクトを混同させるようなエッジが多くある場合は、Correlationオブジェクトをエッジ位置の検出に使用することができます。

また、モデルが登録できる場合にはBlobオブジェクトのかわりに、より信頼性の高いCorrelationオブジェクトが使われる場合もあります。

ここでは、Correlationオブジェクトに適用できるサーチツールの基本的知識について説明します。説明する項目は次のとおりです。

- 対象物とモデル: 説明
- 検索の基本的概念
- 正規化相関
- 正規化相関形状スコア
- 正規化相関最適化 (AcceptおよびConfusion)
- AcceptプロパティとConfusionプロパティの値の設定
- AcceptプロパティとConfusionプロパティの応用知識
- Correlationオブジェクトと回転
- 角度検索のためのモデル登録
- 検索の反復性と精度
- カメラと対象物間距離のキャリブレーション

### Correlationオブジェクトのモデルと対象物

Correlationオブジェクトを使うときは、対象物とモデルの違いをよく理解していただくことが重要となります。

対象物とは、サーチウィンドウの領域中のグレイレベルの特定パターンのことです。対象物には、数個のピクセルの単純なエッジから、数万個のピクセルの複雑なパターンまであります。

相関サーチオペレーションでは、領域中の対象物が、事前にその対象物を登録されたモデルに、どの程度一致するかを測定します。

対象物の理想的な表現であるモデル、あるいはテンプレートに対して、対象物とは、実際のサーチウィンドウ内に現実存在するもののことです。

モデルとは、対象物を表すのに用いられるグレイレベルのパターンのことです。モデルは、テンプレートマッチングシステムにおけるテンプレートと同じものです。

モデルに2つのグレイレベルがあるときは、バイナリモデルと呼ばれ、モデルに3つ以上のグレイレベルがあるときは、グレイモデルと呼ばれています。

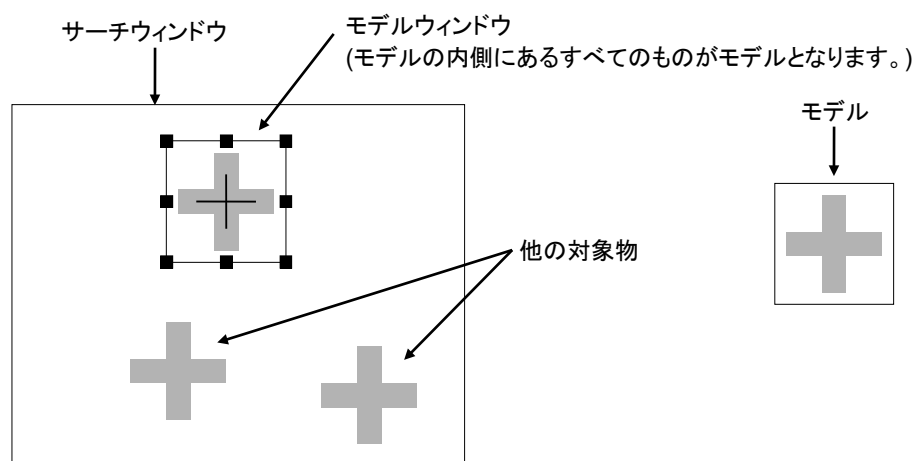
グレイモデルは、バイナリモデルに比べてより強力であり、真の対象物に近い対象物を表わすことができるため、Vision Guide 7.0で使われるモデルは、すべてグレイモデルです。グレイモデルを使うことにより、より信頼性の高い検出結果を得ることができます。

一般的な検索では、1つのサーチウィンドウで登録された代表モデルを使って、そのサーチウィンドウ内にある代表モデルの類似対象物を検索します。図は、対象物 (十字線)を含むサーチウィンドウを示します。

十字線のモデルを登録するには、モデルウィンドウを定義し、実行パネルの<ティーチ>ボタンをクリックしてください。

(モデルの登録については、後述の「Correlationオブジェクトの使用方法」を参照してください。)

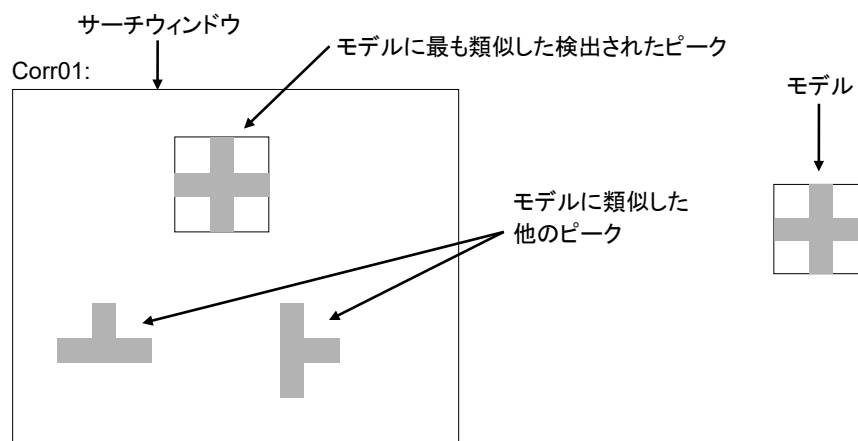
その結果、モデルは右側の図に示すとおりに作成され、サーチウィンドウ内の他の十字線を検索するのに使用することができます。



対象となる数個の対象物を含むサーチウィンドウ(左側)と画像イメージから登録されたモデル(右側)

## 検索の基本的概念

検索では、サーチウィンドウ内でモデルが最も類似している領域を検出することによって、対象物を位置づけます。図は、モデルとサーチウィンドウ、およびサーチウィンドウ内でモデルに最も類似している領域 (ピーク) を示します。このようなモデルは、例えばプリント基板上の基準マークのような対象物を検索するのに使えます。ロボットは、サーチファンクションから戻された位置データを使って、ボード上の部品取り付け位置、あるいは、ボード自体の位置決めの場所を検出できます。



サーチウィンドウ, モデル, ピーク

サーチウィンドウ内のモデルの検索方法は数多くありますが、最も一般的な方法は、モデルとの一致を検出する全数検索方法です。この方法では、モデルはサーチウィンドウ中のあらゆる場所で評価され、最も類似性の高い場所がリザルトとして返されます。この全数検索方法では、信頼性の高いリザルトを返すことができますが、半面、処理が非常に遅くなる場合があります。例えば、サーチウィンドウが36×36ピクセルの正方形で、モデルは6ピクセルの正方形としたとき、モデルと一致するものの場所を検出するには、961ヶ所全部で、類似性の評価を行わなければならないことになります。

しかし、Vision Guide 7.0のサーチ方法では、この全数検索方法を大幅に改良しています。まず、サーチウィンドウをスキャンして、一致する可能性がある場所の位置を検出し、次に、その場所についてだけ、類似性を評価します。その中から、最も一致した場所を返すので、全数検索方法に比べて、数千倍も高速の実行時間でリザルトを返すことができます。

## 正規化相関

正規化相関とは、サーチウィンドウ内の線形差異やモデル輝度の線形差異には左右されずに、画像イメージとモデル間の幾何学的類似性を測定する方法です。

正規化相関は、強力かつ安定なため、Vision Guide 7.0のCorrelationオブジェクトの検索アルゴリズムとして使われています。正規化相関の値は、次に示す状況においても変化することがありません。

- すべてのサーチウィンドウあるいはモデルピクセルを、ある定数で乗算した場合
- すべてのサーチウィンドウあるいはモデルピクセルに、ある定数を加算した場合

正規化相関の最も重要な特性は、正規化相関の値が、サーチウィンドウあるいはモデルの線形な輝度の変化には左右されない、ということです。これは、非常に重要な特性です。

なぜなら、画像イメージの輝度とコントラストの一切は、照度、反射性、カメラの絞り、センサーのゲインとオフセット (可能な自動ゲイン制御回路など)、画像イメージデジタイザのゲインとオフセットなどの要因によって決定され、これらはすべて、ほとんどの製造の場面でコントロールが困難だからです。例えば、電球は経時変化し、周囲の光量レベルは1日の時間の推移とともに変化し、カメラとデジタイザは故障する可能性や交換の必要性が生じ、検出される対象物の反射性は変化する可能性があります。

正規化相関のもう1つの重要な特性は、形状スコア値 (次の「正規化相関の形状スコア」の項参照)が絶対的な有意性を持っているということです。つまり、すべてのモデルとすべてのサーチウィンドウについて同一の完全な一致を定義することができるということです。

この特性は、画像イメージの輝度とコントラストに左右されないばかりではなく、モデルの輝度、コントラスト、サイズにも左右されません。そのため、検査アプリケーションでの対象物の質の測定に、形状スコアを使うことができます。

サーチウィンドウとモデルについて計算した形状スコアが900であった場合、サーチウィンドウとモデルは非常に類似していることになります。

形状スコアが100であった場合は、サーチウィンドウとモデルは類似していません。

これらのステートメントは、モデルあるいはサーチウィンドウ領域について何もわかっていない場合でも有効です。つまり、相関係数は、絶対的な意味を持っているということです。

### 正規化相関形状スコア

正規化相関では、対象物の形状スコア (モデルへの類似程度)は、0~1000までの値として定義されます。形状スコアが大きくなると、対象物とモデルとの類似性は大きくなります。(形状スコア1000は、完全一致を意味します。)

形状スコアは、CorrelationオブジェクトからScoreリザルトとして返される値です。Scoreリザルトの詳細は、Vision Guide 7.0プロパティ&リザルトリファレンスを参照してください。

### 正規化相関最適化 (AcceptおよびConfusion)

従来のテンプレートマッチングでは、実際的な検索アプリケーションのほとんどで長い処理時間がかかります。これは、すべての可能性のある検索位置で全数相関に基づいた処理を行っているためです。

この問題を解決するには、適切な指向検索の方法をとることが必要です。指向検索では、検索中に引き出されたデータを使って、一致する可能性のない場所を回避し、一致する可能性のある場所を、検索の指向対象とします。

相関サーチでは、ヒルクライミング法と呼ばれる指向検索を採用しています。ヒルクライミング法では、ローカルデータ全体から、常に上り最大急勾配の方向に移動するので、1つあるいは複数の変数のファンクションピークを検出することができます。あるポイントについて両隣りの各ポイントが低いときに、ピークに到達したとします。

従来のヒルクライミング法には、次に示すような欠点があります。

- 開始点の場所によって、他に最大値があっても、まちがったピークを得ることがあります。
- プラトー (平坦域)に到達した場合、検索の方向をどちらにするか決定する余地がありません。

開始点があまりにも少ないと、有意性のあるヒルを見落としてしまう可能性があります。また、開始点があまりにも多いと、別の方向から同じヒルに上るという事態が生じてしまうため、ヒルクライミング法の特徴であるスピード優位性が落ちてしまいます。

しかしVision Guide 7.0では、サーチウィンドウ領域とモデルに基づいたヒルについて、次のような見積りを実施することにより、上述の問題を克服することができます。

相関サーチは、数学的にフィルター演算と同じ働きをします。相関ファンクションは、特定の空間周波数を増幅し、その他の空間周波数を減衰する、特定の既知のフィルター (モデル)の出力です。さらに、サーチウィンドウの指定部分の周波数成分がモデルの周波数成分と類似していない場合は、この領域にヒルクライミングの開始点を置く必要はありません。

システムでは、モデルの伝達ファンクションを確認することによって、相関ファンクションの空間周波数成分を見積ります。そして、その値によって、ヒルの最小分布と最小サイズを見積ることができます。最小分布がわかれば、どの地点からヒルクライミングを開始するか計画することができます。また、ヒルの最小サイズがわかれば、誤ったピークに到達する可能性を避けることができます。

モデルから得られる情報の他に、1組のVision Guideプロパティ、つまりAcceptプロパティとConfusionプロパティを規定することにより、検索をコントロールします。

Acceptプロパティは、形状スコアを指定し、この形状スコアと等しいかそれ以上の場合に、特徴は「検出された」と判定されます。(つまり、Foundリザルトは“True”を返します。) ヒルの高さの大まかな見積りが、サーチウィンドウの指定の領域におけるAcceptプロパティの値を超えない場合、その領域のヒルクライミングは終了します。

Confusionプロパティは、サーチウィンドウ内に予想される複雑さの度合を示します。具体的には、検索している実際の対象物ではない対象物で得られる形状スコアの最大値です。これによって、システムは、検索すべきシーンについて重要なヒントが与えられます。つまり、対象物が複雑度のしきい値を超える形状スコアを獲得した場合は、必ず検索している対象物でなければなりません。

システムでは、Confusionプロパティと予想されるリザルトの数(NumberToFindプロパティにより指定)を使って、どのヒルに上る必要があり、どのヒルには上る必要がないのかを判断します。具体的には、形状スコアがConfusionプロパティのしきい値とAcceptプロパティのしきい値を超える対象物に対して、それが予定した個数検出されると、そこで検索を終了します。

検索を開始するには、複数のヒルクライミングを並行して実施しなければならず、開始点として、モデルの伝達ファンクションによって複数の位置が決定されます。ヒルクライミングが進行するにしたがって、各々のヒルの高さの見積り精度は高くなり、最終的には真のピークが検出されます。

ヒルクライミングは、複数のヒルについて、並行して実施され、一度に1ステップずつ、Confusionプロパティで設定されたしきい値を超える見積り高さを持つヒルがなくなるまで、処理が続行します。ヒルクライミングがConfusionプロパティのしきい値に到達したとき、そのヒルのピークに上ります。

### Confusionプロパティのしきい値とAcceptプロパティのしきい値の設定方法

Confusionプロパティのしきい値とAcceptプロパティのしきい値によって、Correlationオブジェクトの検索速度は影響を受けます。

Acceptプロパティは、どのような場合にシーンの一定領域での検索を続行するかセントを与えることによって、検索速度に影響を与えます。

Acceptプロパティが高い値に設定された場合は、対象物はモデルに非常に類似したものとなるので、多くの領域が大まかな確認によって排除され、それ以上の検索が続行されるようなことはありません。

Acceptプロパティが低い値に設定された場合は、モデルにわずかでも類似している対象物はAcceptプロパティのしきい値を超えてしまうので、シーン中のより多くの領域について詳細な確認が必要となります。

このように、Acceptプロパティの値が高ければ、Correlationオブジェクトの実行速度、つまり検索速度は速くなります。

Confusionプロパティは、検索速度に影響すると予想されるリザルトの数と相互作用します。Confusionプロパティと予想されるリザルトの数は一緒になって、画像イメージの検索可能領域の全検索が完了する前に、検索を中止させることができます。

Acceptプロパティを設定して、許容可能な「最悪ケース劣化」の対象物例をシステムによって検出できるようにしてください。劣化は、欠陥、スケール、回転、ビデオノイズなどによって発生することがあります。

Vision Guide 7.0では、Acceptプロパティに“700”のデフォルト値を設定しています。この値は、ほとんどのアプリケーションにとって適切な開始点です。ただし、試行や訂正を重ねることによって、状況に合わせて最適な値に設定することもできます。

なお、アプリケーションが適切に機能するためにスコアが“1000”、あるいはそれに近い値である必要はありませんので注意してください。対象物がこうむる劣化のタイプによりですが、アプリケーションによっては形状スコア“200”でも良好な位置情報が得られる場合があります。けれども、ほとんどのアプリケーションでは、Acceptプロパティに“500”以上の形状スコアを使用してください。

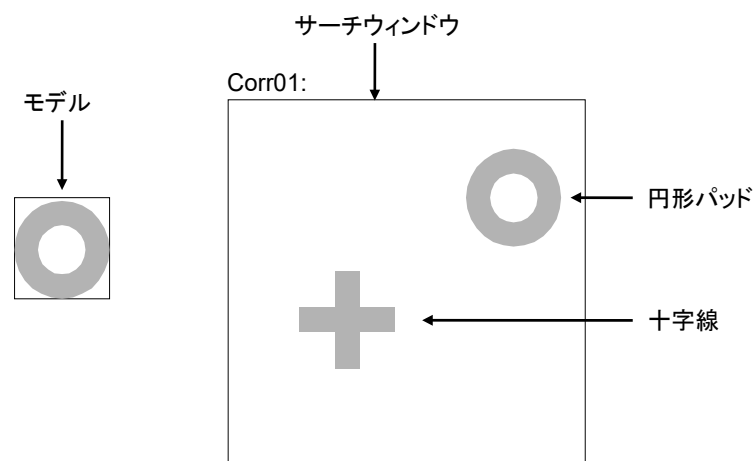
「誤認対象物」を獲得すると予想される最高値 (+誤差余裕) を基にして、Confusionプロパティを設定してください。

Confusionプロパティのしきい値は、Acceptプロパティのしきい値に等しいかそれ以上の値にします。高い値を設定すると、検索時間が長くなりますが、正しい対象物を確実に検出するためにはやむをえません。

Confusionプロパティのデフォルト値は“800”ですが、アプリケーション要件にしたがって調整してください。

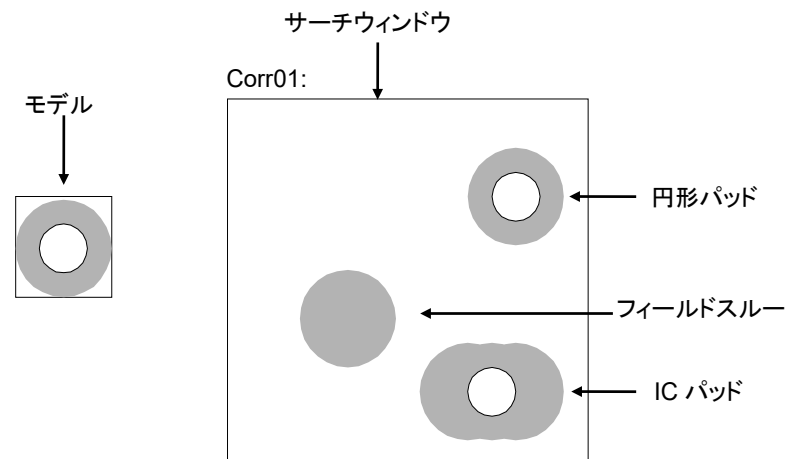
下図は、複雑度のほとんどないシーンを示しています。円形のパッドは、十字線とほとんど類似性はありません。したがって、Confusionプロパティは、かなり低い値 (500程度) に設定することができます。

許容可能な劣化量によりますが、Acceptプロパティは、通常、Confusionプロパティに等しいかそれ以下の値に設定します。このシーンでは劣化がほとんどないので、920の形状スコアが適切と予想されます。



複雑度がほとんどないシーン

下図は、複雑度の多いシーンを示しています。フィールドスルーとICパッドは、両方とも、円形パッドに類似しています。したがって、Confusionプロパティは、かなり高い値 (820程度) に設定しなければなりません。



複雑さの度合いが高いシーン

### AcceptプロパティとConfusionプロパティの応用知識

グレイ定数の領域を持つサーチウィンドウは、その領域で常に0の相関値を獲得します。シーンが、基本的に均一の背景を持つ場合 (例: 1枚の白い紙)はほとんどの場所で相関がないので、**Correlation**オブジェクトが何かを検出した場合、それは検索したい対象物であるはずであり、したがって、**Confusion**プロパティは低い値に設定することができます。

**Accept**プロパティと**Confusion**プロパティの設定の仕方によって、より迅速に対象物の位置を把握させることができる、と考えることもできます。

一般的に、これらのプロパティには慎重な値を設定すべきであって、正確に設定する必要はありません。最も慎重な設定方法は、**Accept**プロパティを低く、**Confusion**プロパティを高く設定する方法です。

検索するシーンについてわかっている情報が少ししかない場合は、より慎重な設定方法をとってください。検索は注意深く実行されますが、検索速度は遅くなります。

(これは、**Correlation**プロパティ位置リザルトを使ってロボットに動作先を示す場合、特に重要となります。)

検索するシーンについてわかっている情報が多い場合は、より大まかな設定方法を実施してください。例えば、ある一つの対象物を探していて、残りのシーンに何もいない場合は、より大まかな値を設定してください。注意深く検索する必要がないので、検索速度は速くなります。

### 複数リザルトのダイアログを使って、検索上の問題をデバッグする方法

たとえ同一生産ロット内でも、ときには、作業しているパーツがかなり変化したり、1つのパーツ上に類似対象物が2つ以上あったりする場合があります。このような場合、**Accept**プロパティを適切な値に設定した場合でも、その値に該当する他のパーツをまちがえて検索してしまう場合があります。これらのケースでは、何が起ころか予想するのが非常に困難です。

**ShowAllResults**ダイアログは、上記の問題をはじめとする種々の問題を解決するために作成されました。

1つのパーツ上に1つの対象物だけを検出したい場合は、複数検出を行って、**Vision Guide 7.0**が時々二次的な対象物を本来検出したい対象物として結果を返す理由を調べることができます。このような問題が発生するには、一般的に、いくつか異なった原因があります。

1. サーチウィンドウ中の2つ以上の対象物が非常に類似しているため、非常に近い値の**Score**リザルトを持つ場合
2. **Confusion**プロパティ、または**Accept**プロパティが十分高く設定されていないために、本来検出したい対象物よりも低いスコアを持つ対象物が**Accept**プロパティの設定値を満たしてしまう場合

**Vision Guide 7.0**を初めてご使用になる方にとって、上記の2つの状況は、サーチウィンドウ内で1つの対象物を検索する場合に極めて混同しやすいので注意してください。



ときには本来検出したい対象物が検出され、またときには別の対象物が検出されるようなときには、リザルトダイアログを使って、問題を特定してください。問題の原因を参照するための手順を、次に示します。

- (1) NumberToFindプロパティを“3”以上に設定してください。
- (2) Vision Guide 7.0開発環境からビジョンオブジェクトを実行してください。
- (3) <ShowAllResults>プロパティボタンをクリックして、リザルトダイアログを表示してください。
- (4) 検出された上位3つあるいはそれ以上の対象物のスコアを確認してください。
- (5) 1つまたは2つの対象物だけが検出された場合 (Vision Guide 7.0は、検出されたとみなされるそれらの対象物について、スコアの設定だけをします。) 1より多い対象物が検出されるように、Acceptプロパティを低くして、ビジョンオブジェクトをもう一度実行してください。(Acceptレベルは、リザルトダイアログを確認したあとで、元に戻すことができます。)
- (6) <ShowAllResults>プロパティボタンをクリックして、リザルトダイアログを表示してください。
- (7) 検出された上位3つあるいはそれ以上の対象物のスコアを確認してください。

上述のように、検出された上位3つあるいはそれ以上の対象物のスコアを確認すると、問題の原因が明確になります。このような問題は、ほとんど次に示すどちらかの場合に発生します。

1. 検出された各々の対象物が、Acceptプロパティの設定値より大きいスコアを持っている場合。このような場合には、Confusionプロパティをより高く設定し、適切な対象物だけを常に強制的に検出するようにして、Acceptプロパティのしきい値に一致するようなその他の対象物は返されないようにしてください。また、Acceptプロパティの値を変更することもできます。
2. 各々の対象物のスコアが非常に近接している場合。このような場合は、次の処理を行って、本来検出したい対象物が区別できるようにしてください。
  - サーチウィンドウを再調整し、検出対象物としてランダムに返される対象物がサーチウィンドウ内部に入らないようにします。
  - 本来検出したい対象物をモデルに再登録します。
  - アプリケーションの照明条件を調整して、本来検出したい対象物が、間違っ検出してしまう対象物のスコアより高いスコアを獲得するようにしてください。

複数リザルトの使用の詳細は、「6.2.24 1つのオブジェクトから複数のリザルトを返す」の項目を参照してください。

### Correlationオブジェクトと回転

テンプレートマッチング手順では、対象物とモデルとでサイズや角度に差がある場合、形状スコアと位置決め精度が低下します。差異が大きい場合には、形状スコアは非常に低くなるか、検索作業では対象物を検出することができなくなります。

角度とサイズ変更の正確な許容誤差は、モデルによって異なりますが、一般的に、角度については $3^{\circ} \sim 10^{\circ}$ 、サイズについては、 $2 \sim 5\%$ の範囲となっています。

角度に依存しない円のような点対称のモデルや、サイズに依存しない1つのエッジと1つの角で構成されるモデルなどは例外です。

サーチウィンドウ内に、2つの異なるシーンを想定してください。

1つのシーンは人間の顔の写真で、鼻がモデルとして教示されています。鼻はXY対称とはみなされないで、回転することによって、この対象物の位置の精度は極度の影響を受けます。2つめのシーンは、基準点 (図に示すような十字線) がモデルのプリント基板です。この場合、基準点のマーク (十字線) はXY対称なので、回転することによって、この対象物の位置の精度は、即座に圧倒的な影響を受けるということはありません。

基本的に、繊細な対象物 (鼻、花や木などの写真) が優位的なモデルは、回転に対して比較的、許容度が低くなります。例えば十字線のような対称的な対象物は、回転に対して比較的、許容度が高くなります。

ただし、以上のことに念頭においていただいた上で、回転角度の決定には、Polarオブジェクトを使うことをお勧めします。CorrelationオブジェクトはXY位置の検出に使い、PolarオブジェクトはCorrelationオブジェクトと関連して、XY位置を対象物の角度検出の中心点に使うことができます。PolarオブジェクトをCorrelationオブジェクトやその他のビジョンオブジェクトと関連して使用方法の詳細は、「6.2.6 Polarオブジェクト(ポーラサーチ)」



を参照してください。

有意性のある角度、またはスケールの変化が予想される場合に使用できる手法はいろいろありますが、主要な手法は次に示すとおりです。


- 複雑な対象物を、小型の単純な対象物に分解します。一般的に小型の単純モデルは、大型の複雑モデルに比べて、スケールや角度の変化に対する感度が極めて低くなります。

- 角度関係プロパティ (AngleEnable, AngleRange, AngleMaxIncrement) を使って、Correlationオブジェクトで作業するときに回転角度の検出をサポートします。  
単純な特徴に分解できない、複雑なシーンの複雑な対象物を位置づけるときに使用するとよいでしょう。ただし、この機能は、さまざまな角度で一連のモデルをもつことで実行されますので、通常検索に比べて処理は何倍も遅くなります。

#### NOTE



Correlationオブジェクトの角度検出付きサーチを行う場合は、Correlationオブジェクトのモデルを登録するときに、AngleEnableプロパティの設定を“True”としておかなければなりません。この設定によって、AngleRangeプロパティやAngleMaxIncrementプロパティによって定義されるさまざまな角度でCorrelationオブジェクトを登録できるようになります。

- パーツの回転角度を決定するには、CorrelationオブジェクトとPolarオブジェクトを一緒に使います。詳細は、「6.2.6 Polarオブジェクト(ポーラサーチ)」を参照してください。

### 角度検出のためのモデル登録

角度検出付きサーチを行うには、まず、一連の回転モデルを登録するよう、システムに指示しなければなりません。

そのための手順としては、**AngleEnable**プロパティを“True”に設定し、登録されるモデルの角度範囲を**AngleRange**プロパティを使って指定します。この方法で回転モデルを登録する場合は、その範囲内で等分間隔で角度を変えたさまざまな回転モデルが1組、自動的に作成されます。

また、その角度範囲内で、モデルが登録される角度最大増加量を指定することができます。この指定は、**Correlation**オブジェクトの**AngleMaxIncrement**プロパティの増加量の値を設定することによって実施できます。

ただし、**AngleMaxIncrement**プロパティに関しては、次の点に留意してください。

- 角度最大増加量を指定すると、モデル登録ファンクションは角度の自動増加量を選択し、選択された増加量と指定された角度最大増加量のうちで、自動的に小さい方の角度増加量を使用します。
- **AngleMaxIncrement**プロパティを“0”に設定すると、モデル登録ファンクションは角度の自動増加量を選択し、その角度増加量を使用します。この場合は、一般的に角度増加量の範囲を2°～5°の間に設定します。  
その結果、最小のモデル必要容量と最大の検索速度が得られますが、期待値に比べておおよそその結果になる場合もあります。

角度を正確に計測したい場合は、**AngleMaxIncrement**プロパティに、必要とする角度精度に対応するような増加量を設定しなければなりません。

角度増加量を小さく設定すれば、その分モデルに必要な容量は大きくなり、検索速度は遅くなりますので、注意してください。



角度の判断には、できる限り**Polar**オブジェクトを使用することをお勧めします。**Polar**オブジェクトは、ロボットガイダンスにビジョンを使用するときに要求される、非常に信頼性の高く高精度のリザルトを提供します。

角度付でモデルを登録するときは、モデルがサーチウィンドウの外側にはみ出すことなく回転できるように、サーチウィンドウは充分大きくなければなりません。

### 検索の反復性と精度

検索の反復性と精度とは、モデルのサイズと詳細 (形状, 対象物の粗さ, 形状の対称性) と、サーチウィンドウの中の対象物の劣化 (ノイズ, 欠陥, 回転やスケールによる影響) により影響を受けます。

無相関ノイズによる位置への影響を測定するには、劣化のない対象物を含む特定のサーチウィンドウで検索を実行し、この検索と完全に同じ検索をオブジェクトの位置を変更せずに再実行し (2番目の画像イメージをフレームバッファに取り込み)、位置についての測定結果を比較します。

そのための手順を次に示します。

- (1) モデルの定義後、実行パネルのオブジェクトの<Run>ボタンを2回以上クリックします。
- (2) <統計>ツールバーボタンをクリックします。
- (3) すると、統計ダイアログが表示され、2回のオブジェクト検索で得られた位置の差異を調べることができます。

劣化のない対象物の大型モデル (30×30)については、1/20ピクセルの繰返し精度を得ることも可能です。しかし、ほとんどの場合、1ピクセルをわずかに下まわる精度 (1/2, 1/3, 1/4 など)が現実的です。

検索精度は、劣化のない対象物を含む特定のサーチウィンドウで検索を実行し、対象物を移動して、検索で得られた位置の差異と実際の差異を比較することで計測できます。

劣化がなく、回転誤差やスケール誤差がなく、X方向、Y方向ともに十分なエッジがある大型モデル (30×30)については、検索精度は、1/4ピクセルになります。(これは、ビジョンシステムでの検索精度であって、ロボットに固有の不正確な精度は影響していませんので、注意してください。したがって、ロボットでパーツを移動するときは、ロボットメカニズムそのものの不正確な精度も考慮に入れなければなりません。)

回転とスケールが検索精度に与える影響は、モデルによって左右されます。

- 回転によって対称形となるモデルでは、良好な検索精度が得られます。
- 繊細な対象物で非対称形モデルでは、良好な検索精度は得られません。

### カメラと対象物間距離のキャリブレーション

最適な検索結果を得るためには、画像イメージ中の対象物の検索時のサイズが、モデルが登録されたときのサイズと同じでなければなりません。

同じカメラと同じレンズを使用すると仮定した場合、モデルを登録した時と検索を実行する時とで、対象物とカメラとの距離が変わると、サーチウィンドウ中では対象物の見かけ上のサイズが異なって見えます。つまり、カメラが対象物に近づくと対象物は大きくなり、カメラが対象物から遠ざかると対象物は小さくなります。



対象物とカメラとの距離が変わるときは、モデルを再登録しなければなりません。


## Correlationオブジェクトの使用方法

以上で正規化相関と検索処理がどのように動作するのか確認し、Vision Guide 7.0のCorrelationオブジェクトの使用方法を理解するための基本的知識を得ることができました。

ここでは、Correlationオブジェクトの使用方法を、次にリストアップしたステップごとに説明します。

- 新規Correlationオブジェクトの作成方法
- サーチウィンドウの位置とサイズの設定
- モデルウィンドウの位置とサイズの設定
- モデル原点の位置決め
- Correlationオブジェクトの関連プロパティの構成
- モデルの登録
- Correlationオブジェクトのテストとリザルトの確認
- プロパティの調整と再テスト
- 単一Correlationオブジェクトからの複数検出を使用した作業


これらのステップを開始する前に、まず新規ビジョンシーケンスを作成するか、使用する

ビジョンシーケンスを選択してください。作業するビジョンシーケンスがない場合は、 <新規シーケンス>ボタンをクリックすると、新規ビジョンシーケンスが作成できます。

すでにビジョンシーケンスを作成済みの場合は、Vision Guideウィンドウのシーケンスツリーをクリックして、作成済みのビジョンシーケンスを選択してください。

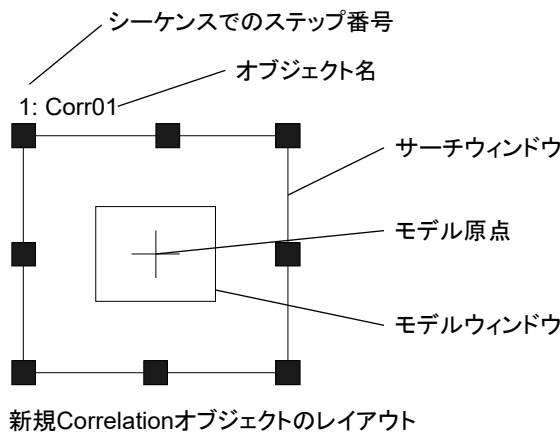
新規ビジョンシーケンスの作成方法、または作成済みビジョンシーケンスの選択方法の詳細は、「5. ビジョンシーケンス」を参照してください。

### Step 1: 新規Correlationオブジェクトの作成

- (1) Vision Guideツールバー - <全ツール> -  <Correlation (相関サーチ)>ボタンをクリックしてください。
- (2) マウスカーソルはCorrelationアイコンに変わります。
- (3) マウスを動かして、Correlationアイコンを、Vision Guideウィンドウの画像イメージ表示部まで、ドラッグしてください。そして、マウスの左側のボタンをクリックして、画像イメージ表示部にCorrelationオブジェクトを置いてください。
- (4) オブジェクト名は自動的に作成されますので注意してください。この例では、このシーケンスで最初に作成されたCorrelationオブジェクトなので、“Corr01”というオブジェクト名になります。(オブジェクト名の変更方法については後述します。)

Step 2: サーチウィンドウの位置とサイズの設定

画面上には、下図のようなCorrelationオブジェクトが表示されます。



- (1) Correlation オブジェクトの名称ラベルをクリックし、ボタンを押し続けながら、Correlation オブジェクトをドラッグして、サーチウィンドウの左上コーナーを好きな位置に合わせてください。
- (2) サイズハンドルをクリックして、マウスをドラッグすると、Correlation オブジェクトサーチウィンドウのサイズが変更できますので、適当な大きさに調整してください。(サーチウィンドウ内が、被検索領域になります。)



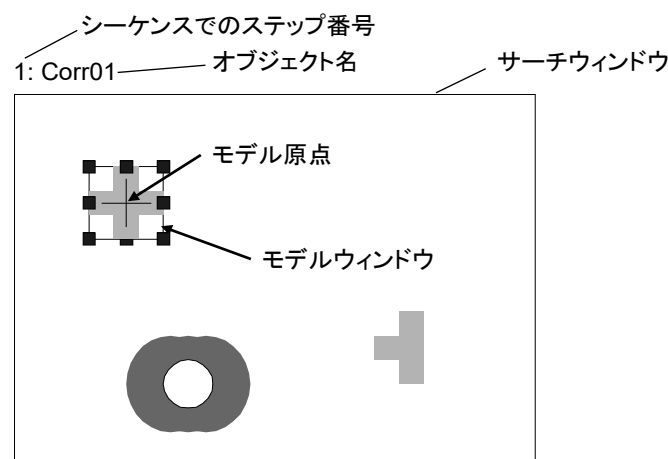
注意

- ビジョンシーケンスの実行結果は、外らん光や外部機器ノイズにより影響を受けることがあります。  
外らん光や外部機器ノイズにより、影響を受けると、取り込み画像が想定外の結果となり、検出位置結果がサーチエリア (検出エリア)内の不特定位置になる可能性があります。  
サーチエリアは、極力小さく、また誤検出に配慮した画像処理シーケンスの作成を行ってください。

Step 3: モデルウィンドウの位置とサイズの設定

- (1) 作業を行うCorrelationオブジェクトサーチウィンドウは、マゼンタ色になり、サイズハンドルがそれぞれの角と各辺の中央に表示されているはずですが、サイズハンドルがない場合は、Correlation オブジェクト名フィールドをクリックしてください。作業を行うCorrelation オブジェクトのサイズハンドルが表示されて、マゼンタ色になったら、手順2に進みます。
- (2) モデルウィンドウを形成するボックスの1辺をクリックしてください。この操作によって、モデルウィンドウが強調表示され、モデルウィンドウのサイズハンドルが表示されます。
- (3) Correlation オブジェクトのモデルとして登録する対象物を、モデルウィンドウで囲みます。まず、モデルウィンドウを形成するボックスの1辺をクリックして、マウスを押し続けながらモデルウィンドウをドラッグし、モデルウィンドウの左上コーナーの位置を合わせてください。
- (4) 次に、サイズハンドルをクリックしてマウスをドラッグし、モデルウィンドウのサイズを調整して、Correlation オブジェクトのモデルとして登録する対象物を囲んでください。

Correlationオブジェクトのレイアウトは、下図のように表され、サーチウィンドウは被検索領域をカバーし、モデルウィンドウは検索したい対象物を包囲しているはずです。実際のサーチウィンドウとモデルウィンドウは、この例のようには表されていないかもしれませんが、画面予想図として参考にしてください。



サーチウィンドウとモデルウィンドウの位置と、サイズ調整後のCorrelationオブジェクト

#### モデルウィンドウの位置とサイズを適切に設定するための注意点:

Correlationオブジェクトのモデルウィンドウの位置とサイズは、これによって検索する対象物が定義されるので、適切な調整を行うことが重要です。モデルウィンドウを作成する場合には、特に次の2点に留意してください。

- (1) 被検索領域であるサーチウィンドウは、なるべく小さい方が検索時間を短縮できます。  
また、特にパーツの回転が大きいと予想されるときには、サーチウィンドウを小さく設定し、モデルも小さく (パーツの一部などに) 設定できれば、パーツの回転による影響が少なくて済みます。
- (2) モデルウィンドウサイズを可能な限りサーチウィンドウサイズに近づけることによって、検索実行時間を短縮することができます。

また、2つのパーツが互いに接触するくらいに隣接しているような場合は、モデルウィンドウを実際の対象物より少しだけ (数ピクセル分) 大きめに設定しておく、対象物を他の物体と区別しやすくなります。

モデルウィンドウの最適なサイズは、それぞれのビジョンアプリケーションによって異なりますので注意してください。

#### Step 4: モデル原点の位置決め

モデル原点は、Correlationオブジェクトを実行するときに、対象物の位置として返されるモデル上の位置を定義します。つまり、モデル原点は、位置データが重要となる場合に、有意性のある場所に位置づけられていなければなりません。

例えば、Correlationオブジェクトを使ってパーツを検出し、ロボットにパーツのピックアップや配置をさせる場合は、モデル原点の位置は、ロボットが簡単にパーツをつかむことのできる場所であることが重要となります。

その位置は、RobotX, RobotY, RobotU, RobotXYUリザルトに基づくロボットの動作先になるからです。

新規Correlationオブジェクトが作成されると、ModelOrgAutoCenterプロパティはデフォルト値の“True”に設定されます。つまり、モデル原点は、モデルウィンドウの中心に自動的に位置づけられ、手動で移動させることができません。

モデル原点を移動させたい場合には、最初にModelOrgAutoCenterプロパティの値を“False”に設定する必要があります。この設定手順と、モデル原点を実際に位置決めする手順を次に示します。

- (1) Vision GuideウィンドウのフローチャートからCorrelationオブジェクトをクリックしてください。オブジェクトウィンドウのプロパティリストでModelOrgAutoCenterプロパティを探し、その設定値フィールドをクリックしてください。
- (2) ドロップダウンリストに、2つの選択項目、“True”と“False”が表示されますので、“False”をクリックしてください。これで、ModelOrgAutoCenterプロパティが“False”に設定され、モデル原点はマウスで移動することができます。
- (3) モデルウィンドウをクリックして、モデルウィンドウを強調表示させてください。
- (4) モデル原点をクリックして、マウスボタンを押し続けながら、モデル原点を新規の位置にドラッグしてください。なお、モデル原点を置くことができるのは、モデルウィンドウの境界内だけですので、注意してください。

### Step 5: Correlationオブジェクトプロパティの設定

Correlationオブジェクトプロパティの設定を行います。プロパティを設定するには、関連するプロパティの設定値フィールドをクリックし、新規の値を入力してください。あるいは、ドロップダウンリストが表示されている場合は、リスト中の該当項目をクリックしてください。

次に、Correlationオブジェクトで一般的に使われるプロパティのいくつかを示します。

Correlationオブジェクトをテストするときは、これらのプロパティを設定する必要はありません。ただし、実際にCorrelationオブジェクトで作業するのが初めての方は、この説明を参考にしてください。

その他、さまざまなビジョンオブジェクトの多くで使われる、AbortSeqOnFail, Graphicsなどのプロパティの詳細は、Vision Guide 7.0プロパティ&リザルトリファレンス、または「Correlationオブジェクトのプロパティ」リストを参照してください。



注意

- ビジョンシーケンスの実行結果は、外らん光や外部機器ノイズにより影響を受けることがあります。  
外らん光や外部機器ノイズにより、影響を受けると、取り込み画像が想定外の結果となり、検出位置結果がサーチエリア (検出エリア)内の不特定位置になる可能性があります。  
Accept, RejectOnEdgeなどのプロパティ値を適切に設定し、誤検出リスクを軽減してください。



Nameプロパティ	<p>新規に作成されたCorrelationオブジェクトに、デフォルトとして与えられるオブジェクト名は“Corrxx”です。xxは、1つのビジョンシーケンス内で複数のCorrelationオブジェクトを用いるとき、それらのCorrelationオブジェクトを区別するための番号です。</p> <p>ビジョンシーケンスで1番目に用いられるCorrelationオブジェクトには、“Corr01”というオブジェクト名がデフォルトで設定されます。オブジェクト名を変更したい場合は、Nameプロパティの設定値フィールドをクリックして、変更後の名称を入力し、リターンキーを押してください。画面上でCorrelationオブジェクト名を表示する場所は、すべて変更後の名称に変わります。</p>
Acceptプロパティ	<p>Acceptプロパティは、対象物が一致するべき、あるいは検出されたと判定するときの基準となる形状スコアの値を設定します。</p> <p>Scoreリザルトに返される値は、このAcceptプロパティに設定されている値との比較が行われます。デフォルト値は“700”で、Correlationオブジェクトの初回実行時には適切な値です。</p>
Confusionプロパティ	<p>サーチウィンドウの中に類似する対象物が複数ある場合、このConfusionプロパティを使うと、検索したい対象物を正確に獲得するのに役立ちます。デフォルト値は“800”で、Correlationオブジェクトの初回実行時には適切な値です。</p>
ModelOrgAutoCenterプロパティ	<p>最初にこのModelOrgAutoCenterプロパティを“False”に設定しておくともモデル原点の位置が変更できます。</p> <p>デフォルト: True</p>
Frameプロパティ	<p>このプロパティによって、事前に定義されていたFrameオブジェクトをCorrelationオブジェクトのリファレンスフレームとして選択することができます。Frameの詳細は、「6.12.3 Frameオブジェクト (フレーム)」の項目を参照してください。</p>
NumberToFindプロパティ	<p>検出したい対象物の数に合わせて、NumberToFindプロパティを1より大きい数値に設定することができます。これによって、Correlationオブジェクトは1つのサーチウィンドウ内で複数の対象物を検出することができます。</p>
AngleEnableプロパティ	<p>Correlationモデルを角度付で検索したい場合は、このプロパティを“True”に設定しなければなりません。複数のモデルを角度付で検索できるようにするには、Correlationオブジェクトのモデル登録の前に、このプロパティを“True”に設定しておかなければなりません。</p>
AngleMaxIncrementプロパティとAngleRangeプロパティ	<p>これらのプロパティは、AngleEnableプロパティと一緒に用いることにより、相関モデルを角度付で検索することができます。</p>
RejectOnEdgeプロパティ	<p>このプロパティによって、サーチウィンドウのエッジにかかって検出されたパーツを除外することができます。通常はこのプロパティをTrueに設定します。</p>

プロパティは、まずデフォルト設定のままにして次のステップに進み、後で必要に応じて設定を変更することができます。

### Step 6: Correlationオブジェクトのモデルの登録

Correlationオブジェクトには、検索するためのモデルが必要であり、モデルの登録と呼ばれるプロセスによって実行します。すでに、Correlationオブジェクトのモデルウィンドウは、モデルとして使いたい対象物を取り囲むように位置決めされているはずです。モデルの登録は、次に示す手順で完成します。

- (1) Correlationオブジェクトが、現在表示されているオブジェクトであることを確認してください。フローチャート、またはシーケンスツリーを見ると、どのオブジェクトが現在作業中のオブジェクトになっているか確認することができます。また、画像イメージ表示部で、どのオブジェクトがマゼンタ色で強調表示されているかによって確認することもできます。
- (2) 実行パネルにある、<ティーチ>ボタンをクリックしてください。モデルが登録されます。モデルの登録に要する時間は、ほとんどの場合わずか数秒です。ただし、AngleEnableプロパティが“True”に設定されているときにモデルを登録する場合は、わずかな角度ずつオフセットした複数のモデルを登録する必要があるため、モデルの登録には、かなりの時間を要します。

### Step 7: Correlationオブジェクトのテストとリザルトの確認

Correlationオブジェクトは、実行パネルのオブジェクトの<Run>ボタンをクリックするだけで簡単に実行できます。

Correlationオブジェクトが実行されると、リザルトが表示されますので確認してください。この時点で確認すべき主要なリザルトは、次のとおりです。

Foundリザルト	Correlationオブジェクトが検出されたかどうかの値を返す  検索している対象物が検出されると、リザルトとして“True”が返ります。対象物が検出されないと、Foundリザルトは“False”を返し、赤で強調表示されます。Correlationオブジェクトが検出されなかった場合、その一般的理由について Step 8を参照してください。
FoundOnEdgeリザルト	対象物がサーチウィンドウのエッジに接して検出されたときに、“True”を返す  この場合、Foundリザルトは“False”を返します。
Scoreリザルト	モデルと、モデルに最もよく似ている対象物とが、どの程度一致しているかを示す  Scoreリザルトの範囲は、0~1000で、1000はモデルと対象物が完全に一致したことを示します。検出された対象物が、検索したい対象物に一致している程度を確認するために、まず、Correlationオブジェクト実行後のScoreリザルトを確認してください。

Timeリザルト	Correlationオブジェクトの実行に要した時間を返す  サーチウィンドウとモデルウィンドウを小さくすることで、検索速度は速くなります。
NumberFoundリザルト	1つ以上のCorrelationオブジェクトを実行するとき、NumberFoundリザルトは、Correlationオブジェクトのモデルに一致した対象物の数を返す
Angleリザルト	Correlationオブジェクトの傾斜角度を返す  この角度は、モデルに設定された初期角度に基づいて計算されますが、ときには概略的な値であり、信頼性が高くないことがあります。回転角度の決定にはPolarオブジェクトを使うことをお勧めします。特にロボットガイダンスには、Polarオブジェクトを使うようにしてください。
PixelXリザルト	対象物のXY座標値 (ピクセル座標系)を返す
PixelYリザルト	この値は、検出された対象物のモデル原点の位置です。別の位置を返したい場合は、モデル原点の位置を変更し、モデルを再登録する必要があります。
CameraXリザルト	カメラ座標系における検出対象物のXY座標値を定義
CameraYリザルト	これらのリザルトは、カメラキャリブレーションが完了していない場合は“no cal”を返し、値を返しません。
RobotXリザルト	ロボット座標系における検出対象物のXY座標値を定義
RobotYリザルト	ロボットは、このXY座標値を動作先とすることができます。(他の変換、処理作業は不要です。)  この値は、検出された対象物のモデル原点位置です。別の位置を返したい場合は、まずモデル原点の位置を変更しモデルを再登録する必要があります。これらのリザルトは、カメラキャリブレーションが完了していない場合は“no cal”を返し、値を返しません。
RobotUリザルト	検出対象物の角度として、ロボット座標系に変換された値を返す  RobotUリザルトは、カメラキャリブレーションが完了していない場合は“no cal”を返し、値を返しません。
ShowAllResults	複数のリザルトで作業しているとき、ShowAllResultsの設定値フィールドにあるボタンをクリックすると、ShowAllResultsダイアログが表示され、現在作業中のビジョンオブジェクトのリザルトすべてを参照することができます。

## NOTE



上記のステップ例では、キャリブレーションを実行していませんので、RobotXYU, RobotX, RobotY, RobotU, CameraX, CameraY, CameraXYUのリザルトは、“no cal”を返します。これは、キャリブレーションを実行しなかったため、ロボット座標系あるいはカメラ座標系に関して座標リザルトを計算することができなかった、という意味です。詳細は、「7. ビジョンキャリブレーション」を参照してください。

### Step 8: プロパティの調整と再テスト

Correlationオブジェクトを数回実行することで、検出上の問題が見つかったり、プロパティ設定の微調整をする必要が出てきたりするかもしれません。

ここでは、一般的に生じる問題のトラブルシューティングや、微調整の方法について説明します。

#### Correlation オブジェクトのトラブルシューティング

##### CorrelationオブジェクトがFalseのFoundリザルトを返す場合

- 返されたScoreリザルトが、Acceptプロパティの設定より低くなっていないか確認してください。Scoreリザルトのほうが低くなっている場合は、Acceptプロパティの設定を少し低め(例えば、現在のScoreリザルトより小さい値)に変更し、Correlationオブジェクトを再度実行してみてください。
- 返されたFoundOnEdgeリザルトが、“True”かどうかを確認してください。Foundリザルトが“False”を返し、FoundOnEdgeリザルトが“True”を返した場合は、対象物は検出されたものの、その一部がサーチウィンドウに接触していることを意味しています。この場合は、サーチウィンドウを大きくし、対象物がサーチウィンドウに収まるようにしてください。サーチウィンドウのサイズ変更が不可能な場合は、カメラの位置を変更するか、モデルウィンドウのサイズを再調整してください。

##### Correlationオブジェクトが誤認対象物を検出した場合

- Acceptプロパティは、充分高い値に設定されているでしょうか。Acceptプロパティの設定値が低すぎると、検索対象となる対象物のかわりに、別の対象物が検出される可能性があります。
- Confusionプロパティは、充分高い値に設定されているでしょうか。Confusionプロパティの値は、Acceptプロパティの値に等しいか、それ以上の値でなければなりません。また、サーチウィンドウ内に、検索する対象物に類似した対象物がある場合は、Confusionプロパティの設定をさらに高い値に変更し、別の対象物を誤って検出しないようにしなければなりません。
- サーチウィンドウを調整して、検索する対象物に近づけ、その対象物だけを隔離するようにしてください。

### Correlation オブジェクトの微調整

オブジェクトが正常に動作するためには、通常、Correlation オブジェクトの微調整が必要となります。

Correlation オブジェクトの微調整に関する主要なプロパティー、およびモデル追加について、次に説明します。

**Acceptプロパティー** Acceptプロパティーを低くすると、Correlation オブジェクトの実行速度は速くなります。しかし、Acceptプロパティーの値が低いと、検出したい対象物とは別の対象物が検出される可能性も強くなります。Correlation オブジェクトを数回実行してみて、Scoreリザルトに返される形状スコアの値の予想がつけられるようになったら、Acceptプロパティーの値を調整してみてください。数回試行を重ね、より速い実行速度で信頼性の高い対象物を得ることができるような適切な値を設定してください。

**Confusionプロパティー** サーチウィンドウ中に類似特徴が複数ある場合は、Confusionプロパティーに相対的に高い値を設定する必要があります。これによって、検出したい対象物を、他の混同しやすい対象物と区別して検出することができます。ただし、Confusionプロパティーの値を高く設定すると、実行速度が遅くなります。サーチウィンドウ中に類似対象物がない場合は、Confusionプロパティーに低い値を設定して実行速度を速くすることができます。

**サンプルの追加** 現在のモデルと同じサイズのモデルウィンドウでティーチすると、「サンプルの追加」を選択できます。形や模様が少し違う、影の出方が変わる、など、モデルが少し変化する場合、変化したモデルを選択してモデルを追加すると、オブジェクト実行時のスコアが安定的になります。角度が大きくずれていたり、変化が大きくてモデルの追加に失敗したときは、もとのモデルが保持されます。

調整が完了し、Correlation オブジェクトのテストを行って、満足なリザルトを得ることができたら、ビジョンオブジェクトの作成ステップは完了します。

次のステップとして、他のビジョンオブジェクトを作成し、ビジョンシーケンス全体の設定やテストに進むことができます。

### Correlation オブジェクトに関するその他の便利なユーティリティ

この時点で、Vision Guide 7.0のヒストグラム機能を確認することもできます。

ヒストグラムは、サーチウィンドウ内のグレイスケール値の分布をグラフィックに表示できるので、非常に便利です。Vision Guideヒストグラムの使用方法の詳細は、「8.1 ヒストグラムの使用方法」を参照してください。

また、Correlation オブジェクトのリザルトを統計的に確認するのに、Vision Guide統計機能を使用することもできます。

Vision Guide統計機能の詳細は、「9. 統計ツール」を参照してください。

### 6.2.4 Blob オブジェクト (2値検査)

#### Blobオブジェクトの概要

Blobオブジェクトは、2値検査ツールを利用して、幾何学的特徴、位相的特徴、その他の画像イメージの特徴量を計算処理します。

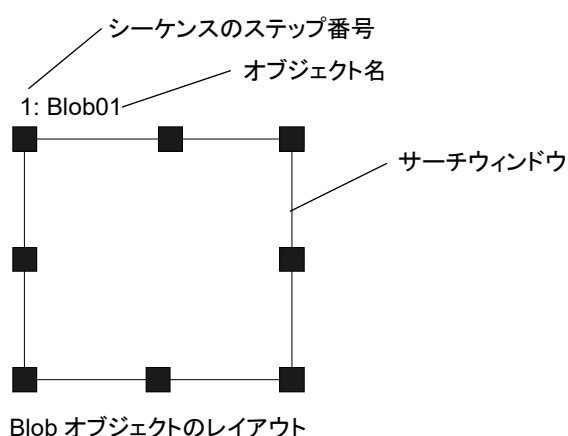
Blobオブジェクトは、画像イメージ中の対象物の有無、サイズ、方向を決定するのに便利です。例えば、Blobオブジェクトを使って、シリコンウェハー上のインクドットの有無やそのサイズと位置の検出、部品方向の決定、さらにはロボットガイダンスなどをすることができます。(ただし、回転方向の決定にはPolarオブジェクトを使用することをお勧めします。)

Blobオブジェクトで計算処理される幾何学的特徴は、次のとおりです。

- 領域と周
- 重心
- 主軸と慣性モーメント
- 連結性
- 外接長方形
- ピクセル座標系、カメラ座標系、ロボット座標系での重心の座標位置
- ブローブの穴、ざらつき、コンパクトさ

#### Blobオブジェクトのレイアウト

Blobオブジェクトのレイアウトは、Correlationオブジェクトのような長方形です。しかし、Blobオブジェクトにはモデルがありません。つまり、Blobオブジェクトのレイアウトには、モデルウィンドウやモデル原点は必要ありません。下のように、Blobオブジェクトはオブジェクト名とサーチウィンドウだけです。サーチウィンドウは、Blobを検索する領域を定義します。Blobオブジェクトの例を下に示します。



### Blobオブジェクトのプロパティ

Blobオブジェクトで使われるプロパティの一覧を次に示し、簡単に説明します。各プロパティの詳細は、Vision Guide 7.0プロパティ&リザルトリファレンスを参照してください。

プロパティ	説明
AbortSeqOnFail	オブジェクトの実行に失敗した (不良判定になった) とき、直ちにシーケンス全体を中断し、それ以降のオブジェクトを処理しないように指定 デフォルト: False
CalRobotPlacePos	設計時とプログラム実行時に RobotPlacePos をキャリブレーションする
Caption	見出しをBlobオブジェクトに割りあてる デフォルト: 空欄
CenterPointObject	オブジェクトの中心点として用いられる位置を指定 このプロパティを“Screen”に設定すると画面上の任意の位置にオブジェクトを配置できますが、他のビジョンオブジェクトを指定した場合、そのオブジェクトの PixelX, PixelYリザルトに中心点が設定されます。 デフォルト: Screen
CenterPntObjResult	CenterPointObjectプロパティから使用するリザルトを指定 Allを指定すると、指定されたビジョンオブジェクトリザルトのすべて (NumberFound)にBlobオブジェクトが適用されます。 デフォルト: 1
CenterPntOffsetX	サーチウィンドウの中心をCenterPointObjectプロパティで位置決めしたあとに、Xオフセットを設定、または返す デフォルト: 0
CenterPntOffsetY	サーチウィンドウの中心をCenterPointObjectプロパティで位置決めしたあとに、Yオフセットを設定、または返す デフォルト: 0
CenterPntRotOffset	中心点の XY オフセット値 (CenterPntOffsetX, CenterPntOffsetY)を、CenterPointObject の Angle リザルトにしたがって回転するか否かを指定 SearchWinType が RotatedRectangle に設定されている場合、サーチウィンドウは Angle リザルトにしたがって回転します。 デフォルト: False
CheckClearanceFor ClearanceCondition	クリアランス (隙間)を確認するオブジェクトを設定 クリアランス (隙間)の判断方法を指定

プロパティ	説明
CoordObject	リザルトをコピーするCoordinatesオブジェクトを指定 コピー処理はオブジェクトが実行される時に行われ、 Decisionの分岐機能などで実行されなかった場合、コピー処理は行われません デフォルト: None
CurrentResult	オブジェクトウィンドウのリザルトリストに表示するリザルト、または、1つのサーチウィンドウ内にある対象物などの複数のリザルトを、システムが検出する設定のときにデータを返すリザルトを定義 デフォルト: 1
Description	ユーザー指定の説明を設定 デフォルト: 空欄
EditWindow	被検索領域の検出マスクを定義
Enabled	このオブジェクトを実行するかを指定 デフォルト: True
FailColor	オブジェクトの検出不良時の色を選択 デフォルト: Red
FillHoles	2値画像の穴を埋めるかを選択 デフォルト: False
Frame	どの位置に置いたフレームを使用するかを指定 デフォルト: None
FrameResult	Frameの何番目のリザルトを使用するかを指定 デフォルト: 1
Graphics	表示するグラフィックを指定
LabelBackColor	オブジェクトラベルの背景の色を設定 デフォルト: Transparent (透明)
MaxArea	オブジェクトの上限領域を規定 被検出ブローブが取得するAreaリザルトは、MaxAreaプロパティに設定された値より低くなければなりません。 デフォルト: 100,000
MinArea	オブジェクトの下限領域を規定 被検出ブローブが取得するAreaリザルトは、MinAreaプロパティに設定された値より高くなければなりません。 デフォルト: 25
MinMaxArea	実行時のみ1ステートメントでMinAreaとMaxAreaを設定、または返す
Name	Blobオブジェクトに独自のプロジェクト名を割りあてる デフォルト: Blob01



プロパティ	説明
NumberToFind	サーチウィンドウの中で検出するオブジェクトの数を指定 デフォルト: 1
PassColor	オブジェクトの検出良時の色を選択 デフォルト: LightGreen
PassType	オブジェクト検出時の良否判定の条件を定義 デフォルト: SomeFound
Polarity	オブジェクトと背景の差 (明るい背景に暗いオブジェクトか、暗い背景に明るいオブジェクトか)を定義 デフォルト: 1 - DarkOnLight
RejectOnEdge	Trueに設定すると、サーチウィンドウのエッジにかかって検出されたブローブを無視します デフォルト: False
SearchWin	実行時のみ  1コールでサーチウィンドウの左端, 上端, 高さ, 幅, または, 中心のX座標, 中心のY座標, 円内周の半径サイズ, 円外周の半径サイズのパラメーターを設定、または返す
SearchWinAngle	被検索領域の角度を定義
SearchWinAngleEnd	被検索領域の終了角度を定義
SearchWinAngleStart	被検索領域の開始角度を定義
SearchWinCenterX	被検索領域の中心のX座標値を定義
SearchWinCenterY	被検索領域の中心のY座標値を定義
SearchWinHeight	被検索領域の高さを定義 (単位: ピクセル) デフォルト: 100
SearchWinLeft	被検索領域の最左端を定義 (単位: ピクセル)
SearchWinPolygonPointX1	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第1頂点のX座標値を定義
SearchWinPolygonPointY1	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第1頂点のY座標値を定義
SearchWinPolygonPointX2	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第2頂点のX座標値を定義
SearchWinPolygonPointY2	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第2頂点のY座標値を定義
SearchWinPolygonPointX3	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第3頂点のX座標値を定義
SearchWinPolygonPointY3	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第3頂点のY座標値を定義
SearchWinPolygonPointX4	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第4頂点のX座標値を定義

プロパティ	説明
SearchWinPolygonPointY4	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第4頂点のY座標値を定義
SearchWinPolygonPointX5	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第5頂点のX座標値を定義
SearchWinPolygonPointY5	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第5頂点のY座標値を定義
SearchWinPolygonPointX6	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第6頂点のX座標値を定義
SearchWinPolygonPointY6	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第6頂点のY座標値を定義
SearchWinPolygonPointX7	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第7頂点のX座標値を定義
SearchWinPolygonPointY7	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第7頂点のY座標値を定義
SearchWinPolygonPointX8	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第8頂点のX座標値を定義
SearchWinPolygonPointY8	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第8頂点のY座標値を定義
SearchWinPolygonPointX9	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第9頂点のX座標値を定義
SearchWinPolygonPointY9	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第9頂点のY座標値を定義
SearchWinPolygonPointX10	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第10頂点のX座標値を定義
SearchWinPolygonPointY10	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第10頂点のY座標値を定義
SearchWinPolygonPointX11	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第11頂点のX座標値を定義
SearchWinPolygonPointY11	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第11頂点のY座標値を定義
SearchWinPolygonPointX12	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第12頂点のX座標値を定義
SearchWinPolygonPointY12	SearchWinTypeが“Polygon”に設定されている時の、被検索領域の第12頂点のY座標値を定義
SearchWinRadiusInner	被検索領域の円内径を定義
SearchWinRadiusOuter	被検索領域の円外径を定義
SearchWinTop	被検索領域の最上端を定義(単位: ピクセル)
SearchWinType	被検査領域のタイプ (Rectangle, RotatedRectangle, Circle, Arc, Polygon)を定義
SearchWinWidth	被検索領域の幅を定義 (単位: ピクセル) デフォルト: 100

プロパティ	説明
SizeToFind	検出するブローブのサイズを選択 デフォルト: 1 - Largest
Sort	オブジェクトのリザルトを並び替える順番を選択 デフォルト: 0 – None
ThresholdAuto	対象物 (オブジェクト)、背景、画像イメージのエッジを表すグレイレベルのしきい値を自動設定するか否かを設定 デフォルト: 無効
ThresholdBlockSize	ThresholdMethodプロパティをLocalAdaptiveに設定したときに使用する、しきい値を決定するために近傍領域を参照する範囲を定義 デフォルト: 1/16ROI
ThresholdColor	しきい値の範囲内にあるピクセルの色を定義 デフォルト: 黒色
ThresholdHigh	ThresholdLowプロパティとともに、対象物 (オブジェクト)、背景、画像イメージのエッジを表すグレイレベル範囲を定義 ThresholdHighプロパティは、画像イメージの対象物領域のグレイレベル範囲の上限を定義 ThresholdLowとThresholdHighで定義されたグレイレベル範囲にあてはまる画像イメージは、どの部分も <u>ピクセルの重みづけ1</u> が割りあてられます。(つまり、対象物の一部分です。) ThresholdAuto が “True” で、かつ ThresholdColor が “White” の場合、本プロパティ値は255に設定され、変更できなくなります。 デフォルト: 128
ThresholdLevel	ThresholdMethodプロパティをLocalAdaptiveに設定したときに使用する、近傍領域との輝度差の割合を定義 デフォルト: 15%
ThresholdLow	ThresholdHighプロパティとともに、対象物 (オブジェクト)、背景、画像イメージのエッジを表すグレイレベル範囲を定義 ThresholdLowプロパティは、画像イメージの対象物領域のグレイレベル範囲の下限を定義 ThresholdLowとThresholdHighで定義されたグレイレベル範囲にあてはまる画像イメージは、どの部分も <u>ピクセルの重みづけ1</u> が割りあてられます。(つまり、対象物の一部分です。) ThresholdAuto が “True” で、かつ ThresholdColor が “Black” の場合、本プロパティ値は0に設定され、変更できなくなります。 デフォルト: 0

## 6. ビジョンオブジェクト

---

プロパティ	説明
ThresholdMethod	2値化の処理方法を設定

## Blobオブジェクトのリザルト

下の一覧は、Blobオブジェクトリザルトを簡単に説明した概略です。各リザルトの詳細は、Vision Guide 7.0プロパティ&リザルトリファレンスを参照してください。

リザルト	説明
Angle	検出されたパーツの回転量を角度で返す
Area	ブローブの領域をピクセルで返す
CameraX	カメラ座標系における検出パーツのX座標位置を返す
CameraY	カメラ座標系における検出パーツのY座標位置を返す
CameraXYU	実行時のみ カメラ座標系における検出パーツ位置のCameraX座標, CameraY座標, CameraU座標を返す
ClearanceOK	クリアランス (隙間)の判断結果を返す
Compactness	ブローブの緊密さを返す
Extrema	実行時のみ ブローブ極値のMinXピクセル座標, MaxXピクセル座標, MinYピクセル座標, MaxYピクセル座標を返す
Found	オブジェクトが検出されたかどうかを返す (つまり、MinAreaプロパティとMaxAreaプロパティの間に一致するAreaリザルトを持つ連結Blobです。)
FoundOnEdge	Blobオブジェクトがサーチウィンドウのエッジに近づきすぎる状態で検出されるとき、“True”を返す
Holes	ブローブで検出される穴の数を返す
MajorDiameter	検出されたブローブを楕円形に近似した場合の長径を返す
MaxFeretDiameter	検出されたブローブの最大フェレ径を返す
MaxX	ブローブの外接長方形の最大Xピクセル座標を返す
MaxY	ブローブの外接長方形の最大Yピクセル座標を返す
MinorDiameter	検出されたブローブを楕円形に近似した場合の短径を返す
MinX	ブローブの外接長方形の最小Xピクセル座標を返す
MinY	ブローブの外接長方形の最小Yピクセル座標を返す
NumberFound	サーチウィンドウ内に検出されたブローブの数を返す (最低0個から、最高はNumberToFindプロパティで設定したBlobオブジェクトによるブローブ検出個数になります。)
Passed	検出結果が良か否かを返す

リザルト	説明
Perimeter	検出されたブローブの、外側エッジの端から端までのピクセルの数
PixelX	検出パーツ位置のX座標をピクセルで返す
PixelY	検出パーツ位置のY座標をピクセルで返す
PixelXYU	実行時のみ 検出パーツ位置のPixelX座標, PixelY座標, PixelU座標をピクセルで返す
RobotX	ロボット座標系における検出パーツのX座標位置を返す
RobotY	ロボット座標系における検出パーツのY座標位置を返す
RobotU	ロボット座標系における検出パーツのU座標位置を返す
RobotXYU	実行時のみ ロボット座標系における、検出パーツ位置のRobotX座標, RobotY座標, RobotU座標を返す
Roughness	ブローブの粗さを返す
ShowAllResults	特定のビジョンオブジェクトのリザルトすべてを、表形式でダイアログに表示 リザルトの比較が簡単にできます。
Time	オブジェクト処理に要した時間を返す (単位: ミリ秒)
TotalArea	検出したリザルトすべての領域の合計を返す

## 2値検査の原理

2値検査のプロセスは、次の手順で行われます。

- 次の項目からなるセグメンテーション
  - しきい値
  - 連結性解析
- Blobリザルトの計算処理

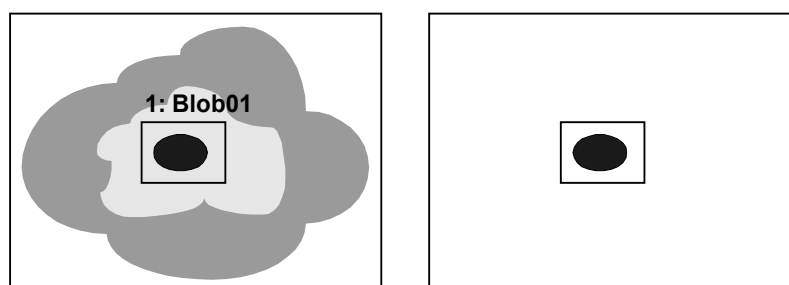
## セグメンテーション

2値検査では、オブジェクトの特徴を測定するため、まず画像イメージ中のどこにオブジェクトがあるのか、つまり、画像イメージ中の他のすべてのものからオブジェクトを区別しなければなりません。この、画像イメージのオブジェクトと背景との分割プロセスをセグメンテーションと呼びます。

Blobオブジェクトは、各ピクセルのグレースケール値に基づいてセグメンテーションできる画像イメージに使用されます。このような種類のセグメンテーションの簡単な例として、しきい値決定があります。

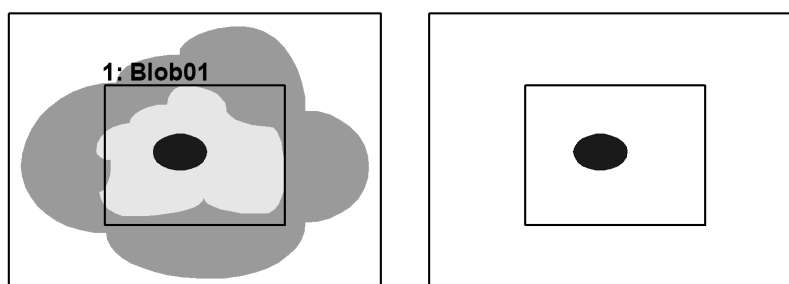
本章で説明するBlobオブジェクトは、グレースケール値で区別することができない、任意のグレースケールイメージについても正確なリザルトを出すことができます。しかし、そのような場合でも、通常は限られた使用価値しかありません。それは、リザルトが、画像を定義しているウィンドウのサイズに影響されるからです。

全体画像イメージの2値検査において、観察対象の特徴は、画像イメージ中で特定のグレイレベルを持つ唯一のオブジェクトでなければなりません。画像イメージ中に同じグレースケール値を持っている他のオブジェクトがあると、この画像イメージはセグメンテーションできません。以下の図A-Dは、それぞれ、全体画像イメージの2値検査のグレースケール値でセグメンテーションできる例とできない例を示します。



A: グレースケール値でセグメンテーションできるシーン

図Aは、カメラの視野を示します（左）。Blobオブジェクトを使って処理されるシーンは、“Blob01”という名称のサーチウィンドウ内に映しだされます。グレースケール値でセグメンテーションが完了すると、オブジェクトと背景は、図Aの右側の図に示すように容易に区別できるようになります。

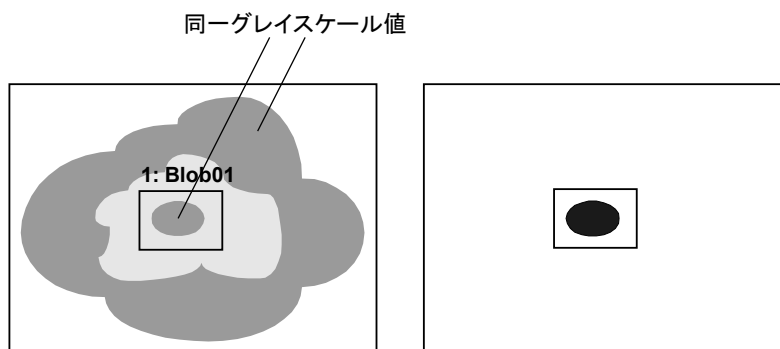


B: 拡大サーチウィンドウで見た図Aのシーン

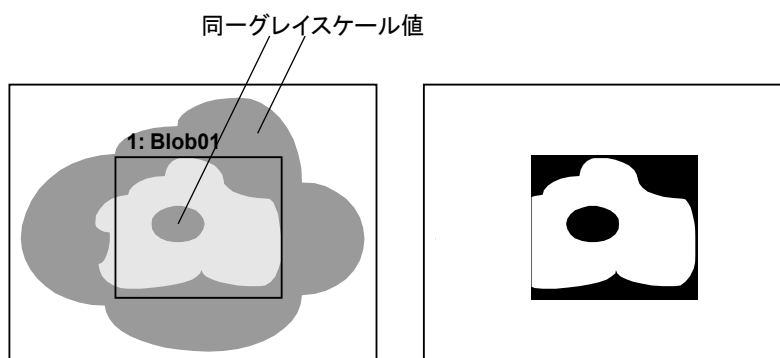
図Bに示すようにサーチウィンドウのサイズを変更すると、背景のサイズだけの変更されます。ブロープの特徴には、何の影響も及ぼしません。

図Cと図Dは、同様の画像イメージ視野を示しています。しかし、このシーンでは、画像イメージ中に同一のグレースケール値を持つ2つのオブジェクトがあります。

このシーンでは、背景領域とブロープの測定特徴の両方が、サーチウィンドウのサイズと、サーチウィンドウで囲まれた画像イメージ部分によって変化します。図C中の画像イメージがグレースケール値でセグメンテーションできたとしても、図Dに示すようにサーチウィンドウを拡大すると、セグメンテーションされた画像イメージは全く別なものに変化してしまいます。同一のグレースケール値を持つオブジェクトは、サーチウィンドウからどちらかを取り除かなければ、互いに区別することができません。

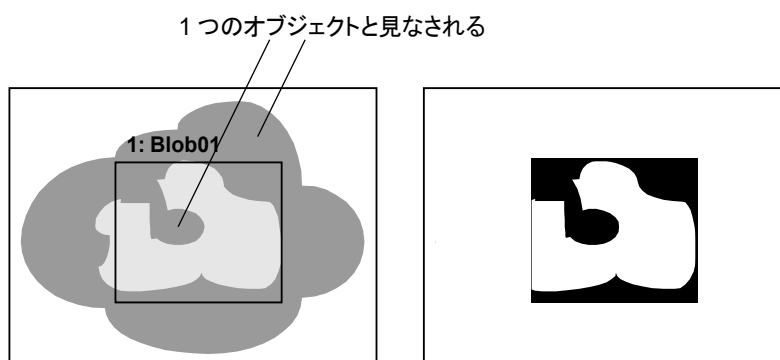


図C: グレースケール値でセグメンテーションできないシーン



図D: 同一のグレースケール値の2つのオブジェクトを囲む拡大サーチウィンドウ

また、図Eに示すような状況についても注意してください。この例では、内側のブローブと背景の一部が連結しています。そのため、セグメンテーションの対象となる中央のオリジナルのブローブとは非常に異なる大きなブローブが形成されてしまいます。



図E: オブジェクトと背景が連結している例



## しきい値

Blobオブジェクトは、画像中の各ピクセルの重みを決定するために、しきい値を使用します。

ユーザー定義のしきい値には、ThresholdLowとThresholdHighの2つがあります。グレースケールの値が2つのしきい値の範囲内にあるピクセルは、ピクセルの重みづけに“1”が与えられ、その他のピクセルには、“0”が与えられます。

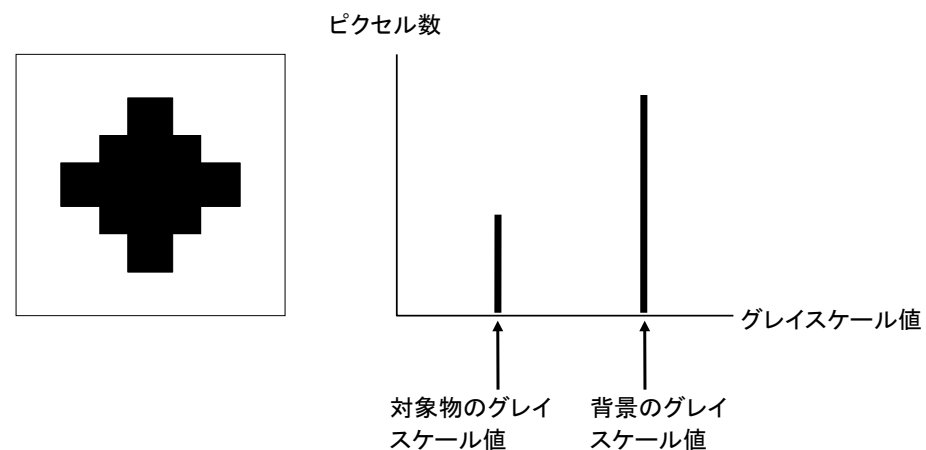
ThresholdColorプロパティは、重みづけ“1”のピクセルの色を定義します。このピクセルの色が、しきい値の範囲内の色 (黒色または白色)になります。

これらのピクセルの重みづけに基づいて、Blobオブジェクトは、画像を対象物(重みづけ“1”のピクセル部)と背景(重みづけ“0”のピクセル部)に分けます。Polarityプロパティでは、白色または黒色のピクセルを含むブローブを検出するように、Blobオブジェクトを設定します。Polarityの設定がDarkOnLightのときは、黒色のピクセルを含むブローブを検出します。一方、Polarityの設定がLightOnDarkのときは、白色のピクセルを含むブローブを検出します。

## ヒストグラムを使って、しきい値を決定する

Vision Guide 7.0 のヒストグラムツールを使って、しきい値のThresholdLowとThresholdHighの値を決定することができます。

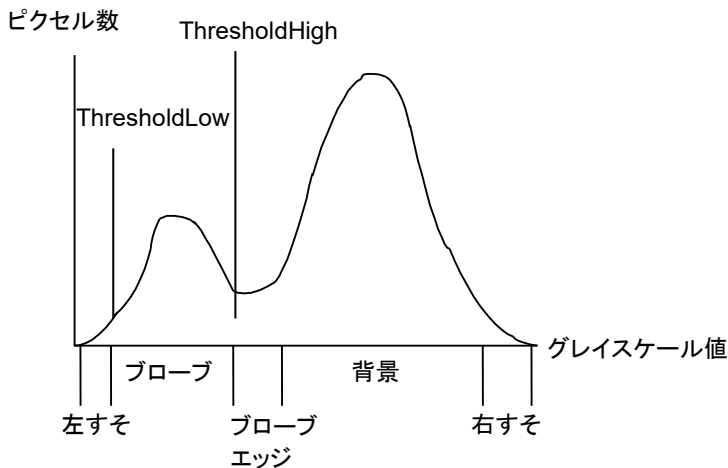
例として、白い背景上の黒いブローブの理想的なバイナリ画像イメージを考えてみましょう。下図に、そのような画像イメージとそのヒストグラムを示します。



### 理想的なバイナリ画像イメージとそのヒストグラム

このヒストグラムでは、0以外の値を持つグレースケール値が2つしかないことに注意してください。

実際の画像イメージでは、このようなヒストグラムを得るようなことは決してありません。さまざまなノイズ(例えば、不均一なプリント印刷、不規則な照明、電氣的なノイズ)の影響と結びついて、ピークは山型に広がります。より現実的なヒストグラムを下図に示します。



### ヒストグラムのピークを山形にするノイズの影響

上図に示すヒストグラムでは、各々のピークは、明瞭に示されています。ピークサイズ(領域)は、前に示した理想的なヒストグラムのピークサイズと同じ比率になっていますが、ここでは各ピークは山型に広がって、2つ以上のグレースケール値を持っています。

2つの主要なピークの間谷の部分のグレースケール値は、ブローブのエッジを現しており、完全に暗くも明るくもありません。

ブローブは重みづけ“1”のピクセルを持っているので、Threshold値を調整します。

### 連結性 (連結2値検査)

連結性は、非ゼロ質量の連結ピクセルに基づいた検査として定義することができます。もっと簡単に説明すると、連結性は、ブローブとしてみなされる連結ピクセルのグループを検出するのに使われます。

連結性は、Blobオブジェクトによって自動的に実行され、次に、検出されたブローブの測定が計算されます。Blobオブジェクトの連結性は、Blobオブジェクトの実行前に設定されたNumberToFindプロパティに基づいて、検出されたBlobオブジェクトの数を戻り値として戻します。

### Blobリザルトの計算

2値検査のその他のすべての手順が完了すると、検出されたブローブについてリザルトが計算されます。Blobオブジェクトのリザルトリストについては、「Blobオブジェクトのリザルト」の説明項目を参照してください。

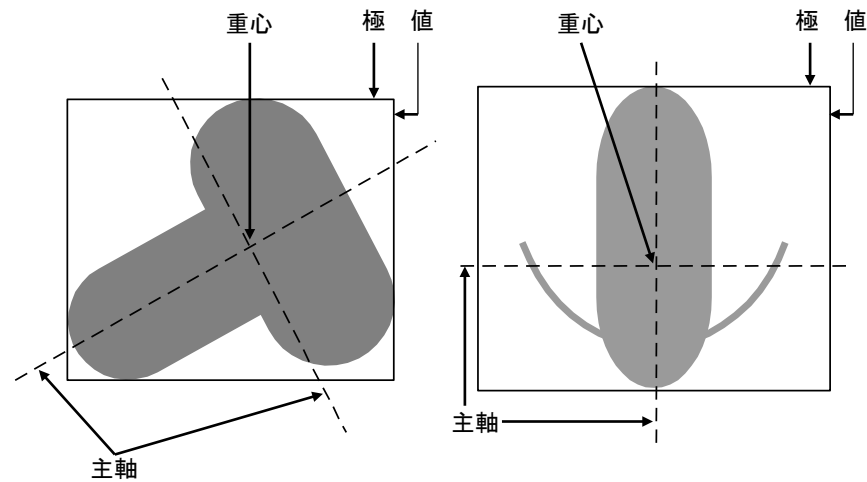
ビジョンオブジェクトのすべてのリザルトに関する詳細は、Vision Guide 7.0プロパティ&リザルトリファレンスを参照してください。

「Vision Guide 7.0 プロパティ&リザルトリファレンス」で説明しているリザルトのいくつか、例えばFoundリザルト、Timeリザルト、PixelXリザルトなどは、さまざまなビジョンオブジェクトで使われます。FoundリザルトやTimeリザルトは、極めて一般的なリザルトであり、すべてのビジョンオブジェクトで一般的に同じ方法で利用できますが、位置に関するリザルトな

どのように、いくつかのリザルトはBlobオブジェクトに使われるときに特別な意味を持ちます。これらのリザルトについて、以下に説明します。

#### MinX, MinY, MaxX, MaxY リザルト

MinXリザルト, MinYリザルト, MaxXリザルト, MaxYリザルトが結合すると、ブロープの外接長方形を作成します。外接長方形とは、ブロープを包囲する最小長方形のことです。外接長方形を理解していただくには、下図を参照してください。



主軸, 重心, 極値

#### ロボット座標系、カメラ座標系、ピクセル座標系での位置のデータ

座標位置のリザルトは、重心の位置を返します。重心は、必ずしもパーツの中心ではありませんので注意してください。

パーツをピックアップするときに重心を使うと、パーツによっては問題が生じることがあります。BlobオブジェクトからのRobotX, RobotY, RobotUの座標位置リザルトをパーツのピックアップ位置として使うときには、重心でパーツをピックアップすることが可能であることを確認してください。

重心でパーツをピックアップしたくないときは、オフセットを計算するか、Correlationオブジェクトなどの他のビジョンオブジェクトを使ってパーツを検出し、より便利なピックアップ位置を返すようにしなければなりません。

#### TotalArea リザルト

TotalAreaリザルトは、検出されたリザルトすべての領域の合計です。これは、ピクセルを計算するのに役立ちます。NumberToFindを0に設定することにより、Blobオブジェクトは、MinAreaとMaxAreaの間の領域をもつ、すべてのブロープを検出します。TotalAreaは、すべてのリザルトの合計領域を表示します。

#### Blob オブジェクトの Angle リザルト制限

BlobオブジェクトのAngleリザルトは、その範囲に制限がありますので注意してください。

BlobオブジェクトのAngleリザルトは、+90°から-90°までの範囲にある角度の値を返します。Blobオブジェクトは、360°全体の角度のリザルトを返すことはできません。




BlobオブジェクトのAngleリザルトは、必ずしも信頼できるリザルトを返すというわけではありませんので、ロボットを使用するときは注意してください。BlobオブジェクトのAngleリザルトの範囲には制限があり、場合によっては信頼できる値ではないことがあります。ロボットガイダンスのパーツの角度方向の計算にはBlobオブジェクトのAngleリザルトを使わずに、Polarオブジェクトを使うことをお勧めします。Polarオブジェクトは、Blobオブジェクトから重心として検出されたX, Y位置を使うことができ、Blobオブジェクトの重心に基づいた角度を計算します。詳細は、「6.2.6 Polarオブジェクト(ポーラサーチ)」で後述します。

### Blobオブジェクトの使用方法

以上、Vision Guide 7.0のBlobオブジェクトの使用方法を理解するための基本的知識として、2値検査がどのように動作するのか確認しました。ここでは、Blobオブジェクトの使用方法を説明します。

Blobオブジェクトを使うのに必要なステップは次のとおりです。


- 新規Blobオブジェクトの作成
- サーチウィンドウの位置とサイズの設定
- Blobオブジェクトの関連プロパティの設定
- Blobオブジェクトのテストとリザルトの確認
- プロパティの調整と再テスト

これらのステップを開始する前に、まず新規ビジョンシーケンスを作成するか、使用するビジョンシーケンスを選択してください。作業するビジョンシーケンスがない場合は、 <新規シーケンス>ボタンをクリックすると新規ビジョンシーケンスが作成できます。

すでにビジョンシーケンスを作成済みの場合は、Vision Guideウィンドウのシーケンスツリーをクリックして、作成済みのビジョンシーケンスを選択してください。

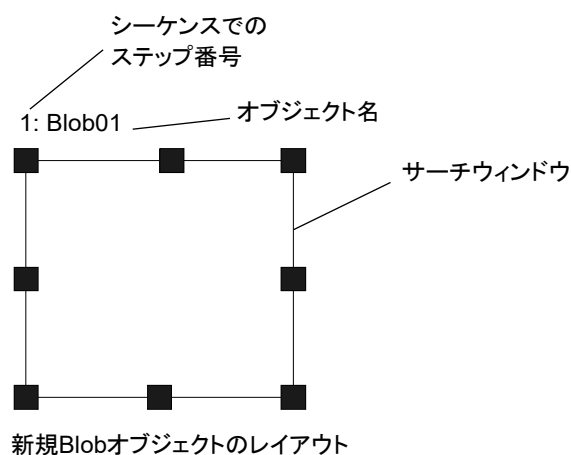
新規ビジョンシーケンスの作成方法、または作成済みビジョンシーケンスの選択方法の詳細は、「5. ビジョンシーケンス」を参照してください。

### Step 1: 新規Blobオブジェクトの作成

- (1) Vision Guideツールバー - <全ツール> -  <Blob (2値検査)>ボタンをクリックしてください。
- (2) 画像イメージ表示部にマウスを動かしてください。マウスポインターがBlobアイコンに変わります。
- (3) アイコンが画像イメージ表示部の目的の位置に来るまで、マウスを動かし続けてください。それから、マウスを左クリックして、オブジェクトを作成してください。
- (4) オブジェクト名は自動的に作成されますので注意してください。この例では、このシーケンスで最初に作成されたBlobオブジェクトなので、“Blob01”という名称になります。(名称の変更方法については後述します。)

## Step 2: サーチウィンドウの位置とサイズの設定

画面上には、下図のようなBlobオブジェクトが表示されます。



- (1) Blobオブジェクトの名称ラベルをクリックし、マウスを押し続けながら、Blobオブジェクトをドラッグして、サーチウィンドウの左上コーナーを好きな位置に合わせてください。
- (2) サイズハンドルをクリックして、マウスをドラッグすると、Blobオブジェクトサーチウィンドウのサイズが変更できますので、適当な大きさに調整してください。(サーチウィンドウは、ブローブの被検索領域になります。)



注意

- ビジョンシーケンスの実行結果は、外らん光や外部機器ノイズにより影響を受けることがあります。  
外らん光や外部機器ノイズにより、影響を受けると、取り込み画像が想定外の結果となり、検出位置結果がサーチエリア (検出エリア)内の不特定位置になる可能性があります。  
サーチエリアは、極力小さく、また誤検出に配慮した画像処理シーケンスの作成を行ってください。

## Step 3: Blobオブジェクトプロパティの設定

このステップでは、Blobオブジェクトのプロパティを設定することができます。

次に、Blobオブジェクトに特有で、よく使用されるプロパティを示します。

その他、さまざまなビジョンオブジェクトの多くで使われるAbortSeqOnFail, Graphicsなどのプロパティの詳細は、Vision Guide 7.0プロパティ&リザルトリファレンス、または、「Blobオブジェクトのプロパティ」リストを参照してください。



注意

- ビジョンシーケンスの実行結果は、外らん光や外部機器ノイズにより影響を受けることがあります。  
外らん光や外部機器ノイズにより、影響を受けると、取り込み画像が想定外の結果となり、検出位置結果がサーチエリア (検出エリア)内の不特定位置になる可能性があります。  
MaxArea, MinArea, RejectOnEdgeなどのプロパティ値を適切に設定し、誤検出リスクを軽減してください。

- Nameプロパティ**      新規に作成されたBlobオブジェクトにデフォルトとして与えられる名称は、“Blobxx”です。xxは、1つのビジョンシーケンス内で複数のBlobオブジェクトを用いるとき、それらのBlobオブジェクトを区別するための番号です。  
ビジョンシーケンスで1番目に用いられるBlobオブジェクトには、“Blob01”という名称がデフォルトで設定されます。  
名称を変更したい場合は、Nameプロパティの設定値フィールドをクリックして、変更後の名称を入力し、リターンキーを押してください。画面上でそのBlobオブジェクト名を表示する場所の全部が、変更後の名称を表示するようになります。
- Polarityプロパティ**      明るい背景の暗いオブジェクトを検出したい場合と、暗い背景の明るいオブジェクトを検出したい場合があります。この選択をするのが、Polarityプロパティの基本的な目的です。  
デフォルトは、DarkOnLight (明るい背景の暗いオブジェクト)の検出に設定されています。  
このデフォルト設定を変更したいときは、Polarityプロパティの値フィールドをクリックしてください。ドロップダウンリストが表示され、“Dark on Light”と“Light on Dark”の選択項目が表示されます。選択して、クリックしてください。
- MinArea, MaxArea**      “Found”とみなされる (Foundリザルトの値が“True”だった)Blobオブジェクトの面積を定義  
デフォルトの範囲は、25 ~ 100,000 (MinArea ~ MaxArea)と非常に広く設定されています。つまり、MinAreaプロパティとMaxAreaプロパティを調整しないまま、新規作成のBlobオブジェクトを実行すると、ほとんどのブローブは検出されたとして報告されます。通常は、検出対象のブローブに適切な範囲を反映するために、これらのプロパティを調整します。このようにして、範囲の外側に何らかのブローブが検出された場合は、それは検出対象のブローブではないと判断することができます。

**RejectOnEdge プロパティ** このプロパティによって、サーチウィンドウのエッジにかかって検出されたパーツを除外することができます。通常はこのプロパティを“True”に設定します。

以上で、Blobオブジェクトのテストができるようになりました。必要となるその他のプロパティは、テスト後に再び設定することになります。

#### Step 4: Blobオブジェクトのテストとリザルトの確認

Blobオブジェクトのテストを実行するには、実行パネルのオブジェクトの<Run>ボタンをクリックしてください。Blobオブジェクトのリザルトが表示されます。ここで確認する主要なリザルトは次に示すとおりです。けれども、役に立つリザルトがこれから他にも出てきます。

<b>Foundリザルト</b>	オブジェクトが検出されたかどうかの値を返す  検出されたブローブがMinAreaプロパティとMaxAreaプロパティで定義された面積の範囲内にはないときは、Foundリザルトは“False”を返します。
<b>Areaリザルト</b>	検出されたブローブの面積 (単位: ピクセル)
<b>Angleリザルト</b>	ブローブの方向角度  副軸の角度から計算し、90° ~ 90°までの範囲の値になります。
<b>Timeリザルト</b>	Blobオブジェクトの実行に要した時間
<b>PixelX, PixelY</b>	検出パーツの重心のXY座標位置 (単位: ピクセル)
<b>MinX, MinY, MaxX, MaxY</b>	これらの4つの値で、ブローブの外接長方形を定義します。 (ブローブの外接点に接触して形成される長方形)

#### NOTE



RobotXYU, RobotX, RobotY, RobotUのリザルトとCameraX, CameraY, CameraXYUのリザルトは、この時点で“no cal”を返します。これは、キャリブレーションが実行されていないので、ロボット座標系あるいはカメラ座標系に関して座標リザルトを計算することができなかった、ということを意味しています。詳細は、「7. ビジョンキャリブレーション」を参照してください。

#### Step 5: プロパティの調整と再テスト

Blobオブジェクトを数回実行することにより、ブローブ検出上の問題が見つかったり、プロパティ設定の微調整をする必要が出てくることもあるかもしれません。一般的に生じる問題や、微調整の方法について次に説明します。

**問題:** BlobオブジェクトのFoundリザルトがFalseを返すときは、次の事項を確認し、調整してください。

- 暗い背景での明るいオブジェクトを検出するのか、明るい背景での暗いオブジェクトを検出するのかを定義するPolarityプロパティの値が、実際と異なっているかもしれません。Polarityプロパティの値を確認し、検出したいオブジェクトと背景との明暗に一致させてください。また、サーチウィンドウ内に表示されるオブジェクトと背景との明暗にも一致していなければなりません。

- Areaリザルトを確認し、この値とMinAreaプロパティとMaxAreaプロパティで定義された値とを比較してください。Areaリザルトが、MinAreaプロパティとMaxAreaプロパティで定義された範囲に入っていない場合は、MinAreaプロパティとMaxAreaプロパティを調整して、Blobオブジェクトを再実行してください。
- ヒストグラムを使って、画像イメージ中のグレイスケール値の分布を確認してください。ヒストグラムの詳細は、「8. ヒストグラムツール」を参照してください。


**微調整:** アプリケーションによっては、Blobオブジェクトの微調整が必要となる場合があります。Blobオブジェクトの微調整に関する主要なプロパティについて、次に説明します。


- MinArea, MaxArea – Blobオブジェクトを数回実行してみると、Areaリザルトに返される概略値がわかってきます。MinAreaプロパティとMaxAreaプロパティの値を変更するときは、これらの概略値を使ってください。一般的に、MinAreaプロパティとMaxAreaプロパティの値は、Foundリザルトを制限するように設定し、検出対象のブローブだけについてTrueのFoundリザルトが返るようにすると好都合です。(このように設定することによって、検出したいブローブと面積が異なっているブローブを排除することができます。)
- ThresholdHigh, ThresholdLow – これらのプロパティは、背景とブローブの一部を識別するために、グレイレベルのしきい値を設定するパラメーターを調整します。これらのプロパティは、ヒストグラムツールを使って設定するのが、最も適切です。Vision Guide 7.0プロパティ&リザルトリファレンス「ThresholdHigh Property」と「ThresholdLow Property」を参照してください。ヒストグラムについては、「8. ヒストグラムツール」に詳細を説明しています。

プロパティの調整が完了し、Blobオブジェクトのテストを行って満足なリザルトを得ることができたら、ビジョンオブジェクトの作成ステップは完了です。

他のビジョンオブジェクトを作成したり、ビジョンシーケンス全体の設定やテストを行う次のステップに進むことができます。

### Blobオブジェクトに関するその他の便利なユーティリティ

この時点で、Vision Guide 7.0のヒストグラム機能を確認することもできます。 <ヒストグラム>ボタンは、サーチウィンドウ内のグレイスケール値の分布をグラフで表示できるので、非常に便利です。Vision Guideヒストグラムツールは、ブローブの一部であるものと背景の一部であるものを定義する、ThresholdLowプロパティとThresholdHighプロパティのグレイレベルを設定するのに便利な構造になっています。ブローブの検出に問題があるときは、ヒストグラム機能が有効です。Vision Guideヒストグラムの使用方法の詳細は、「8.1 ヒストグラムの使用方法」に説明があります。

また、Blobオブジェクトのリザルトを統計的に確認するのに、 <統計>ボタンを使用することもできます。Vision Guide統計機能の詳細は、「9. 統計ツール」を参照してください。



### ピクセル計数器としてのBlobオブジェクトの使い方

Blobオブジェクトはピクセル計数器として使用できます。ピクセル計数器は、ブローブしきい値の範囲にある画像イメージのピクセルすべてをカウントします。

次のステップにしたがってください。

- (1) Blobオブジェクトを作成してください。
- (2) 極性を設定してください。
- (3) Highしきい値とLowしきい値を設定してください。
- (4) NumberToFindを“0”に設定してください。これで、Blobオブジェクトが画像イメージのブローブすべてを検出ようになります。
- (5) MinAreaを“1”に、MaxAreaを“999999”に設定してください。1ピクセル以上のブローブをカウントします。
- (6) シーケンスを実行してください。

TotalAreaリザルトを使って、Blobしきい値の範囲にあるピクセルの合計数を読み取ってください。

#### 6.2.5



#### Edge オブジェクト (エッジ検出)

##### Edgeオブジェクトの概要

Edgeオブジェクトは、画像イメージ中のエッジの場所を特定するのに使われます。

画像イメージ中のオブジェクトのエッジは、グレイスケール値の暗から明への変化、あるいは明から暗への変化で表されます。このような変化は、数個のピクセルにわたって観察されます。

Edgeオブジェクトでは、Polarityプロパティーによって定義された、明から暗への変化、または暗から明への変化を検出し、検出された位置をシングルエッジのエッジ位置として定義します。EdgeTypeプロパティーを変更してエッジペアを検出することもできます。エッジペアの場合、2つの対極のエッジを検索し、中間点を返します。Edgeオブジェクトは複数のリザルトに対応するため、検出するシングルエッジ、およびエッジペアの数を指定できます。

Edge オブジェクトは、SearchType プロパティーを使って、ラインや円弧に沿って検索するように設定できます。

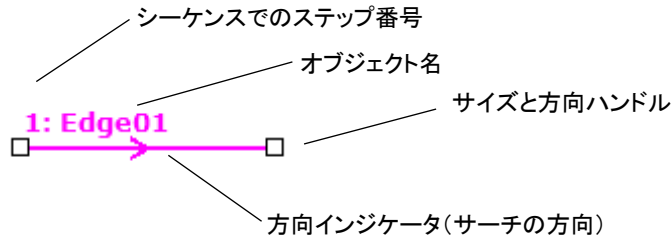
SearchTypeがLineのEdgeオブジェクトは、検索長さをもつ形状が、Lineオブジェクトに似ています。検索長さとは、Edgeオブジェクトの長さのことです。SearchTypeがLineのEdgeオブジェクトの強力な特徴の1つは、どんな角度でも置けることです。これで、Edgeオブジェクトの角度を変えることにより、エッジを検出したい領域に垂直なEdgeオブジェクトベクトルを維持することができます。通常、作業したい領域とともに動くFrameにEdgeオブジェクトを関連させることにより、これができます。

### Edgeオブジェクトのレイアウト

Edge オブジェクトには2つのレイアウトがあります。

#### SearchType がLineの場合

SearchTypeがLineの場合、Edgeオブジェクトのサーチウィンドウは、Edgeオブジェクトに沿ったラインに相当します。Edgeオブジェクトは、方向インジケータが示す方向のこのラインに沿って、変化(明から暗へ、または暗から明へ)を検索します。

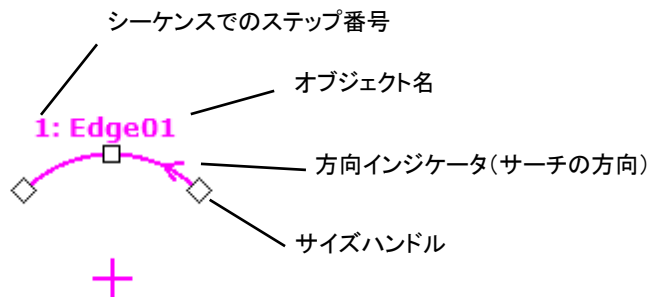


#### Edge オブジェクト(ライン)のレイアウト

Edgeオブジェクトは、(垂直方向、水平方向だけでなく)あらゆる方向を検索するように置くことができます。これは、Edgeオブジェクトのサイズと方向ハンドルを使って、Edgeオブジェクトのいずれかの終点を目的のエッジを検出する方向 (およびユーザー指定間隔ごと)に動かすことでできます。オブジェクト全体を移動するには、ラベルかラインをドラッグします。

#### SearchType が Arc の場合

SearchType が Arc の場合、Edge オブジェクトのサーチウィンドウは Edge オブジェクトに沿った円弧に相当します。Edge オブジェクトは、方向インジケータが示す方向のこの円弧に沿って、変化 (明から暗へ、または暗から明へ)を検索します。



#### Edge オブジェクト(円弧)のレイアウト

円弧のサイズを変更するには、終点にあるサイズハンドルのいずれかをドラッグします。半径を変更するには、真ん中のサイズハンドルをドラッグします。オブジェクト全体を移動するには、ラベルか中心点をドラッグします。

### Edgeオブジェクトのプロパティ

下の一覧は、Edgeオブジェクトプロパティの概略です。各プロパティの詳細は、Vision Guide 7.0プロパティ & リザルトリファレンスを参照してください。

SearchTypeプロパティを変更すると、プロパティは指定したタイプに従って変更を設定します。SearchTypeがLineの場合、以下のプロパティはVision Guideウィンドウのプロパティグリッドには表示されません。

AngleEnd  
AngleStart  
CenterPointObject  
CenterPntObjResult  
CenterPntOffsetX  
CenterPntOffsetY  
CenterPntRotOffset

SearchTypeがArcの場合、以下のプロパティは表示されません。

X1  
Y1  
X2  
Y2

プロパティ	説明
AbortSeqOnFail	オブジェクトの実行に失敗した (不良判定になった) とき、直ちにシーケンス全体を中断し、それ以降のオブジェクトを処理しないように指定 デフォルト: False
Accept	特徴が検出されたと判定するための形状スコアの値を指定 対象物が検出されたとみなされるためには、スコアの値を超えてはなりません。 デフォルト: 100
AngleEnd	円弧/楕円弧をサーチする範囲の終了角度を指定 デフォルト: 135
AngleStart	円弧/楕円弧をサーチする範囲の開始角度を指定 デフォルト: 45
CalRobotPlacePos	設計時とプログラム実行時に RobotPlacePos をキャリブレーションする
Caption	見出しをEdgeオブジェクトに割りあてる デフォルト: 空欄

プロパティ	説明
CenterPointObject	<p>オブジェクトの中心点として用いられる位置を指定</p> <p>このプロパティを“Screen”に設定すると画面上の任意の位置にオブジェクトを配置できますが、他のビジョンオブジェクトを指定した場合、そのオブジェクトのPixelX, PixelYリザルトに中心点が設定されます。</p> <p>デフォルト: Screen</p>
CenterPntObjResult	<p>CenterPointObjectプロパティからどのリザルトを使うのかを指定</p> <p>Allを指定すると、指定されたビジョンオブジェクトリザルトのすべて (NumberFound)にEdgeオブジェクトが適用されます。</p> <p>デフォルト: 1</p>
CenterPntOffsetX	<p>サーチウィンドウの中心をCenterPointObjectプロパティで位置決めしたあとに、Xオフセットを設定、または返す</p> <p>デフォルト: 0</p>
CenterPntOffsetY	<p>サーチウィンドウの中心をCenterPointObjectプロパティで位置決めしたあとに、Yオフセットを設定、または返す</p> <p>デフォルト: 0</p>
CenterPntRotOffset	<p>中心点の XY オフセット値 (CenterPntOffsetX, CenterPntOffsetY)を、CenterPointObject の Angle リザルトにしたがって回転するか否かを指定</p> <p>SearchWinType が RotatedRectangle に設定されている場合、サーチウィンドウは Angle リザルトにしたがって回転します。</p> <p>デフォルト: False</p>
CheckClearanceFor	クリアランス (隙間)を確認するオブジェクトを設定
ClearanceCondition	クリアランス (隙間)の判断方法を指定
ContrastTarget	<p>エッジサーチのためのコントラストを設定</p> <p>デフォルト: 0 (best contrast)</p>
ContrastVariation	<p>ContrastTargetの許容されるコントラストの変化量を選択</p> <p>デフォルト: 0</p>
CoordObject	<p>リザルトをコピーするCoordinatesオブジェクトを指定</p> <p>コピー処理はオブジェクトが実行される時に行われ、Decisionの分岐機能などで実行されなかった場合、コピー処理は行われません</p> <p>デフォルト: None</p>
CurrentResult	<p>オブジェクトウィンドウのリザルトリストに表示するリザルト、または、1つのサーチウィンドウ内にある対象物などの複数のリザルトを、システムが検出する設定のときにデータを返すリザルトを定義</p>
Description	<p>ユーザー指定の説明を設定</p> <p>デフォルト: 空欄</p>
EdgeSort	検出されたエッジ結果のソート方法を設定

プロパティ	説明
EdgeThreshold	この値より低いエッジは無視するという、しきい値を設定 デフォルト: 2
EdgeType	検索するエッジタイプ (シングル、またはペア)を設定 デフォルト: 1 - シングル
Enabled	このオブジェクトを実行するかを指定 デフォルト: True
EndPntObjResult	どのリザルトを EndPointObject から用いるのかを特定
EndPointObject	どのビジョンオブジェクトを使って、直線の終点を定義するかを特定
EndPointType	どのタイプの終点を使って直線の終点を定義するかを特定
FailColor	オブジェクトの検出不良時の色を選択 デフォルト: Red
Frame	どの位置に置いたフレームを使用するかを指定 デフォルト: none
FrameResult	Frameの何番目のリザルトを使用するかを指定 デフォルト: 1
Graphics	表示するグラフィックを指定 デフォルト: 1 - All
LabelBackColor	オブジェクトラベルの背景の色を設定 デフォルト: Transparent (透明)
Name	Edgeオブジェクトに独自のオブジェクト名を割りあてる デフォルト: Edge01
NumberToFind	検出するエッジ数を設定 デフォルト: 1
PassColor	オブジェクトの検出良時の色を選択 デフォルト: LightGreen
PassType	オブジェクト検出時の良否判定の条件を定義 デフォルト: SomeFound
Polarity	Edgeオブジェクトが明から暗への移行を検索するのか、暗から明への移行を検索するかを指定 デフォルト: 1 - LightToDark
Radius	オブジェクトの中心点から、オブジェクトの検索リング外側までの距離を定義
SearchType	直線と円弧検索のどちらを使用するかを設定 デフォルト: Line
ScoreWeightContrast	コントラストによるスコアの割合を設定 デフォルト: 50
ScoreWeightStrength	エッジの強度によるスコアの割合を設定 デフォルト: 50

プロパティ	説明
SearchWidth	エッジサーチの幅を定義 範囲: 3 ~ 99 デフォルト: 3
StartPntObjResult	どのリザルトをStartPointObjectから用いるのかを指定
StartPointObject	どのビジョンオブジェクトを使って、直線の始点を定義するかを指定
StartPointType	どのタイプの始点を使って直線の始点を定義するかを指定
StrengthTarget	サーチするエッジの強度を設定 デフォルト: 0
StrengthVariation	StrengthTargetの変化量の合計を設定 デフォルト: 0
X1	エッジの始点位置のX座標
X2	エッジの終点位置のX座標
Y1	エッジの始点位置のY座標
Y2	エッジの終点位置のY座標

## Edgeオブジェクトのリザルト

下の一覧は、Edgeオブジェクトリザルトを簡単に説明した概略です。各リザルトの詳細は、Vision Guide 7.0プロパティ&リザルトリファレンスを参照してください。


リザルト	説明
CameraX	カメラ座標系における検出エッジ位置のX座標を返す
CameraY	カメラ座標系における検出エッジ位置のY座標を返す
CameraXYU	実行時のみ カメラ座標系における検出エッジ位置のCameraX座標, CameraY座標, CameraU座標を返す
ClearanceOK	クリアランス (隙間)の判断結果を返す
Contrast	検出エッジのコントラストを返す
Found	オブジェクトが検出されたかどうか、対象物あるいはパーツがAcceptプロパティの現在の設定を超える形状スコア値を持っているかどうかの値を返す
NumberFound	検出されたエッジ数を返す (最小0個から、最大はNumberToFindプロパティで設定した検出個数になります。)
Passed	検出結果が良か否かを返す
PixelX	検出Edge位置のX座標をピクセルで返す
PixelY	検出Edge位置のY座標をピクセルで返す
PixelXYU	実行時のみ 検出Edge位置のPixelX座標, PixelY座標, PixelU座標をピクセルで返す
RobotX	ロボット座標系における検出エッジ位置のX座標を返す
RobotY	ロボット座標系における検出エッジ位置のY座標を返す
RobotXYU	実行時のみ ロボット座標系における検出Edge位置のRobotX座標, RobotY座標, RobotU座標を返す
Score	検出エッジの全般的なスコアを示すINTEGER値を返す
Strength	検出エッジの強度を返す
Time	オブジェクト処理に要した時間を返す (単位: ミリ秒)

### Edgeオブジェクトの使用方法

ここからは、Edgeオブジェクトの作成と使用の方法を説明します。

- 新規Edgeオブジェクトの作成方法
- サーチウィンドウの位置とサイズの設定
- Edgeオブジェクトの関連プロパティの設定
- Edgeオブジェクトのテストとリザルトの確認
- プロパティの調整と再テスト


これらのステップを開始する前に、まず新規ビジョンシーケンスを作成するか、使用する

ビジョンシーケンスを選択してください。作業するビジョンシーケンスがない場合は、 <新規シーケンス>ボタンをクリックすると、新規ビジョンシーケンスが作成できます。

すでにビジョンシーケンスを作成済みの場合は、Vision Guideウィンドウのシーケンスツリーをクリックして、作成済みのビジョンシーケンスを選択してください。

新規ビジョンシーケンスの作成方法、または作成済みビジョンシーケンスの選択方法の詳細は、「5. ビジョンシーケンス」を参照してください。

#### Step 1: 新規Edgeオブジェクトの作成

- (1) Vision Guideツールバー - <全ツール> -  <Edge (エッジ検出)>ボタンをクリックしてください。
- (2) 画像イメージ表示部にマウスを動かしてください。マウスポインターがEdgeオブジェクトアイコンに変わります。
- (3) アイコンが画像イメージ表示部の目的の位置に来るまで、マウスを動かし続けてください。それから、マウスを左クリックして、オブジェクトを作成してください。
- (4) オブジェクト名が自動的に作成されるのに注目してください。この例では、このシーケンスで最初に作成されたEdgeオブジェクトなので、"Edge01"というオブジェクト名になります。(オブジェクト名の変更方法については後述します。)



### Step 2: Edgeオブジェクトの位置決め

画面上には、下図のようなEdgeオブジェクトが表示されています。



#### 新規Edgeオブジェクト

新規Edgeオブジェクトを作成した後、ラインと円弧のどちらに沿って検索するかをSearchTypeプロパティで設定できます。SearchTypeがLine (デフォルト)の場合、サイズハンドルのどちらかをクリックすることや、ラインの端を新しい位置にドラッグすることで、検索長さや回転を変更できます。SearchTypeがArcの場合、サイズハンドルのどちらかをドラッグすることで円弧を変更できます。半径を変更するには、真ん中のハンドルをドラッグします。

また、Edgeオブジェクトの名称ラベルやエッジラインの周囲をクリックして、マウスを押し続けながら、Edgeオブジェクト全体を画面上の新しい位置にドラッグしてください。位置が決まったら、マウスを離してください。Edgeオブジェクトが画面上の新しい位置にきます。

### Step 3: Edgeオブジェクトプロパティの設定

Edgeオブジェクトプロパティの設定を行います。プロパティを設定するには、関連するプロパティの設定値フィールドをクリックし、新規の値を入力してください。あるいは、ドロップダウンリストが表示されている場合は、リスト中の該当項目をクリックしてください。

次に、Edgeオブジェクトで一般的に使われるプロパティのいくつかを示します。その他、さまざまなビジョンオブジェクトの多くで使われるAbortSeqOnFail, Graphicsなどのプロパティの詳細は、Vision Guide 7.0プロパティ&リザルトリファレンスを参照してください。

EdgeType(シングル)	<p>検索するエッジの種類を選択</p> <p>エッジペアを設定した場合、それぞれの方向のエッジを検索し、中間点を位置として検出します。</p>
Nameプロパティ ("Edgexx")	<p>新規に作成されたEdgeオブジェクトに、デフォルトとして与えられるオブジェクト名は "Edgexx" です。xxは、1つのビジョンシーケンス内で複数のEdgeオブジェクトを用いるとき、それらのEdgeオブジェクトで区別するための番号です。ビジョンシーケンスで1番目に用いられるEdgeオブジェクトには、"Edge01" というオブジェクト名がデフォルトで設定されます。オブジェクト名を変更したい場合は、Nameプロパティの設定値フィールドをクリックして、変更後の名称を入力し、リターンキーを押してください。画面上でEdgeオブジェクト名を表示する場所は、すべて変更後の名称に変わります。</p>
NumberToFind(1)	<p>エッジ検索ラインに沿って1つ以上のエッジを検索します。</p>
Polarity (LightToDark)	<p>DarkToLight (暗から明へ)のエッジを検索する場合は、Polarityを変更してください。</p>

### Step 4: Edgeオブジェクトの実行とリザルトの確認

Edgeオブジェクトを実行するには、次のようにしてください。

実行パネルのオブジェクトの<Run>ボタンをクリックしてください。Edgeオブジェクトのリザルトが表示されます。この時点で確認すべき主要なリザルトは次のとおりです。

PixelXリザルト	エッジ検索ラインに沿って検出されたエッジのXY座標位置
PixelYリザルト	(ピクセル単位)
CameraXリザルト	カメラ座標系でEdgeオブジェクトのXY座標を定義
CameraYリザルト	CameraXリザルトとCameraYリザルトは、カメラがキャリブレーションされている場合は値を返すだけです。もしキャリブレーションされていない場合は、“no cal”が返されます。
RobotXリザルト	ロボット座標系でEdgeオブジェクトのXY座標を定義
RobotYリザルト	ロボットにこのXY座標に動くよう命令できます。(他の変換や手順は必要ありません。) RobotXリザルトとRobotYリザルトは、カメラがキャリブレーションされている場合は値を返すだけです。もしキャリブレーションされていない場合は、“no cal”が表示されます。

### 6.2.6 Polarオブジェクト (ポーラサーチ)

#### Polarオブジェクトの概要

Polarオブジェクトでは、デカルト座標を持つ画像イメージを、それに対応する極座標を持つ画像イメージに高速に変換することができます。Polarオブジェクトは、オブジェクトの回転を判定するのに優れたツールです。Polarオブジェクトは信頼性が高く、高速なスピードで実行できるので、オブジェクトの角度回転の計算が必要な場合にご活用ください。

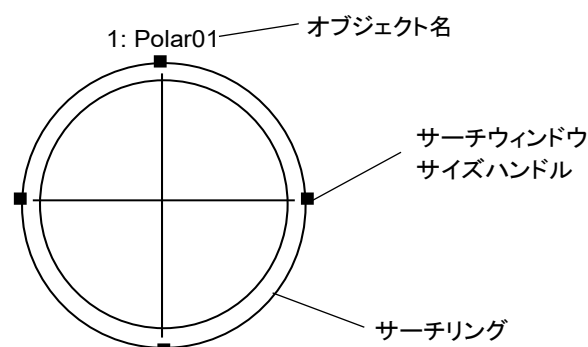
Polarオブジェクトは、回転を判定するツールであるため、オブジェクトの中心回りの回転は計算精度に影響は出ませんが、オブジェクトのx-y平面の平行移動は計算精度の大きく影響するという特性を持っています。つまり、Polarオブジェクトは、オブジェクトの回転の計算を非常に得意としています。そのためにも、オブジェクトの中心位置が動かないことが第一条件です。

このため、Vision Guide 7.0では、PolarオブジェクトにCenterPointプロパティを装備しています。CenterPointプロパティでは、Polarオブジェクトの中心点を、CorrelationオブジェクトやBlobオブジェクトなど、他のオブジェクトの位置リザルトにロックすることができます。

つまり、最初にCorrelationオブジェクトやBlobオブジェクトなどでXY位置を検出すると、PolarオブジェクトはそのXY位置を中心点としてオブジェクトの回転を計算することができます。

## Polarオブジェクトのレイアウト

Polarオブジェクトのレイアウトの外観は、これまで説明してきた他のオブジェクトのレイアウトと全く異なっています。しかし、用途は非常に似ています。Polarオブジェクトは円形を基本とするレイアウトで、中点と半径が表示されます。Polarオブジェクトの位置は、Polarオブジェクト名 (または、外側の周囲を構成する円上のどこか)をクリックし、新規の位置にオブジェクトをドラッグすることによって、移動することができます。



### Polar オブジェクトのレイアウト

#### NOTE



Polarオブジェクトの中点 (CenterPointプロパティによって定義される)は、他のオブジェクトの位置を基準として設定することができます。つまり、Polarオブジェクトの位置を調整した場合でも、オブジェクトやシーケンスを実行することによって、Polarオブジェクトの中点が変わることがあります。Polarオブジェクトの詳細は、本章で後述を参照してください。

Polarオブジェクトのサーチウィンドウは円形です。外側の境界線は、上図に示すような外側のサーチリングによって定義されます。内側のリングは、外側のリングより小さく、外側リングのnピクセル内側にあります。ピクセル数は、Thicknessプロパティによって定義されます。Thicknessプロパティを変更すると、“Thickness”、または内側と外側のリングの距離が変わるのがわかります。これは、検索している領域を目に見える表示にしているからです。

Polarオブジェクトサーチウィンドウの外側境界線のサイズを調節するには、4つのサーチウィンドウサイズハンドルの内の1つをクリックし、リングを任意の大きさになるまで内側あるいは外側にドラッグしてください。

## Polarオブジェクトのプロパティ

下の一覧は、Polarオブジェクトプロパティの概略です。各プロパティの詳細は、Vision Guide 7.0プロパティ&リザルトリファレンスを参照してください。

プロパティ	説明
AbortSeqOnFail	オブジェクトの実行に失敗した (不良判定になった)とき、直ちにシーケンス全体を中断し、それ以降のオブジェクトを処理しないように指定 デフォルト: False
Accept	対象物が検出されたと判定するための形状スコアの値を指定 対象物のスコアが設定値を超えているものを検出します。設定値を低く設定すると、誤検出の原因となります。 デフォルト: 700
AngleOffset	Polarオブジェクトの登録が完了した後、検索の対象物の角度インジケータグラフィックを調整し、Polarオブジェクトに適用されるオフセット角度を定義 デフォルト: 0
CalRobotPlacePos	設計時とプログラム実行時に RobotPlacePos をキャリブレーションする
Caption	見出しをPolarオブジェクトに割りあてる デフォルト: 空欄
CenterPointObject	オブジェクトの中心点として用いられる位置を指定 このプロパティを“Screen”に設定すると画面上の任意の位置にオブジェクトを配置できますが、他のビジョンオブジェクトを指定した場合、そのオブジェクトのPixelX, PixelYリザルトに中心点が設定されます。 デフォルト: Screen
CenterPntObjResult	CenterPointObjectプロパティからどのリザルトを使うのかを指定 Allを指定すると、指定されたビジョンオブジェクトリザルトのすべて (NumberFound)にPolarオブジェクトが適用されます。 デフォルト: 1
CenterPntOffsetX	サーチウィンドウの中心をCenterPointObjectプロパティで位置決めしたあとに、Xオフセットを設定、または返す デフォルト: 0
CenterPntOffsetY	サーチウィンドウの中心をCenterPointObjectプロパティで位置決めしたあとに、Yオフセットを設定、または返す デフォルト: 0
CenterPntRotOffset	中心点の XY オフセット値 (CenterPntOffsetX, CenterPntOffsetY)を、CenterPointObject の Angle リザルトにしたがって回転するか否かを指定 デフォルト: False

プロパティ	説明
CenterX	ポーラーサーチツールの中心点として用いられる位置のX座標を指定  CenterPointObjectプロパティが他のビジョンオブジェクトに設定されると、自動的に設定されます。
CenterY	ポーラーサーチツールの中心点として用いられる位置のY座標を指定  CenterPointObjectプロパティが他のビジョンオブジェクトに設定されると、自動的に設定されます。
CheckClearanceFor	クリアランス (隙間)を確認するオブジェクトを設定
ClearanceCondition	クリアランス (隙間)の判断方法を指定
Confusion	検索する画像イメージ中に予想される複雑さの度合いを指定 これは、検索している実際の対象物ではない対象物で得られる形状スコアの最大値です。 デフォルト: 800
CoordObject	リザルトをコピーするCoordinatesオブジェクトを指定 コピー処理はオブジェクトが実行される時に行われ、Decisionの分岐機能などで実行されなかった場合、コピー処理は行われません デフォルト: None
CurrentResult	オブジェクトウィンドウのリザルトリストに表示するリザルト、または、1つのサーチウィンドウ内にある対象物などの複数のリザルトを、システムが検出する設定のときにデータを返すリザルトを定義
Description	ユーザー指定の説明を設定 デフォルト: 空欄
Enabled	このオブジェクトを実行するかを指定 デフォルト: True
FailColor	オブジェクトの検出不良時の色を選択 デフォルト: Red
Frame	どの位置に置いたフレームを使用するかを指定 デフォルト: none
FrameResult	Frameの何番目のリザルトを使用するかを指定 デフォルト: 1
Graphics	表示するグラフィックを指定 デフォルト: 1 – All
LabelBackColor	オブジェクトラベルの背景の色を設定 デフォルト: Transparent (透明)
ModelObject	検索に使用するモデルを決定 デフォルト: Self
Name	Polarオブジェクトに独自のオブジェクト名を割りあてる デフォルト: Polar01

プロパティ	説明
PassColor	オブジェクトの検出良時の色を選択 デフォルト: LightGreen
PassType	オブジェクト検出時の良否判定の条件を定義 デフォルト: SomeFound
Radius	オブジェクトの中心点から、オブジェクトの検索リング外側までの距離を定義 デフォルト: 50
SaveTeachImage	モデルをティーチするときに、画像ファイルに保存するかを設定
ScoreMode	Fail時のリザルトを表示させるための閾値を設定、または返す
ShowModel	教示モデルの内部グレースケール表示を閲覧 検出マスクを設定
Thickness	Polarオブジェクトの被検索領域である検索リングの厚さを定義 厚さは、Radiusプロパティで定義されたリング外側から内側に向かってピクセルを測定します。 デフォルト: 5

### Polarオブジェクトのリザルト

下の一覧は、Polarオブジェクトリザルトを簡単に説明した概略です。各リザルトの詳細は、Vision Guide 7.0プロパティ&リザルトリファレンスを参照してください。

リザルト	説明
Angle	検出されたパーツの回転量を角度で返す
CameraX	カメラ座標系におけるPolarオブジェクトCenterPoint位置のX座標を返す
CameraY	カメラ座標系におけるPolarオブジェクトCenterPoint位置のY座標を返す
CameraXYU	実行時のみ カメラ座標系における検出パーツ位置のCameraX座標, CameraY座標, CameraU座標を返す
ClearanceOK	クリアランス (隙間)の判断結果を返す
Found	オブジェクトが検出されたかどうか、つまり、対象物あるいはパーツがAcceptプロパティの現在の設定を超える形状スコア値を持っているかどうかの値を返す
Passed	検出結果が良か否かを返す
PixelX	検出パーツ位置の (モデル原点を基準にした)X座標をピクセルで返す
PixelY	検出パーツ位置の (モデル原点を基準にした)Y座標をピクセルで返す

リザルト	説明
PixelXYU	実行時のみ 検出パーツ位置のPixelX座標, PixelY座標, PixelU座標をピクセルで返す
RobotX	ロボット座標系におけるPolarオブジェクトCenterPoint位置のX座標を返す
RobotY	ロボット座標系におけるPolarオブジェクトCenterPoint位置のY座標位置を返す
RobotU	ロボット座標系におけるPolarオブジェクト角度リザルトのU座標位置を返す
RobotXYU	実行時のみ ロボット座標系における検出パーツ位置のRobotX座標, RobotY座標, RobotU座標を返す
Score	実行時に検出された対象物が、Polarオブジェクトで検索するモデルに一致する度合いを表すINTEGER数値を返す
Time	オブジェクト処理に要した時間を返す (単位: ミリ秒)

## NOTE



X座標とY座標 (CameraX, CameraY, PixelX, PixelY, RobotX, RobotY....)を返すPolarオブジェクトのリザルトはすべて、CenterPointプロパティからその座標の値を受け取ります。Polarオブジェクトの実行前にCenterPointプロパティの値が設定されると、それらの値は、Polarオブジェクトのリザルトとして引き渡されます。Polarオブジェクトは、回転角度の計算をするのみで、中心位置のX,Y座標の計算はしませんが、CenterPointプロパティからのX, Y座標位置を提供しますので、別のビジョンオブジェクトからX, Y座標リザルトを取得する必要はありません。

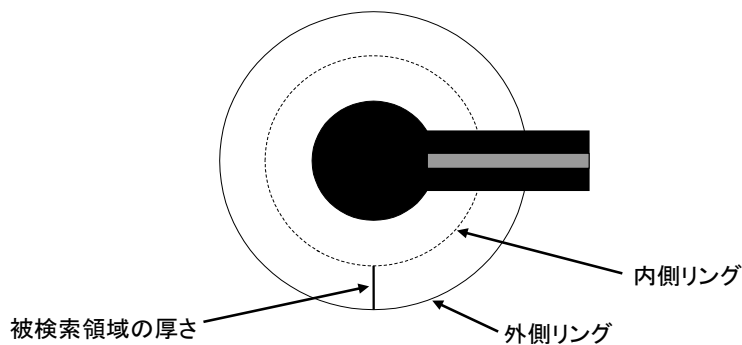
### Polarオブジェクトの基本的知識

Polarオブジェクトの目的は、指定されたオブジェクトやパターンの回転量を検出することです。このために、まず、基本的に特定の厚さを持つ円形リングであるポーラーモデルを登録する必要があります。登録を完了し、Polarオブジェクトを実行すると、円形リングと検出したいパーツの現在の回転とを比較して角度のずれを計算し、Polarオブジェクトの結果の1つとして返します。

例えば、中心点のまわりに回転するパーツがあり、ロボットでそのパーツをピックアップしてパレット上に置くとした場合、このパーツは、さまざまな方向を向いているので、オブジェクトの回転量を判定しない限り、ピックアップするのは困難です。

理解しやすいように、例をあげて説明します。オブジェクトが図Aの黒色と灰色で示すパーツのような物体である場合は、オブジェクトの外側部分を横切るポーラーモデルを定義することができます。

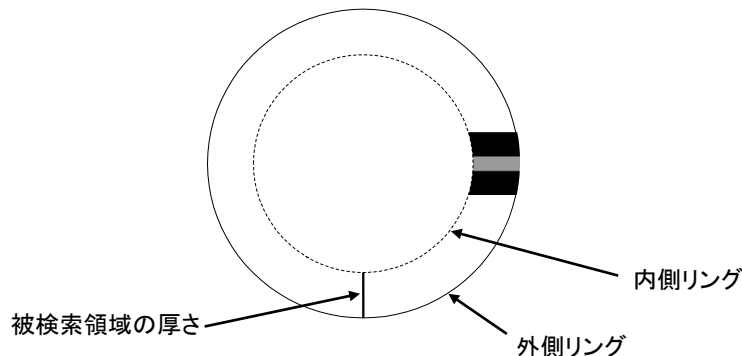
内側リングと外側リングの間にある領域に注目してください。この領域は、このパーツに登録されたPolarオブジェクトのモデルを定義します。リング内のいくつかの領域は、リングの残りの部分と強い特徴差異があるため、非常に識別しやすいということがわかるはずです。これらの領域は、パーツの回転量を検出するための重要なキーとなります。



図A: Polarオブジェクトを使用するパーツ例

図Bは、登録されたモデルの外観を示します。登録されたモデルは、ほとんど白く、一部が黒と灰色のリングです。

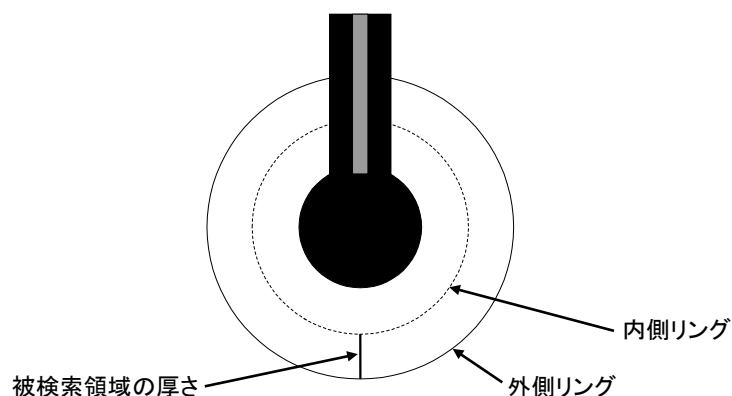
Polarオブジェクトを実行すると、Polarオブジェクトは、サーチウィンドウ内でこのタイプのパターン、つまり、一部が黒と灰色でリングの残りの部分が白いパターンを検索します。



図B: 図Aに示すパーツのポーラーモデル

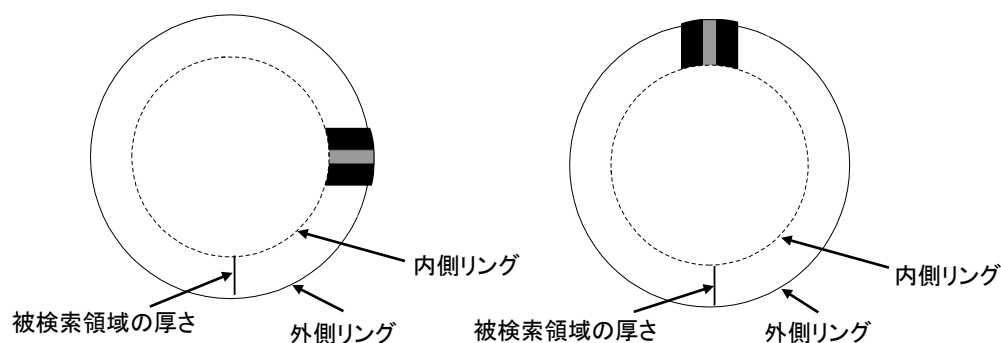


図Cでは、モデルに対して約90°回転したパーツを示しています。Polarオブジェクトは、モデルとオブジェクトの新規位置との角度のずれを計算します。



図C: 登録されたモデルから90°回転したパーツ

図D では、初期に0°で登録されたモデルを左図に、登録されたモデルから90°回転したパーツを右図に示しています。



図D: ポーラーモデルと回転したパーツのポーラー

#### Polar オブジェクトに関連する主要なパラメーター

前述の項目で示した図から、Polarオブジェクトに関連する主要なパラメーターは3つあり、各パラメーターは最適ナリザルを取得する上で重要であることが理解していただけたことと思います。これらのパラメーターは、次のプロパティによって定義されます。

- CenterPointプロパティ
- Radiusプロパティ
- Thicknessプロパティ
- AngleOffsetプロパティ

CenterPointプロパティは、Polarオブジェクトの中心位置を定義します。

前述したように、登録されたモデルの中心点と、回転角度を検索するパーツの中心位置とが完全に一致するように、Polarオブジェクトの中心点を適切な位置に合わせる必要があります。適切な位置に合わせないと、XY平面での並行移動によって、角度リザルトが不正確になる場合があります。

Radiusプロパティは、Polarオブジェクトの中心点から、オブジェクトの検索リング外側までの距離を定義します。

このプロパティによって、被検索領域の外側境界が定義されます。

**Thickness**プロパティは、**Polar**オブジェクトの外側リングから想像上の内側リングまでの距離を (ピクセル単位で)定義します。

実際には被検索領域の厚さを定義します。

**AngleOffset**プロパティは、**Polar**オブジェクトの回転位置を表示するために使われる、インジケータグラフィックラインの角度位置を設定するメカニズムを提供します。

このラインは、**Polar**オブジェクト実行後、時計の3時方向 (0°の位置)に現われます。しかし、検索する対象物の最も関心のある回転位置にインジケータ グラフィックを合わせたい場合は、必要に応じてインジケータ グラフィックの位置を変更することができます。(この**AngleOffset**プロパティについては、通常、モデルの登録と**Polar**オブジェクトの実行をした後で設定されますので、注意してください。)

### オブジェクトの回転を判定する方法

オブジェクトの回転を判定する必要があるアプリケーションとして、一般的に集積回路の型をロボットでピックアップする例をあげることができますが、この場合、型のXY位置と回転方向を知る必要があります。型は背景と異なるグレイの影で表され、型の表面は、0° ~ 360°までの回転角度を示すグレイレベル情報を含んでいます。

このアプリケーションでは、型の中心を検出するのに、(“**Blob01**”というオブジェクト名を持つ)**Blob**オブジェクトを使います。なお、**Blob**オブジェクトの代わりに**Correlation**オブジェクトを使うこともできます。

**Polar**オブジェクトを作成し、**Polar**オブジェクトの中心位置に**Blob**オブジェクトのX, Y位置リザルトを使用します。これは、**Polar**オブジェクトの**CenterPoint**プロパティを “**Blob01**” にすることで設定されます。

次に、**Radius**プロパティと**Thickness**プロパティとして適切な値を**Polar**オブジェクトに設定します。そして、グレイレベルのポラーモデルを登録し、0°の型モデルを提供します。

方向の異なる新規の型を検索する場合、まず、ポラーサーチウィンドウが、**Blob**オブジェクト“**Blob01**”からのXY位置リザルトと、**Radius**プロパティおよび**Thickness**プロパティの値から構成されます。

次に、このポラーサーチウィンドウ内で、事前に登録されていたモデルの回転角度に等しい角度が検索されます。そして、モデルが検出された角度が、**Angle**リザルト(ピクセル単位)と**RobotU**リザルト(ロボット座標系)として返されます。



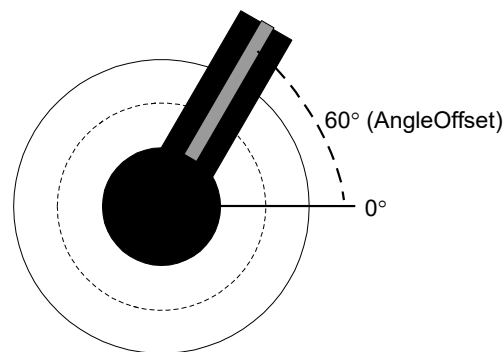
**重要注意事項:** 返された**RobotU**リザルトは、ロボット座標系における実際のパーツ回転角度ですが、グリッパーは正確に回転角度0°で装着されているわけではないので注意してください。グリッパーは少なくとも2° ~ 3°ずれているはずです。パーツへの動作では、この点をプログラムで調整してください。

### 登録されたモデルの角度のずれの調整

ポーラーモデルを登録すると、オリジナルの円形モデルの回転角度は0°とみなされ、このとき、回転角度0°の位置は時計の3時方向で表わされます。下図は、被検索領域がおおよそ1時方向にあると登録されたモデルを示しています。このモデルが登録されると、モデルの0°の位置は、すべてのPolarオブジェクトで時計の3時方向となります。

しかし、インジケータ表示したいのは実際のパーツ回転角度ですから、Angleoffsetプロパティを使用して、ポーラー角度のインジケータ グラフィックの位置を適切に調整しなければなりません。

例えば、おおよそ1時方向を指すように登録されたパーツには、60°の角度のずれが必要となります。(PolarオブジェクトのAngleOffsetプロパティには、60°の値を設定しなければなりません。)



AngleOffsetプロパティに 60°の値を設定したポーラーモデル

### Polarオブジェクト実行上の課題

Polarオブジェクトを教示するときに主に配慮しなければならない点は、ポーラーモデルがオブジェクト方向を判定するための情報を非常に多く含んでいるということです。

したがって、Polarオブジェクトの期待精度や実行スピードによって、Polarオブジェクトの検索に使われるポーラーモデルのサイズを決定することになります。

Polarオブジェクトの回転リザルトに1/2°の精度を必要とする場合には、Polarオブジェクトの幅はわずかに180ピクセルしか必要としません。(1/4ピクセル検索精度と仮定して1ピクセルあたり2°、あるいはPolarオブジェクト解像度単位あたり0.5°)

また、Polarオブジェクトの精度は、モデルのグレイレベル情報によって左右されます。

Polarオブジェクトの厚さ (ピクセル単位)を選択するときは、被検索画像イメージの中心位置の特定にわずかな違いがあった場合でも、角度検出によって信頼性のあるリザルトを出すように、十分な情報を含むようにしなければなりません。

Thicknessプロパティに1ピクセルを設定すると、画像イメージが1ピクセル分まちがった場所に置かれた場合には、ポーラー変換によって全く異なるピクセルがポーラー画像イメージに転送されることになります。

画像イメージ元の場所にいくらかの余裕を与えるために、Thicknessプロパティには5の値が設定されますが、この設定によると、画像イメージ元の場所が1ピクセルずれた場合でも、それに対応するポーラー画像イメージのピクセルは、1/5ずれるだけです。(Thicknessプロパティの最小値を5に設定しているのは、以上のような理由によります。)

PolarオブジェクトのRadiusプロパティとThicknessプロパティは、モデル中のグレイ情報量と希望する検索速度に基づいて選択してください。


検索速度は、RadiusプロパティとThicknessプロパティの値に比例します。Radiusプロパティを低い値に設定し、Thicknessプロパティに5の値を設定した場合に、最高速の検索速度を得ることができます。ただし、多くの場合で、Thicknessプロパティの設定値として5の値は、ポーラーモデルを正確に検出するのに十分な値ではありません。

### Polarオブジェクトの使用方法

ここからは、Polarオブジェクトの作成方法と使用方法を説明します。


- 新規Polarオブジェクトの作成方法
- サーチウィンドウの位置とサイズの設定
- Polarオブジェクトの関連プロパティの設定
- Polarオブジェクトのテストとリザルトの確認
- プロパティの調整と再テスト

次に説明するステップを開始する前に、新規ビジョンシーケンスを作成するか、使用する

ビジョンシーケンスを選択してください。作業するビジョンシーケンスがない場合は、 <新規シーケンス>ボタンをクリックすると、新規ビジョンシーケンスが作成できます。

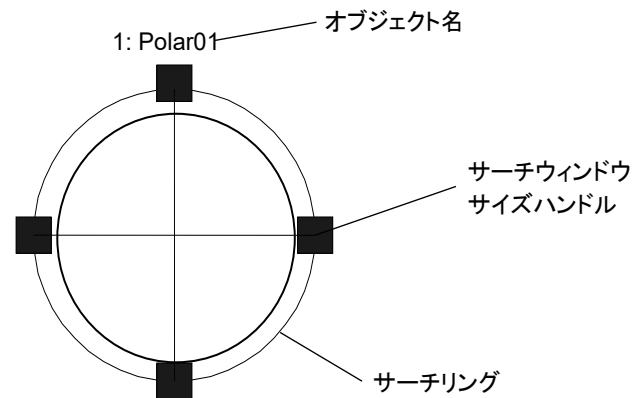
すでにビジョンシーケンスを作成済みの場合は、Vision Guideウィンドウのシーケンスツリーをクリックして、作成済みのビジョンシーケンスを選択してください。新規ビジョンシーケンスの作成方法、または作成済みビジョンシーケンスの選択方法の詳細は、「5. ビジョンシーケンス」を参照してください。

### Step 1: 新規Polarオブジェクトの作成

- (1) Vision Guideツールバー - <全ツール> -  <Polar (ポーラサーチ)>ボタンをクリックしてください。
- (2) 画像イメージ表示部にマウスを動かしてください。マウスポインターがPolarオブジェクトアイコンに変わります。
- (3) アイコンが画像イメージ表示部の目的の位置に来るまで、マウスを動かし続けてください。それから、マウスを左クリックして、オブジェクトを作成してください。
- (4) オブジェクト名が自動的に作成されるのに注目してください。この例では、このシーケンスで最初に作成されたPolarオブジェクトなので、“Polar01”というオブジェクト名になります。(オブジェクト名の変更方法については後述します。)

### Step 2: Polarオブジェクトの位置決め

画面上には、下図のようなPolarオブジェクトが表示されています。



#### 新規Polarオブジェクト

Polarオブジェクトには円形のサーチウィンドウがあります。オブジェクト全体の位置、または半径を変更することができます。

オブジェクト全体を移動するには、オブジェクト名または外側の境界線の周辺をクリックして、マウスの左側ボタンを押し続けながら、オブジェクト全体を画面上の新しい位置にドラッグしてください。位置が決まったら、マウスを離してください。Polarオブジェクトが画面上の新しい位置にきます。

半径を変更するには、マウスポインターをサイズハンドルの1つへ移動させ、マウスの左側ボタンを押して、それからマウスを動かしてください。

### Step 3: Polarオブジェクトプロパティの設定

Polarオブジェクトプロパティの設定を行います。プロパティを設定するには、関連するプロパティの設定値フィールドをクリックし、新規の値を入力してください。あるいは、ドロップダウンリストが表示されている場合は、リスト中の該当項目をクリックしてください。

次に、Polarオブジェクトで一般的に使われるプロパティのいくつかを示します。その他、さまざまなビジョンオブジェクトの多くで使われるAbortSeqOnFail, Graphicsなどのプロパティの詳細は、Vision Guide 7.0プロパティ&リザルトリファレンス、および本章の「Polarオブジェクトのプロパティ」を参照してください。

#### Nameプロパティ

新規に作成されたPolarオブジェクトに、デフォルトとして与えられるオブジェクト名は“Polarxx”です。xxは、1つのビジョンシーケンス内で複数のPolarオブジェクトを用いるとき、それらのPolarオブジェクトで区別するための番号です。ビジョンシーケンスで1番目に用いられるPolarオブジェクトには、“Polar01”というオブジェクト名がデフォルトで設定されます。オブジェクト名を変更したい場合は、Nameプロパティの設定値フィールドをクリックして、変更後の名称を入力し、リターンキーを押してください。画面上でPolarオブジェクト名を表示する場所は、すべて変更後の名称に変わります。

#### CenterPointObject プロパティ

通常、このプロパティをシーケンスの前の方で行われるオブジェクトの1つに設定します。これが、実行時のPolarオブジェクトの中心点を決定します。

- Thicknessプロパティ** 通常、パーツ角度を定めるためのモデル情報を持つのに十分な値に、このプロパティを設定します。
- AngleOffsetプロパティ** 通常、このプロパティを目的位置での最終角度リザルトの位置になるように設定します。例えば、時計の分針を検索している場合、表示された角度が分針にぴったり合うようにAngleOffsetプロパティを調整します。

#### Step 4: Polarオブジェクトの実行とリザルトの確認

Polarオブジェクトを実行するには、次のようにしてください。

実行パネルのオブジェクトの<Run>ボタンをクリックしてください。CenterPointObjectが最初に実行されます。

Polarオブジェクトのリザルトが表示されます。この時点で確認すべき主要なリザルトは次のとおりです。

**Angleリザルト** 検出モデルの角度(単位は度)

### 6.2.7 OCRオブジェクト

#### OCRオブジェクトの概要

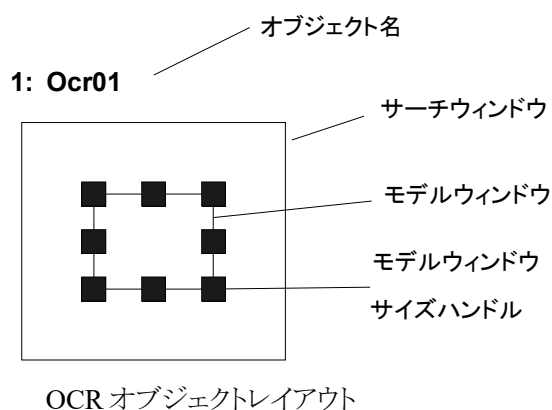
OCRオブジェクト(光学文字認識)は、一列に並んだ文字列を、特定のフォントや文字サイズのイメージとして認識することができます。OCRオブジェクトのGUIには、SEMIスタンダードをもとにフォントを作成したり、ユーザー定義フォントをイメージ中の文字やASCII定義ファイルから作成するフォントウィザードがあります。作成したフォントをディスクへエクスポートしたり、同じプロジェクト内の、あるいは別のプロジェクトのOCRオブジェクトへインポートすることができます。



OCRオブジェクトは、OCRオプションがインストールされており、有効になっている場合のみ使用可能です。

#### OCRオブジェクトのレイアウト

OCRオブジェクトには、次に示すようにサーチウィンドウとモデルウィンドウがあります。



円弧に沿って並んだ文字列は、“Arc”型サーチウィンドウを使用することで、特定のフォントや文字サイズイメージとして認識することができます。

## OCRオブジェクトプロパティー

OCRオブジェクトで使われるプロパティーの一覧を示し、簡単に説明します。各プロパティーの詳細は、Vision Guide 7.0 プロパティー & リザルトリファレンスマニュアルを参照してください。

プロパティー	説明
AbortSeqOnFail	オブジェクトの実行に失敗した (不良判定になった) とき、直ちにシーケンス全体を中断し、それ以降のオブジェクトを処理しないように指定 デフォルト: False
Caption	見出しをOCRオブジェクトに割りあてる デフォルト: 空欄
CenterPointObject	オブジェクトの中心点として用いられる位置を指定 このプロパティーを“Screen”に設定すると画面上の任意の位置にオブジェクトを配置できますが、他のビジョンオブジェクトを指定した場合、そのオブジェクトのPixelX, PixelYリザルトに中心点が設定されます。 デフォルト: Screen
CenterPntObjResult	CenterPointObjectプロパティーからどのリザルトを使うのかを指定 Allを指定すると、指定されたビジョンオブジェクトリザルトのすべて (NumberFound) にOCRオブジェクトが適用されます。 デフォルト: 1
CenterPntOffsetX	サーチウィンドウの中心をCenterPointObjectプロパティーで位置決めしたあとに、Xオフセットを設定、または返す デフォルト: 0
CenterPntOffsetY	サーチウィンドウの中心をCenterPointObjectプロパティーで位置決めしたあとに、Yオフセットを設定、または返す デフォルト: 0
CenterPntRotOffset	中心点のXYオフセット値 (CenterPntOffsetX, CenterPntOffsetY) を、CenterPointObjectのAngleリザルトにしたがって回転するか否かを指定 デフォルト: False
CurrentResult	オブジェクトウィンドウのリザルトリストに表示するリザルト、または、1つのサーチウィンドウ内にある対象物などの複数のリザルトを、システムが検出する設定のときにデータを返すリザルトを定義 FindChar=Falseの場合は、グレイアウト デフォルト: 1
Description	ユーザー指定の説明を設定 デフォルト: 空欄
Dictionarymode	辞書モードを指定 デフォルト: All

プロパティ	説明
Direction	円弧に沿って並んだ文字の方向を設定 デフォルト: InsideOut
Enabled	このオブジェクトを実行するかを指定 デフォルト: True
ExportFont	現在のフォントをディスクにエクスポート
FailColor	オブジェクト検出が不良時の色を指定 デフォルト: Red
FindChar	1文字ずつオブジェクトとして扱うかを指定 デフォルト: False
Frame	どの位置に置いたフレームを使用するかを指定 デフォルト: none
FrameResult	Frameの何番目のリザルトを使用するかを指定 デフォルト: 1
Graphics	表示するグラフィックを指定 Default: 1 – All
ImportFont	Vision GuideのGUIから、ファイルダイアログを起動し、フォントファイルをインポート
InvalidChar	Text resultで無効な文字を表すために使われた文字を設定、あるいは返す デフォルト: “?”
LabelBackColor	オブジェクトラベルの背景の色を設定 デフォルト: Transparent (透明)
ModelWin	実行時のみ 1コールで、モデルウィンドウの左端, 上端, 高さ, 幅のパラメータを設定、または返す
ModelWinAngle	モデルウィンドウの角度を定義
ModelWinCenterX	モデルウィンドウの中心のX座標値を定義
ModelWinCenterY	モデルウィンドウの中心のY座標値を定義
ModelWinLeft	モデルウィンドウの最左端の位置を規定
ModelWinHeight	モデルウィンドウの高さを定義 デフォルト: 50
ModelWinTop	モデルウィンドウの最上端の位置を規定
ModelWinType	モデルウィンドウのタイプを定義
ModelWinWidth	モデルウィンドウの幅を定義 デフォルト: 50
Name	OCRオブジェクトに独自のオブジェクト名を割りあてる デフォルト: Ocr01
PassColor	オブジェクトの検出良時の色を選択 デフォルト: Light Green



プロパティ	説明
PassType	オブジェクト検出時の良否判定の条件を定義 デフォルト: SomeFound
Polarity	オブジェクトと背景の差 (明るい背景に暗いオブジェクトか、暗い背景に明るいオブジェクトか)を定義 デフォルト: 1 – DarkOnLight
SearchWin	実行時のみ  1 コールで、サーチウィンドウの左端, 上端, 高さ, 幅、または、中心の X 座標, 中心の Y 座標, 円内周の半径サイズ, 円外周の半径サイズのパラメーターを設定、または返す
SearchWinAngleEnd	被検索領域の終了角度を定義
SearchWinAngleStart	被検索領域の開始角度を定義
SearchWinCenterX	被検索領域の中心のX座標値を定義
SearchWinCenterY	被検索領域の中心のY座標値を定義
SearchWinHeight	被検索領域の高さを定義 (単位: ピクセル) デフォルト: 100
SearchWinLeft	被検索領域の最左端の位置を定義 (単位: ピクセル)
SearchWinRadiusInner	被検索領域の円内径を定義
SearchWinRadiusOuter	被検索領域の円外径を定義
SearchWinTop	被検索領域の最上端の位置を定義 (単位: ピクセル)
SearchWinType	被検査領域のタイプ (Rectangle, Arc)を定義
SearchWinWidth	被検索領域の幅を定義 (単位: ピクセル) デフォルト: 100

### OCRオブジェクトのリザルト

次は、OCR オブジェクトのリザルトを簡単に説明した表です。各リザルトの詳細は、Vision Guide 7.0 プロパティ&リザルトリファレンスマニュアルを参照してください。

リザルト	説明
Angle	検出された文字の回転量を角度で返す
Found	オブジェクトが検出されたかどうかを返す
Passed	検出結果が良か否かを返す
CameraX	カメラ座標系における検出文字のX座標位置を返す
CameraY	カメラ座標系における検出文字のY座標位置を返す
NumberFound	検出された文字数を返す
MaxX	文字の外接長方形の最大Xピクセル座標を返す
MaxY	文字の外接長方形の最大Yピクセル座標を返す
MinX	文字の外接長方形の最小Xピクセル座標を返す
MinY	文字の外接長方形の最小Yピクセル座標を返す
PixelX	検出文字位置のX座標をピクセルで返す

## 6. ビジョンオブジェクト

---


リザルト	説明
PixelY	検出文字位置のY座標をピクセルで返す
RobotX	ロボット座標系における検出文字のX座標位置を返す
RobotY	ロボット座標系における検出文字のY座標位置を返す
ShowAllResults	特定のビジョンオブジェクトのリザルトすべてを、表形式でダイアログに表示  リザルトの比較が簡単にできます。
Text	サーチ中に検出した文字を返す
Time	オブジェクト処理に要した時間を返す (単位: ミリ秒)

## OCRオブジェクトの使用方法


OCR オブジェクトを使用するときの手順を、以下に挙げています。次項で詳しく説明します。

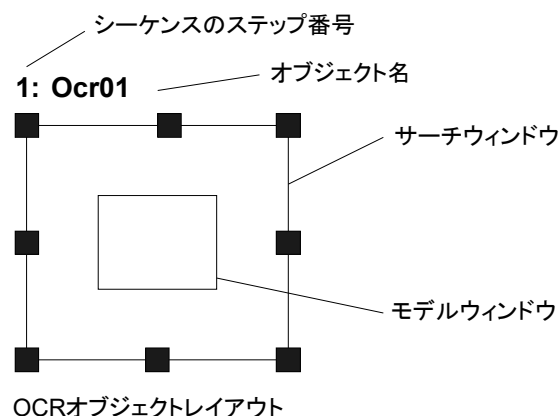
- 新規 OCR オブジェクトの作成
- 新規オブジェクトのフォントの作成、またはインポート
- フォントのキャリブレーション (必要に応じて)
- OCR オブジェクトの関連プロパティの設定
- OCR オブジェクトのテストとリザルトの確認
- プロパティの調整と再テスト

下に示す手順を始める前に、新規のビジョンシーケンスを作成しているか、または使用するビジョンシーケンスを選択していなければなりません。もし、ビジョンシーケンスがない場合、

 <新規シーケンス>ボタンをクリックすると、新規のビジョンシーケンスを作成することができます。また、Vision Guide ウィンドウのシーケンスツリーから以前に作成していたシーケンスをクリックし、OCR オブジェクトをクリックして選択することもできます。

## 新規OCRオブジェクトの作成

1. Vision Guide ツールバー - <全ツール> -  <OCR>ボタンをクリックしてください。
2. OCR オブジェクトボタンの上に OCR アイコンが現れます。
3. OCR アイコンをクリックして、Vision Guide ウィンドウのイメージディスプレイへドラッグします。
4. オブジェクトの名前は自動的に作成されます。例えば、シーケンスに作成された最初の OCR オブジェクトならば、“Ocr01”となります。(名前の変更方法については、後で説明します。)



### フォントの作成

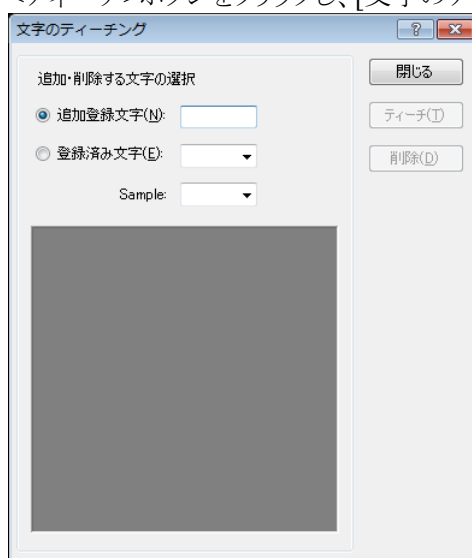
フォントは、読み取る文字画像に合わせて作成する必要があります。他のプロジェクトからフォントを **ImportFont** プロパティでインポートすることもできます。この場合、フォントを作成する必要がありません。

また、一般的な欧米、日本語のフォントはシステム内に組み込まれています。入力画像のサイズによってはフォント作成が必要な場合があります。

ユーザー定義フォントは、イメージファイルから作成することができます。

以下の手順に従い、フォントを作成します。

- (1) フローチャートから **OCR オブジェクト** をクリックし、モデルウィンドウを文字に合わせてます。
- (2) <ティーチ>ボタンをクリックし、[文字のティーチング]ダイアログを表示します。



- (3) 新規文字を登録する場合は、<追加登録文字>ボタンを選択し、<ティーチ>ボタンをクリックします。1つの文字に対して複数のイメージを登録することができます。登録可能な文字は、日本語(全角)、半角英数文字、半角記号です。登録済み文字をフォントから削除する場合、<登録済み文字>ボタンを選択後、削除する文字をドロップダウンで選択し、<削除>ボタンをクリックします。

### 6.2.8 CodeReaderオブジェクト

#### CodeReaderオブジェクトの概要

CodeReader オブジェクトは、バーコードや二次元コードを読み取ります。

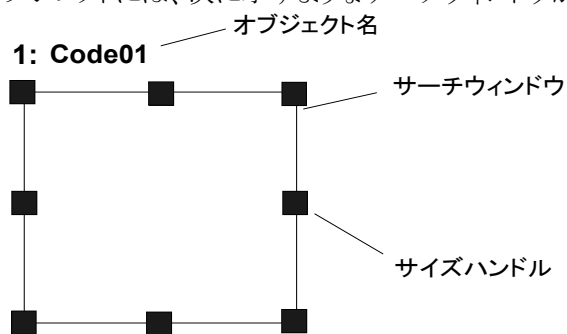
対応コードは以下のとおりです。

EAN13	数字 0 ~ 9, 固定長
Code39	0 ~ 9, A ~ Z, ./+-%\$Spc, 可変長
DataMatrix	二次元コード *1
Interleaved 2 of 5	数字 0 ~ 9, 固定長
Code128	Full ASCII, 可変長
Codabar	数字 0 ~ 9, \$ - : / . +
PDF417	二次元コード
QR	二次元コード
EAN	8桁 0 ~ 9, 固定長
UPC A	数字 0 ~ 9, 固定長
UPC E	数字 0 ~ 9, 固定長

\*1: ECC200 規格のみ対応

#### CodeReaderオブジェクトレイアウト

CodeReader オブジェクトには、次に示すようなサーチウィンドウがあります。



CodeReaderオブジェクトレイアウト

## CodeReaderオブジェクトプロパティ

次の表は、CodeReader オブジェクトで使われるプロパティの簡単な説明です。各プロパティの詳細は、Vision Guide 7.0 プロパティ&リザルトリファレンスマニュアルを参照してください。

プロパティ	説明
AbortSeqOnFail	オブジェクトの実行に失敗した (不良判定になった) とき、直ちにシーケンス全体を中断し、それ以降のオブジェクトを処理しないように指定 デフォルト: False
Caption	見出しを CodeReader オブジェクトに割りあてる デフォルト: 空欄
CenterPointObject	オブジェクトの中心点として用いられる位置を指定 このプロパティを“Screen”に設定すると画面上の任意の位置にオブジェクトを配置できますが、他のビジョンオブジェクトを指定した場合、そのオブジェクトの PixelX, PixelY リザルトに中心点が設定されます。 デフォルト: Screen
CenterPntObjResult	CenterPointObject プロパティからどのリザルトを使うのかを指定 All を指定すると、指定されたビジョンオブジェクトリザルトのすべて (NumberFound) に CodeReader オブジェクトが適用されます。 デフォルト: 1
CenterPntOffsetX	サーチウィンドウの中心を CenterPointObject プロパティで位置決めしたあとに、X オフセットを設定、または返す デフォルト: 0
CenterPntOffsetY	サーチウィンドウの中心を CenterPointObject プロパティで位置決めしたあとに、Y オフセットを設定、または返す デフォルト: 0
CenterPntRotOffset	中心点の XY オフセット値 (CenterPntOffsetX, CenterPntOffsetY) を、CenterPointObject の Angle リザルトにしたがって回転するか否かを指定 デフォルト: False
CodeType	CodeReader オブジェクトで検索するバーコードまたは 2 次元コードのタイプを設定、あるいは返す デフォルト: 0 – Auto
CoordObject	リザルトをコピーする Coordinates オブジェクトを指定 コピー処理はオブジェクトが実行される時に行われ、Decision の分岐機能などで実行されなかった場合、コピー処理は行われません デフォルト: None

プロパティ	説明
CurrentResult	オブジェクトウィンドウの結果リストに表示する結果、または、1つのサーチウィンドウ内にある対象物などの複数の結果を、システムが検出する設定のときにデータを返す結果を定義
Description	ユーザー指定の説明を設定 デフォルト: 空欄

プロパティ	説明
Enabled	このオブジェクトを実行するかを指定 デフォルト: True
FailColor	オブジェクトの検出結果が不良時の色を指定します。 デフォルト: Red
Frame	指定したフレームにしたがって、カレントオブジェクトの検索位置を設定 デフォルト: none
FrameResult	Frame の何番目のリザルトを使用するかを指定 デフォルト: 1
Graphics	表示するグラフィックを指定 デフォルト: 1 – All
LabelBackColor	オブジェクトラベルの背景の色を設定 デフォルト: Transparent (透明)
Name	CodeReader オブジェクトに独自のオブジェクト名を割りあてる デフォルト: Code01
NumberToFind	検出するコードの数を設定 最大: 8 デフォルト: 1
Orientation	バーコードの向きを設定
PassColor	オブジェクトの検出良時の色を選択 デフォルト: Light Green
PassType	オブジェクト検出時の良否判定の条件を定義 デフォルト: SomeFound
SearchWin	実行時のみ 1 コールで、サーチウィンドウの左端, 上端, 高さ, 幅のパラメーターを設定、または返す
SearchWinHeight	被検索領域の高さを定義 (単位: ピクセル) デフォルト: 100
SearchWinLeft	被検索領域の最左端を定義 (単位: ピクセル)
SearchWinTop	被検索領域の最上端を定義 (単位: ピクセル)
SearchWinWidth	被検索領域の幅を定義 (単位: ピクセル) デフォルト: 100



### CodeReaderオブジェクトのリザルト

次は、CodeReader オブジェクトのリザルトについて簡単に説明しています。各リザルトについての詳細は、Vision Guide 7.0 プロパティ&リザルトリファレンスマニュアルを参照してください。


リザルト	説明
Angle	検出されたコードの傾斜角度を返す
CameraX	カメラ座標系における検出パーツ位置の X 座標を返す (単位: mm)
CameraY	カメラ座標系における検出パーツ位置の Y 座標を返す (単位: mm)
Found	オブジェクトが検出されたかどうかを返す
FoundCodeType	検出されたコードタイプを返す
NumberFound	検出されたコード数を返す  (最小 0 個から、最大は NumberToFind プロパティで設定した 検出個数になります。)
Passed	検出結果が良か否かを返す
PixelX	検出パーツ位置の X 座標をピクセルで返す
PixelY	検出パーツ位置の Y 座標をピクセルで返す
PixelXYU	実行時のみ  検出パーツ位置の PixelX 座標, PixelY 座標, PixelU 座標をピク セルで返す
RobotX	ロボット座標系における検出パーツの X 座標位置を返す
RobotY	ロボット座標系における検出パーツの Y 座標位置を返す
RobotU	ロボット座標系における検出パーツの U 座標位置を返す
RobotXYU	実行時のみ  ロボット座標系における、検出パーツ位置の RobotX 座標, RobotY 座標, RobotU 座標を返す
ShowAllResults	特定のビジョンオブジェクトのリザルトすべてを、表形式でダイア ログに表示  リザルトの比較が簡単にできます。
Text	サーチ中に検出した文字を返す
Time	オブジェクト処理に要した時間を返す (単位: ミリ秒)

### CodeReaderオブジェクトの使用方法


CodeReader オブジェクトを使用するときの手順を、以下に挙げています。次項で詳しく説明します。

- 新規 CodeReader オブジェクトの作成
- サーチウィンドウの位置とサイズの設定
- CodeReader オブジェクトの関連プロパティの設定
- CodeReader オブジェクトのテストとリザルトの確認
- プロパティの調整と再テスト

下記に示す手順を始める前に、新規のビジョンシーケンスを作成しているか、または使用するビジョンシーケンスを選択していなければなりません。もし、ビジョンシーケンスがない

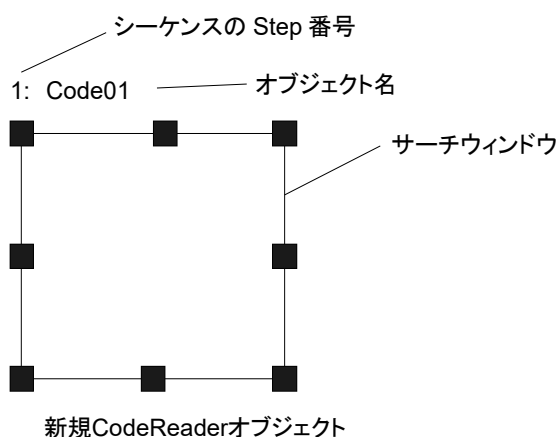
場合、 <新規シーケンス>ボタンをクリックすると、新規のビジョンシーケンスを作成することができます。また、Vision Guide ウィンドウのシーケンスツリーから以前に作成していたシーケンスをクリックし、CodeReader オブジェクトをクリックして選択することもできます。

#### Step 1: 新規CodeReaderオブジェクトの作成

1. Vision Guide ツールバー - <全ツール> -  <CodeReader>ボタンをクリックしてください。
2. CodeReader オブジェクトボタンの上に CodeReader アイコンが現れます。
3. CodeReader アイコンをクリックして、Vision Guide ウィンドウのイメージディスプレイへドラッグします。
4. オブジェクトの名前は自動的に作成されます。例えば、シーケンスに作成された最初の CodeReader オブジェクトならば、“Code01”となります。(名前の変更方法については、後で説明します。)

## Step 2: サーチウィンドウの位置とサイズの設定

画面上には、下図のような CodeReader オブジェクトが表示されます。



- (1) CodeReader オブジェクトの名称ラベルをクリックし、マウスを押し続けながら、CodeReader オブジェクトをドラッグして、サーチウィンドウの左上コーナーを好きな位置に合わせてください。
- (2) サイズハンドルをクリックして、マウスをドラッグすると、CodeReader オブジェクトサーチウィンドウのサイズが変更できますので、適当な大きさに調整してください。(サーチウィンドウは、グローブの被検索領域になります。)



バーコードの両側にスペースを設けてください。これがなければ、検索が失敗します。2次元バーコードの場合は、周囲に1セル以上の空白が必要です。

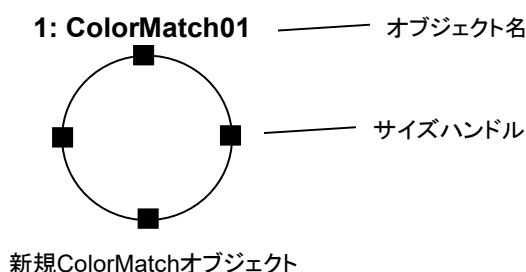
### 6.2.9 ColorMatch オブジェクト

#### ColorMatch オブジェクトの概要

CodeMatch オブジェクトは、カラーモデルと一致する色を検出します。

#### ColorMatch オブジェクトのレイアウト

ColorMatch オブジェクトのレイアウトは長方形、回転長方形、円形で、中点や半径があります。ColorMatch オブジェクトの位置は、ColorMatch オブジェクト名またはサーチウィンドウの外周をクリックし、移動させたい場所にドラッグすると、移動することができます。



CenterPoint プロパティで定義する ColorMatch オブジェクトの中点は、他のオブジェクトの位置を基準として設定することができます。つまり、ColorMatch オブジェクトの位置を調整した場合でも、オブジェクトやシーケンスを実行することによって、ColorMatch オブジェクトの中点が変わることがあります。

ColorMatch オブジェクトのサーチウィンドウは長方形、回転長方形、円形です。

SearchWinType が Circle の場合に ColorMatch オブジェクトサーチウィンドウの外側境界線のサイズを調節するには、サーチウィンドウサイズハンドルの中の 1 つをクリックし、内側あるいは外側にドラッグして半径を調整してください。

### ColorMatch プロパティ

ColorMatchオブジェクトで使われるプロパティの概略です。各プロパティの詳細は、Vision Guide 7.0 プロパティ&リザルトリファレンスマニュアルを参照してください。

プロパティ	説明
AbortSeqOnFail	オブジェクトの実行に失敗した (不良判定になった)とき、直ちにシーケンス全体を中断し、それ以降のオブジェクトを処理しないように指定 デフォルト: False
Accept	対象物が検出されたと判定するためのスコアの値を指定 設定値を低く設定すると、誤検出の原因となります。 デフォルト: 700
Caption	見出しを ColorMatch オブジェクトに割りあてる デフォルト: 空欄
CenterPointObject	オブジェクトの中心点として用いられる位置を指定 このプロパティを“Screen”に設定すると画面上の任意の位置にオブジェクトを配置できますが、他のビジョンオブジェクトを指定した場合、そのオブジェクトのPixelX, PixelYリザルトに中心点が設定されます。 デフォルト: Screen
CenterPntObjResult	CenterPointObjectプロパティからどのリザルトを使うのかを指定 Allを指定すると、指定されたビジョンオブジェクトリザルトのすべて (NumberFound)にColorMatchオブジェクトが適用されます。 デフォルト: 1
CenterPntOffsetX	サーチウィンドウの中心を CenterPointObject プロパティで位置決めしたあとに、X オフセットを設定、または返す デフォルト: 0
CenterPntOffsetY	サーチウィンドウの中心を CenterPointObject プロパティで位置決めしたあとに、Y オフセットを設定、または返す デフォルト: 0
CenterPntRotOffset	中心点の XY オフセット値 (CenterPntOffsetX, CenterPntOffsetY) を、CenterPointObject の Angle リザルトにしたがって回転するか否かを指定 SearchWinType が RotatedRectangle に設定されている場合、サーチウィンドウは Angle リザルトにしたがって回転します。 デフォルト: False

プロパティ	説明
CenterX	オブジェクトの中心点として用いられる位置の X 座標を指定 CenterPoint プロパティが他のビジョンオブジェクトに設定されると、自動的に設定されます。
CenterY	オブジェクトの中心点として用いられる位置の Y 座標を指定 CenterPoint プロパティが他のビジョンオブジェクトに設定されると、自動的に設定されます。
ColorMode	使用する色空間(RGB/HSV)を設定 デフォルト: RGB
CoordObject	リザルトをコピーするCoordinatesオブジェクトを指定 コピー処理はオブジェクトが実行される時に行われ、Decisionの分岐機能などで実行されなかった場合、コピー処理は行われません デフォルト: None
CurrentModel	実行時のみ ModelColor プロパティや VTeach で使用するモデルを指定 CurrentModel の値は 1 から NumberOfModels です。 デフォルト: 1
CurrentResult	オブジェクトウィンドウのリザルトリストに表示するリザルト、または、1つのサーチウィンドウ内にある対象物などの複数のリザルトを、システムが検出する設定のときにデータを返すリザルトを定義 デフォルト: 1
Description	ユーザー指定の説明を設定 デフォルト: 空欄
Enabled	このオブジェクトを実行するかを指定 デフォルト: True
FailColor	オブジェクトの検出結果が不良時の色を指定します。 デフォルト: Red
Frame	指定したフレームにしがたって、カレントオブジェクトの検索位置を設定 (フレームに対してオブジェクトの位置を決めることができます。) デフォルト: None
FrameResult	Frameの何番目のリザルトを使用するかを指定 デフォルト: 1
Graphics	表示するグラフィックを指定 デフォルト: 1 – All
LabelBackColor	オブジェクトラベルの背景の色を設定 デフォルト: Transparent (透明)
ModelColor	実行時のみ 実行時に RGB カラー値を直接入力し、モデルを教示 デフォルト: RGB(0, 0, 0)

プロパティ	説明
ModelColorTol	実行時のみ モデル色の許容差を設定 画素色が許容差の範囲内であれば、画素は変化しません。 デフォルト: 10 (ColorMode = RGB), 0,0,50 (ColorMode = HSV)
ModelName	実行時のみ 現在使用しているモデルの名前を設定
ModelObject	検索に使用するモデルを決定 デフォルト: Self
Name	ColorMatchオブジェクトに独自のオブジェクト名を割りあてる デフォルト: ColorMatch01
NumberOfModels	実行時のみ 使用しているカラーモデル数 実行中、このプロパティを設定すると、CurrentModel, VTeach を使用し、カラーモデルを教示できます。 デフォルト: 1
NumberToFind	作業中のサーチウィンドウで検出する対象物の数を定義 デフォルト: 1
PassColor	オブジェクトの検出良時の色を選択 デフォルト: LightGreen
PassType	オブジェクト検出時の良否判定の条件を定義 デフォルト: SomeFound
Radius	オブジェクトの中心点から、オブジェクトの検索リング外側までの 距離を定義 デフォルト: 50
SearchWin	実行時のみ 1コールでサーチウィンドウの左端, 上端, 高さ, 幅のパラメーター を設定、または返す
SearchWinAngle	被検索領域の角度を定義
SearchWinCenterX	被検索領域の中心のX座標値を定義
SearchWinCenterY	被検索領域の中心のY座標値を定義
SearchWinHeight	被検索領域の高さを定義 (単位: ピクセル)
SearchWinLeft	被検索領域の最左端を定義 (単位: ピクセル)
SearchWinTop	被検索領域の最上端を定義 (単位: ピクセル)
SearchWinType	被検査領域のタイプ (Rectangle, RotatedRectangle, Circle)を定義
SearchWinWidth	被検索領域の幅を定義 (単位: ピクセル)

## ColorMatch リザルト

ColorMatch オブジェクトのリザルトの概略です。各リザルトの詳細は、Vision Guide 7.0 プロパティ&リザルトリファレンスマニュアルを参照してください。

リザルト	説明
CameraX	カメラ座標系における検出パーツ位置の (モデル原点を基準にした)X 座標を返す (単位: mm)
CameraY	カメラ座標系における検出パーツ位置の (モデル原点を基準にした)Y 座標を返す (単位: mm)
ColorIndex	検索したカラーモデルのインデックスを返す
ColorName	検索したカラーモデルの名前を返す
ColorValue	ColorMode の設定により、検索したカラーモデルの RGB 値または HSV 値を返す
CameraXYU	実行時のみ カメラ座標系における検出パーツ位置の CameraX 座標, CameraY 座標, CameraU 座標を返す (単位: mm)
Found	色がカラーモデルのどれかと一致したかどうかを返す
NumberFound	検出されたオブジェクト数を返す (最小 0 個から、最大は NumberToFind プロパティで設定した検出個数になります。)
Passed	検出結果が良か否かを返す
PixelX	検出パーツ位置の X 座標をピクセルで返す
PixelY	検出パーツ位置の Y 座標をピクセルで返す
PixelXYU	実行時のみ 検出パーツ位置の PixelX 座標, PixelY 座標, PixelU 座標をピクセルで返す
RobotX	ロボット座標系における検出パーツの X 座標位置を返す
RobotY	ロボット座標系における検出パーツの Y 座標位置を返す
RobotXYU	実行時のみ ロボット座標系における検出パーツ位置の RobotX 座標, RobotY 座標, RobotU 座標を返す
Score	実行時に検出された対象物の色が、モデルの色に一致する度合いを表す 0 ~ 1000 の INTEGER 数値を返す
ShowAllResults	特定のビジョンオブジェクトのリザルトすべてを、表形式でダイアログに表示 リザルトの比較が簡単にできます。
Time	オブジェクト処理に要した時間を返す (単位: ミリ秒)

### ColorMatch オブジェクトの使用方法

これまでColorMatchオブジェクトの使い方を理解するための基礎を説明してきました。ここでは、ColorMatchオブジェクトを使う手順を説明します。

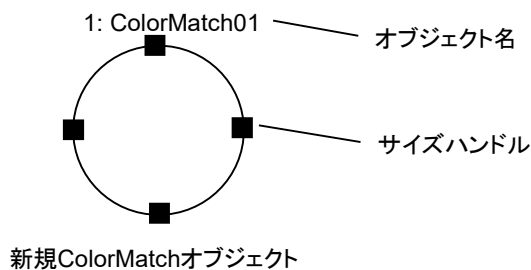
- 新規ColorMatchオブジェクトの作成
- サーチウィンドウの位置とサイズの設定
- ColorMatchオブジェクトに関連するプロパティの設定
- カラーモデルの教示
- ColorMatchオブジェクトのテストとリザルトの確認
- プロパティの調整と再テスト
- 単一ColorMatchオブジェクトからの複数検出を使用した作業

#### Step 1: 新規ColorMatch オブジェクトの作成

- (1) Vision Guide ツールバー - <全ツール> -  <Color Match>ボタンをクリックしてください。
- (2) マウスカーソルが ColorMatch アイコンに変わります。
- (3) マウスカーソルを Vision Guide ウィンドウの画像表示部まで動かし、マウス左ボタンをクリックします。画像表示部に ColorMatch オブジェクトが置かれます。
- (4) オブジェクト名は自動的に作成されますので注意してください。ここでは、このシーケンスで最初に作成された ColorMatch オブジェクトなので、“ColorMatch01”というオブジェクト名になります。(オブジェクト名の変更方法については後述します。)

#### Step 2: サーチウィンドウの位置とサイズの設定

画面上には、下図のような ColorMatch オブジェクトが表示されます。



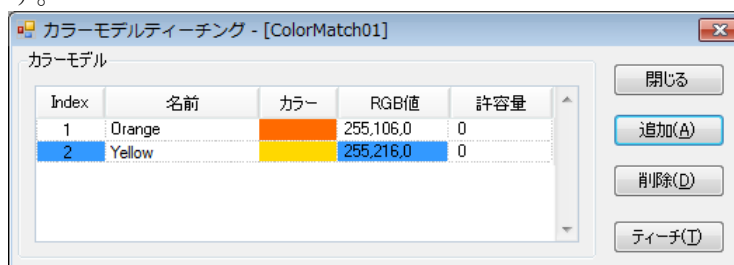
- (1) オブジェクト名をクリックし、マウスボタンを押し続けながら ColorMatch オブジェクトをドラッグし、サーチウィンドウを好きな位置に合わせてください。
- (2) サイズハンドルをクリックし、マウスをドラッグすると、ColorMatch オブジェクトサーチウィンドウのサイズが変更できます。(サーチウィンドウは、色の被検索領域です。)



## Step 3: ColorMatchオブジェクトのモデルの登録

ColorMatchオブジェクトを使用するには、まず検索するためのカラーモデルを登録します。カラーモデルを登録すると、ColorMatchオブジェクトウィンドウ内の全画素から平均色が決定されます。モデル名を自由に付けることができます。

- (1) ColorMatch オブジェクトが、現在表示されているオブジェクトであることを確認します。フローチャート、またはシーケンスツリーで、どのオブジェクトが現在作業中になっているかを確認できます。また、画像表示部にマゼンタ色で強調表示されているオブジェクトが現在使用しているオブジェクトです。
- (2) 実行パネルにある、<ティーチ>ボタンをクリックしてください。下の画面が表示されます。



ティーチウィンドウが表示されているあいだ、カラーモデルを登録するため必要に応じてColorMatchオブジェクトの位置を変更することができます。

- (3) ColorMatch オブジェクトを登録したい色の上へ置きます。このとき、登録する色をウィンドウ全体に表示させてください。
- (4) <追加>ボタンをクリックし、新しいカラーモデルを追加します。
- (5) モデルの列をクリックし、登録したいモデルを選択します。
- (6) <ティーチ>ボタンをクリックし、色を登録します。
- (7) 色に名前を付けます。この名前は ColorName リザルトとして使用しますので、色が分かるように名前を付けてください。
- (8) 許容量のデフォルトは、ColorMode が RGB の場合は 0、HSV の場合は 0, 0, 50 です。許容値を変更することで、色の変化が少ない場合や照明が一定でない場合の色の一致に役立ちます。
- (9) さらにカラーモデルを登録するには、上の手順 (3)から (8)を繰り返してください。



通常、ColorMatchオブジェクトウィンドウを使用してカラーモデルを登録しますが、ティーチウィンドウでRBG (またはHSV)値を直接入力してカラーモデルを登録することもできます。

### 6.2.10 LineFinder オブジェクト (直線検出)

#### LineFinderオブジェクトの概要

LineFinderオブジェクトは、画像イメージ中のラインの場所を特定するのに使われます。

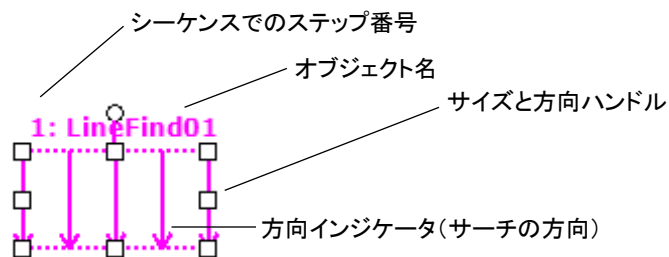
LineFinderオブジェクトは、複数のEdgeオブジェクトを自動的に処理してエッジ位置を特定し、各エッジ位置から特定される直線を求めます。

画像イメージ中のオブジェクトのエッジは、グレースケール値の暗から明への変化、あるいは明から暗への変化で表されます。このような変化は、数個のピクセルにわたって観察されます。

LineFinderオブジェクトでは、Polarityプロパティによって定義された、明から暗への変化、または暗から明への変化を検出し、検出された位置をシングルエッジのエッジ位置として定義します。EdgeTypeプロパティを変更してエッジペアを検出することもできます。エッジペアの場合、2つの対極のエッジを検索し、中間点を返します。

#### LineFinderオブジェクトのレイアウト

LineFinderオブジェクトは、CorrelationオブジェクトやBlobオブジェクトとは外観が異なります。LineFinderオブジェクトのサーチウィンドウは、Edgeオブジェクトに沿ったラインに相当します。LineFinderオブジェクトは、方向インジケータが示す方向のこのラインに沿って、変化(明から暗へ、または暗から明へ)を検索します。



#### LineFinder オブジェクトのレイアウト

LineFinderオブジェクトは、(垂直方向、水平方向だけでなく)あらゆる方向を検索するように置くことができます。BlobオブジェクトのSearchWinType=AngledRectangleと同様に、LineFinderオブジェクトのサーチウィンドウ回転ハンドルを使って、LineFinderオブジェクトを目的のエッジを検出する方向に動かすことができます。

## LineFinderオブジェクトのプロパティ

下の一覧は、LineFinderオブジェクトプロパティの概略です。各プロパティの詳細は、Vision Guide 7.0プロパティ&リザルトリファレンスを参照してください。

プロパティ	説明
AbortSeqOnFail	オブジェクトの実行に失敗した (不良判定になった)とき、直ちにシーケンス全体を中断し、それ以降のオブジェクトを処理しないように指定 デフォルト: False
Accept	特徴が検出されたと判定するための形状スコアの値を指定します。対象物が検出されたとみなされるためには、スコアの値を超えなくてはなりません。 デフォルト: 100
AngleBase	基準角度を設定 デフォルト: 0
AngleMode	角度の出力形式を指定 デフォルト: 1-Default
AngleStart	角度サーチの中心を指定
Caption	見出しをLineFinderオブジェクトに割りあてる デフォルト: 空欄
CenterPntObjResult	CenterPointObjectプロパティから使用するリザルトを指定 Allを指定すると、指定されたビジョンオブジェクトリザルトのすべて (NumberFound)にLineFinderオブジェクトが適用されます。
CenterPntOffsetX	サーチウィンドウの中心をCenterPointObjectプロパティで位置決めしたあとに、Xオフセットを設定、または返す
CenterPntOffsetY	サーチウィンドウの中心をCenterPointObjectプロパティで位置決めしたあとに、Yオフセットを設定、または返す
CenterPntRotOffset	中心点のXYオフセット値 (CenterPntOffsetX, CenterPntOffsetY)を、CenterPointObjectのAngleリザルトにしたがって回転するか否かを指定  SearchWinTypeがRotatedRectangleに設定されている場合、サーチウィンドウはAngleリザルトにしたがって回転します。
CenterPointObject	オブジェクトの中心点として用いられる位置を指定 このプロパティを“Screen”に設定すると画面上の任意の位置にオブジェクトを配置できますが、他のビジョンオブジェクトを指定した場合、そのオブジェクトのPixelX, PixelYリザルトに中心点が設定されます。
CheckClearanceFor	クリアランス (隙間)を確認するオブジェクトを設定
ClearanceCondition	クリアランス (隙間)の判断方法を指定
ContrastTarget	エッジサーチのためのコントラストを設定 デフォルト: 0 (best contrast)

プロパティ	説明
ContrastVariation	ContrastTargetの許容されるコントラストの変化量を選択 デフォルト: 0
CurrentResult	オブジェクトウィンドウのリザルトリストに表示するリザルト、または、1つのサーチウィンドウ内にある対象物などの複数のリザルトを、システムが検出する設定のときにデータを返すリザルトを定義
Description	ユーザー指定の説明を設定 デフォルト: 空欄
Directed	線の方向を使って角度を設定するかを指定 デフォルト: True
EdgeSort	検出されたエッジ結果のソート方法を設定
EdgeThreshold	この値より低いエッジは無視するという、しきい値を設定 デフォルト: 2
EdgeType	検索するエッジタイプ (シングル、またはペア)を設定 デフォルト: 1 – Single
Enabled	このオブジェクトを実行するかを指定 デフォルト: True
FailColor	オブジェクトの検出結果が不良時の色を指定 デフォルト: Red
FittingThreshold	直線のフィッティングに使用するエッジリザルトを指定
Frame	どの位置に置いたフレームを使用するかを指定 デフォルト: none
FrameResult	Frameの何番目のリザルトを使用するかを指定 デフォルト: 1
Graphics	表示するグラフィックを指定 デフォルト: 1 – All
LabelBackColor	オブジェクトラベルの背景の色を設定 デフォルト: Transparent (透明)
Name	LineFinderオブジェクトに独自のオブジェクト名を割りあてる デフォルト: LineFind01
NumberOfEdges	検出するエッジ数を設定 デフォルト: 5
PassColor	オブジェクトの検出良時の色を選択 デフォルト: LightGreen
PassType	オブジェクト検出時の良否判定の条件を定義 デフォルト: SomeFound
Polarity	LineFinderオブジェクトが明から暗への移行を検索するのか、暗から明への移行を検索するかを指定 デフォルト: 1 – LightToDark

プロパティ	説明
ScoreWeightContrast	コントラストによってスコアの割合を設定 デフォルト: 50
ScoreWeightStrength	エッジの強度によってスコアの割合を設定 デフォルト: 50
SearchWidth	エッジサーチの幅を定義 範囲: 3 ~ 99 デフォルト: 3
SearchWin	実行時のみ 1コールでサーチウィンドウの左端, 上端, 高さ, 幅のパラメータを設定、または返す
SearchWinAngle	被検索領域の角度を定義
SearchWinCenterX	被検索領域の中心のX座標値を定義
SearchWinCenterY	被検索領域の中心のY座標値を定義
SearchWinHeight	被検索領域の高さを定義 (単位: ピクセル)
SearchWinLeft	被検索領域の最左端を定義 (単位: ピクセル)
SearchWinTop	被検索領域の最上端を定義 (単位: ピクセル)
SearchWinWidth	被検索領域の幅を定義 (単位: ピクセル)
StrengthTarget	サーチするエッジの強度を設定 デフォルト: 0
StrengthVariation	StrengthTargetの変化量の合計を設定 デフォルト: 0
X1	エッジの始点位置のX座標
X2	エッジの終点位置のX座標
Y1	エッジの始点位置のY座標
Y2	エッジの終点位置のY座標

## LineFinderオブジェクトのリザルト

下の一覧は、Edgeオブジェクトリザルトを簡単に説明した概略です。各リザルトの詳細は、Vision Guide 7.0プロパティ&リザルトリファレンスを参照してください。

リザルト	説明
Angle	イメージ座標系における検出されたエッジの角度を返す
CameraX1	検出エッジラインの始点X座標をカメラ座標で返す
CameraY1	検出エッジラインの始点Y座標をカメラ座標で返す
CameraX2	検出エッジラインの終点X座標をカメラ座標で返す
CameraY2	検出エッジラインの終点Y座標をカメラ座標で返す
ClearanceOK	クリアランス(隙間)の判断結果を返す
Contrast	検出エッジの平均コントラストを返す
EdgeCameraXYU	検索中に検出したエッジの CameraX, CameraY, Angle 位置座標を返す
EdgePixelXYU	検索中に検出したエッジの PixelX, PixelY, Angle 位置座標を返す
EdgeRobotXYU	検索中に検出したエッジの RobotX, RobotY, Angle 位置座標を返す
FitError	各エッジポイントと検出された直線との間の距離を二乗平均平方根 (RMS)で返す
Found	オブジェクトが検出されたかどうか、対象物あるいはパーツが Acceptプロパティの現在の設定を超える形状スコア値を持っているかどうかの値を返す
Passed	検出結果が良か否かを返す
PixelLine	実行時のみ X1, Y1, X2, Y2の4つの直線座標をピクセル単位で返す
Length	検出エッジラインの長さをカメラ座標(mm)返す
NumberFound	検出された直線の数 を返す (最小0個から、最大はNumberToFindプロパティで設定した検出個数になります。)
PixelLength	検出エッジラインの長さ(ピクセル長)を返す
PixelX1	検出エッジラインの始点X座標をイメージ座標で返す
PixelY1	検出エッジラインの始点Y座標をイメージ座標で返す
PixelX2	検出エッジラインの終点X座標をイメージ座標で返す
PixelY2	検出エッジラインの終点Y座標をイメージ座標で返す
RobotX1	検出エッジラインの始点X座標をロボット座標で返す
RobotY1	検出エッジラインの始点Y座標をロボット座標で返す
RobotX2	検出エッジラインの終点X座標をロボット座標で返す
RobotY2	検出エッジラインの終点Y座標をロボット座標で返す
RobotU	検出エッジラインの角度をロボット座標で返す


リザルト	説明
Strength	検出エッジの平均強度を返す
Time	オブジェクト処理に要した時間を返す (単位: ミリ秒)

### LineFinderオブジェクトの使用方法

ここからは、LineFinderオブジェクトの作成と使用の方法を説明します。

- 新規LineFinderオブジェクトの作成方法
- サーチウィンドウの位置とサイズの設定
- LineFinderオブジェクトの関連プロパティの設定
- LineFinderオブジェクトのテストとリザルトの確認
- プロパティの調整と再テスト


これらのステップを開始する前に、まず新規ビジョンシーケンスを作成するか、使用する

ビジョンシーケンスを選択してください。作業するビジョンシーケンスがない場合は、 <新規シーケンス>ボタンをクリックすると、新規ビジョンシーケンスが作成できます。

すでにビジョンシーケンスを作成済みの場合は、Vision Guideウィンドウのシーケンスツリーをクリックして、作成済みのビジョンシーケンスを選択してください。

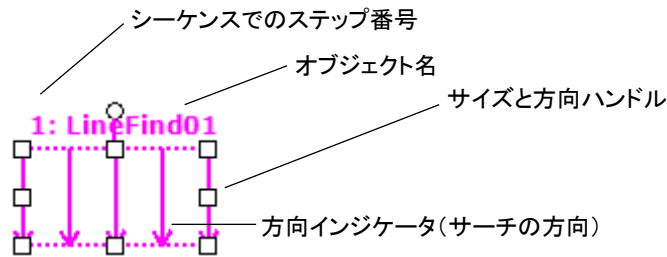
新規ビジョンシーケンスの作成方法、または作成済みビジョンシーケンスの選択方法の詳細は、「5. ビジョンシーケンス」を参照してください。

#### Step 1: 新規LineFinderオブジェクトの作成

- (1) Vision Guideツールバー - <全ツール> -  <Line Finder (直線検出)>ボタンをクリックしてください。
- (2) 画像イメージ表示部にマウスを動かしてください。マウスポインターがLineFinderオブジェクトアイコンに変わります。
- (3) アイコンが画像イメージ表示部の目的の位置にくるまで、マウスを動かし続けてください。それから、マウスを左クリックして、オブジェクトを作成してください。
- (4) オブジェクト名が自動的に作成されます。この例では、このシーケンスで最初に作成されたLineFinderオブジェクトなので、“LineFind01”というオブジェクト名になります。(オブジェクト名の変更方法については後述します。)

## Step 2: サーチウィンドウの位置とサイズの設定

画面上には、下図のようなLineFinderオブジェクトが表示されます。



新規LineFinder オブジェクトのレイアウト

- (1) LineFinderオブジェクトの名称ラベルをクリックし、マウスを押し続けながら、LineFinderオブジェクトをドラッグして、サーチウィンドウの左上コーナーを好きな位置に合わせてください。
- (2) サイズハンドルをクリックして、マウスをドラッグすると、LineFinderオブジェクトサーチウィンドウのサイズが変更できますので、適当な大きさに調整してください。

## Step 3: LineFinderオブジェクトプロパティーの設定

LineFinderオブジェクトプロパティーの設定を行います。プロパティーを設定するには、関連するプロパティーの設定値フィールドをクリックし、新規の値を入力してください。あるいは、ドロップダウンリストが表示されている場合は、リスト中の該当項目をクリックしてください。

次に、LineFinderオブジェクトで一般的に使われるプロパティーのいくつかを示します。その他、さまざまなビジョンオブジェクトの多くで使われるAbortSeqOnFail、Graphicsなどのプロパティーの詳細は、「Vision Guide 7.0プロパティー&リザルトリファレンス」を参照してください。角度の出力形式は、AngleModeプロパティーで設定します。詳細は、「Lineオブジェクト」を参照してください。

## EdgeType(シングル) 検索するエッジの種類を選択

エッジペアを設定した場合、それぞれの方向のエッジを検索し、中間点を位置として検出します。

Nameプロパティー  
("LineFindxx")

新規に作成されたLineFinderオブジェクトに、デフォルトとして与えられるオブジェクト名は“LineFindxx”です。xxは、1つのビジョンシーケンス内で複数のLineFinderオブジェクトを用いるとき、それらのLineFinderオブジェクトで区別するための番号です。ビジョンシーケンスで1番目に用いられるLineFinderオブジェクトには、“LineFind01”というオブジェクト名がデフォルトで設定されます。オブジェクト名を変更したい場合は、Nameプロパティーの設定値フィールドをクリックして、変更後の名称を入力し、リターンキーを押してください。画面上でLineFinderオブジェクト名を表示する場所は、すべて変更後の名称に変わります。

## NumberOfEdges(1) エッジ検索ラインに沿って1つ以上のエッジを検索

Polarity  
(LightToDark)

「暗から明へ(DarkToLight)」のエッジを検索する場合は、Polarityを変更してください。



**Step 4: LineFinderオブジェクトの実行とリザルトの確認**

LineFinderオブジェクトを実行するには、次のようにしてください。

実行パネルのオブジェクトの<Run>ボタンをクリックしてください。LineFinderオブジェクトのリザルトが表示されます。この時点で確認すべき主要なリザルトは次のとおりです。

Angleリザルト	検出されたエッジラインの角度をイメージ座標で返す
Lengthリザルト	検出されたエッジラインの長さをカメラ座標で返す 単位: mm
PixelLengthリザルト	検出されたエッジラインの長さをイメージ座標で返す 単位: ピクセル
PixelX1リザルト PixelY1リザルト	検出されたエッジラインの始点XY座標位置をイメージ座標で返す
PixelX2リザルト PixelY2リザルト	検出されたエッジラインの終点XY座標位置をイメージ座標で返す
CameraX1リザルト CameraY1リザルト	検出されたエッジラインの始点XY座標位置をカメラ座標で返す もしキャリブレーションされていないならば、"no cal"が返されます。
CameraX2リザルト CameraY2リザルト	検出されたエッジラインの終点XY座標位置をカメラ座標で返す もしキャリブレーションされていないならば、"no cal"が返されます。
RobotX1リザルト RobotY1リザルト	検出されたエッジラインの始点XY座標位置をロボット座標で返す もしキャリブレーションされていないならば、"no cal"が返されます。
RobotX2リザルト RobotY2リザルト	検出されたエッジラインの終点XY座標位置をロボット座標で返す もしキャリブレーションされていないならば、"no cal"が返されます。
RobotUリザルト	検出されたエッジラインの角度をロボット座標で返す

### 6.2.11 LineInspector オブジェクト (直線検査)

#### LineInspectorオブジェクトの概要

LineInspectorオブジェクトは、画像イメージ中の直線の検査に使われます。

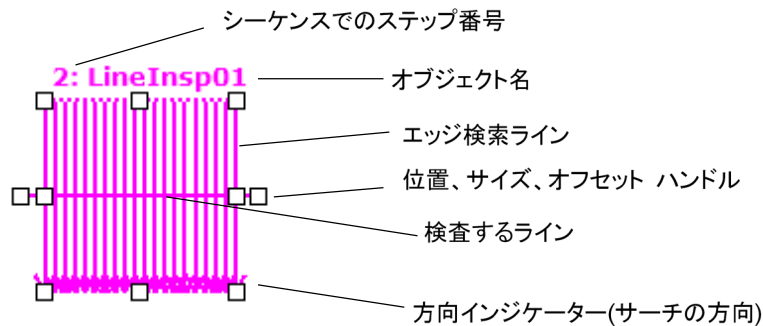
LineInspectorオブジェクトは、複数のEdgeオブジェクトを自動的に処理して検査する直線上の欠陥を特定します。

画像イメージ中のオブジェクトのエッジは、グレースケール値の暗から明への変化、あるいは明から暗への変化で表されます。このような変化は、数個のピクセルにわたって観察されます。

LineInspector オブジェクトでは、Polarity プロパティーによって定義された、明から暗への変化、または暗から明への変化を検出し、検出された位置をシングルエッジのエッジ位置として定義します。EdgeType プロパティーを変更してエッジペアを検出することもできます。エッジペアの場合、2つの対極のエッジを検索し、中間点を返します。

#### LineInspectorオブジェクトのレイアウト

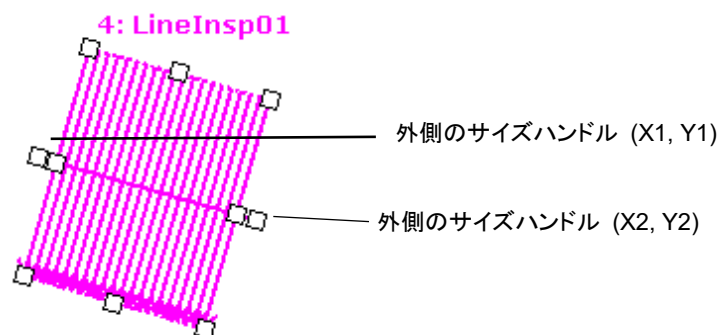
LineInspector オブジェクトは LineFinder ツールに似た形をしています。LineInspector オブジェクトのサーチウィンドウには複数のエッジ検索ラインがあります。LineInspector オブジェクトは方向インジケータが示す方向のこのラインに沿って、変化 (明から暗へ、または暗から明へ)を検索します。エッジ検索によるデータは、ライン周辺の欠陥の特定に使われます。



#### LineInspectorオブジェクトのレイアウト

LineInspectorオブジェクトは、(垂直方向、水平方向だけでなく)あらゆる方向を検索するように置くことができます。

オブジェクトを回転するには、外側のサイズハンドルを時計回りや反時計回りにドラッグします。検査する直線にLineFinderの結果を使用できます。この場合、LineObjectに指定するLineFinderを設定します。



検査する直線の幅を変更するには、外側のサイズハンドルをラインの中心から離したり近づけたりします。

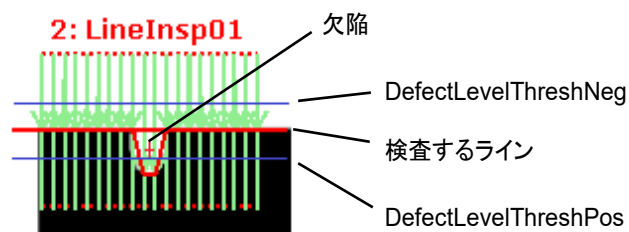
エッジ検索のサイズを変更するには、内側のサイズハンドルをドラッグします。

### LineInspectorサーチ

下図は欠陥のあるオブジェクトの一部です。



LineInspectorは下図のように欠陥を検出します。検査するラインからのそれぞれのエッジ検索位置はDefectLevelThreshPosまたはDefectLevelThreshNegプロパティの値を超えていなければなりません。また、欠陥領域はMinArea以上、MaxArea未満でなければなりません。



## LineInspectorオブジェクトのプロパティ

下の一覧は、LineInspectorオブジェクトプロパティの概略です。各プロパティの詳細は、「Vision Guide 7.0プロパティ&リザルトリファレンス」を参照してください。

プロパティ	説明
AbortSeqOnFail	オブジェクトの実行に失敗した (不良判定になった)とき、直ちにシーケンス全体を中断し、それ以降のオブジェクトを処理しないように指定 デフォルト: False
Accept	特徴が検出されたと判定するための形状スコアの値を指定します。対象物が検出されたとみなされるためには、スコアの値を超えなくてはなりません。 デフォルト: 100
CalRobotPlacePos	設計時とプログラム実行時に RobotPlacePos をキャリブレーションする
Caption	見出しをLineInspectorオブジェクトに割りあてる デフォルト: 空欄
ContrastTarget	エッジサーチのためのコントラストを設定 デフォルト: 0 (best contrast)
ContrastVariation	ContrastTargetの許容されるコントラストの変化量を選択 デフォルト: 0
CoordObject	リザルトをコピーするCoordinatesオブジェクトを指定 コピー処理はオブジェクトが実行される時に行われ、Decisionの分岐機能などで実行されなかった場合、コピー処理は行われません デフォルト: None
CurrentResult	オブジェクトウィンドウのリザルトリストに表示するリザルト、または、1つのサーチウィンドウ内にある対象物などの複数のリザルトを、システムが検出する設定のときにデータを返すリザルトを定義
DefectAreaExtended	欠陥領域補間を拡張するかを設定 デフォルト: False
DefectLevelThreshNeg	直線より下にある欠陥のしきい値を設定 デフォルト: 2
DefectLevelThreshPos	直線より上にある欠陥のしきい値を設定 デフォルト: 2
Description	ユーザー指定の説明を設定 デフォルト: 空欄
EdgeSort	検出されたエッジ結果のソート方法を設定
EdgeThreshold	この値より低いエッジは無視するという、しきい値を設定 デフォルト: 2

プロパティ	説明
EdgeType	検索するエッジタイプ (シングル、またはペア)を設定 デフォルト: 1 - single
Enabled	このオブジェクトを実行するかを指定 デフォルト: True
EndPntObjResult	どのリザルトをEndPointObjectから用いるのかを特定 デフォルト: 1
EndPointObject	どのビジョンオブジェクトを使って、直線の終点を定義するかを特定 デフォルト: Screen
EndPointType	どのタイプの終点を使って直線の終点を定義するかを特定 デフォルト: 0 - Point
FailColor	オブジェクトの検出結果が不良時の色を指定します。 デフォルト: Red
Frame	どの位置に置いたフレームを使用するかを指定 デフォルト: none
FrameResult	Frameの何番目のリザルトを使用するかを指定 デフォルト: 1
Graphics	表示するグラフィックを指定 デフォルト: 1 - All
InspectEndOffset	検査を終了するラインの終点からオフセットを設定 デフォルト: 15
InspectStartOffset	検査を開始するラインの始点からオフセットを設定 デフォルト: 15
LabelBackColor	オブジェクトラベルの背景の色を設定 デフォルト: Transparent(透明)
LineObject	検査前にラインを検索するオブジェクトを定義 デフォルト: None
LineObjResult	どのリザルトをLineObjectから用いるのかを特定 デフォルト: 1
MaxArea	オブジェクトの上限領域を規定 被検出ブローブが取得するAreaリザルトは、MaxAreaプロパティに設定された値より低くならないでなりません。 デフォルト: 100,000
MinArea	オブジェクトの下限領域を規定 被検出ブローブが取得するAreaリザルトは、MinAreaプロパティに設定された値より高くないでなりません。 デフォルト: 25

プロパティ	説明
MissingEdgeType	未検出のエッジの処理方法を定義 デフォルト: Interpolate
Name	LineInspectorオブジェクトに独自のオブジェクト名を割りあて る デフォルト: LineInsp01
NumberOfEdges	検出するエッジ数を設定 デフォルト: 20
NumberToFind	作業中のサーチウィンドウで検出する対象物の数を定義 デフォルト: 1
PassColor	オブジェクトの検出良時の色を選択 デフォルト: LightGreen
PassType	オブジェクト検出時の良否判定の条件を定義 デフォルト: AllNotFound
Polarity	LineInspectorオブジェクトが明から暗への移行を検索するの か、暗から明への移行を検索するのかを指定 デフォルト: 1 – LightToDark
ScoreWeightContrast	コントラストによるスコアの割合を設定 デフォルト: 50
ScoreWeightStrength	エッジの強度によるスコアの割合を設定 デフォルト: 50
SearchWidth	エッジサーチの幅を定義 範囲: 3 ~ 99 デフォルト: 3
SearchWinHeight	被検索領域の高さを定義 (単位: ピクセル)
SearchWinLeft	被検索領域の最左端を定義 (単位: ピクセル)
SearchWinTop	被検索領域の最上端を定義 (単位: ピクセル)
SearchWinWidth	被検索領域の幅を定義 (単位: ピクセル)
SizeToFind	検出する欠陥のサイズを選択 デフォルト: 1 - Largest
StartPntObjResult	どのリザルトをStartPointObjectから用いるのかを特定 デフォルト: 1
StartPointObject	どのビジョンオブジェクトを使って、直線の始点を定義する かを指定 デフォルト: Screen
StartPointType	どのタイプの始点を使って直線の始点を定義するのかを指定 デフォルト: “0 – Point”
StrengthTarget	サーチするエッジの強度を設定 デフォルト: 0

プロパティ	説明
StrengthVariation	StrengthTargetの変化量の合計を設定 デフォルト: 0
X1	エッジの始点位置のX座標
X2	エッジの終点位置のX座標
Y1	エッジの始点位置のY座標
Y2	エッジの終点位置のY座標

### LineInspectorオブジェクトのリザルト

下の一覧は、LineInspectorオブジェクトリザルトを簡単に説明した概略です。各リザルトの詳細は、Vision Guide 7.0プロパティ&リザルトリファレンスを参照してください。


リザルト	説明
Area	欠陥の面積をピクセルで返す
CameraX	カメラ座標系における欠陥のX座標位置を返す
CameraY	カメラ座標系における欠陥のY座標位置を返す
Contrast	検出エッジの平均コントラストを返す
DefectLevel	欠陥の高さを返す
Found	オブジェクトが検出されたかどうか、対象物あるいはパーツがAcceptプロパティの現在の設定を超える形状スコア値を持っているかどうかの値を返す
Length	欠陥の長さをmm単位で返す
Passed	検出結果が良か否かを返す
PixelLength	欠陥の長さをピクセル単位で返す
NumberFound	検出された欠陥の数を返す (最小0個から、最大はNumberToFindプロパティで設定した検出個数になります。)
PixelX	欠陥のX座標位置を返す
PixelY	欠陥のY座標位置を返す
PixelXYU	実行時のみ 検出欠陥位置のPixelX座標, PixelY座標, PixelU座標をピクセルで返す
RobotX	ロボット座標系における欠陥のX座標値を返す
RobotY	ロボット座標系における欠陥のY座標値を返す
RobotU	ロボット座標系における欠陥のU座標値を返す
RobotXYU	実行時のみ ロボット座標系における、検出欠陥位置のRobotX座標, RobotY座標, RobotU座標を返す
Strength	検出エッジの平均強度を返す
Time	オブジェクト処理に要した時間を返す (単位: ミリ秒)
TotalArea	検出した欠陥すべての領域の合計をピクセル単位で返す

### LineInspectorオブジェクトの使用方法

ここからは、LineInspectorオブジェクトの作成と使用の方法を説明します。

- 新規LineInspectorオブジェクトの作成方法
- サーチウィンドウの位置とサイズの設定
- LineInspectorオブジェクトの関連プロパティの設定
- LineInspectorオブジェクトのテストとリザルトの確認
- プロパティの調整と再テスト


これらのステップを開始する前に、まず新規ビジョンシーケンスを作成するか、使用する

ビジョンシーケンスを選択してください。作業するビジョンシーケンスがない場合は、 <新規シーケンス>ボタンをクリックすると、新規ビジョンシーケンスが作成できます。

すでにビジョンシーケンスを作成済みの場合は、Vision Guideウィンドウのシーケンスツリーをクリックして、作成済みのビジョンシーケンスを選択してください。

新規ビジョンシーケンスの作成方法、または作成済みビジョンシーケンスの選択方法の詳細は、「5. ビジョンシーケンス」を参照してください。

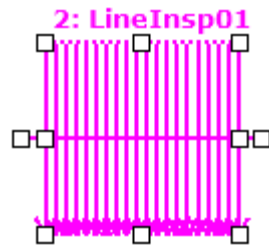
### Step 1: 新規LineInspectorオブジェクトの作成

- (1) Vision Guideツールバー - <全ツール> -  <Line Inspector (直線検査)>ボタンをクリックしてください。
- (2) 画像イメージ表示部にマウスを動かしてください。マウスポインターがLineInspectorオブジェクトアイコンに変わります。
- (3) アイコンが画像イメージ表示部の目的の位置にくるまで、マウスを動かして続けてください。それから、マウスを左クリックして、オブジェクトを作成してください。
- (4) オブジェクト名が自動的に作成されます。この例では、このシーケンスで最初に作成されたLineInspectorオブジェクトなので、“LineInsp01”というオブジェクト名になります。(オブジェクト名の変更方法については後述します。)



## Step 2: サーチウィンドウの位置とサイズの設定

画面上には、下図のようなLineInspectorオブジェクトが表示されます。



新規LineInspector オブジェクトのレイアウト

1. LineInspectorオブジェクトの名称ラベルをクリックし、マウスを押し続けながら、LineInspectorオブジェクトをドラッグして、サーチウィンドウの左上コーナーを好きな位置に合わせてください。
2. サイズハンドルをクリックして、マウスをドラッグすると、LineInspectorオブジェクトサーチウィンドウのサイズが変更できますので、適当な大きさに調整してください。

## Step 3: LineInspectorオブジェクトプロパティの設定

LineInspectorオブジェクトプロパティの設定を行います。プロパティを設定するには、関連するプロパティの設定値フィールドをクリックし、新規の値を入力してください。あるいは、ドロップダウンリストが表示されている場合は、リスト中の該当項目をクリックしてください。

次に、LineInspector オブジェクトで一般的に使われるプロパティのいくつかを示します。その他、さまざまなビジョンオブジェクトの多くで使われる AbortSeqOnFail, Graphics などのプロパティの詳細は、「Vision Guide 7.0 プロパティ&リザルトリファレンス」を参照してください。

EdgeType(シングル)	検索するエッジの種類を選択 エッジペアを設定した場合、それぞれの方向のエッジを検索し、中間点を位置として検出します。
Nameプロパティ ("LineInspxx")	新規に作成されたLineInspectorオブジェクトに、デフォルトとして与えられるオブジェクト名は"LineInspxx"です。xxは、1つのビジョンシーケンス内で複数のLineInspectorオブジェクトを用いるとき、それらのLineInspectorオブジェクトで区別するための番号です。ビジョンシーケンスで1番目に用いられるLineInspectorオブジェクトには、“LineInsp01”というオブジェクト名がデフォルトで設定されます。オブジェクト名を変更したい場合は、Nameプロパティの設定値フィールドをクリックして、変更後の名称を入力し、リターンキーを押してください。画面上でLineInspectorオブジェクト名を表示する場所は、すべて変更後の名称に変わります。
NumberToFind (1)	エッジ検索ラインに沿って1つ以上のエッジを検索
Polarity (LightToDark)	「暗から明へ(DarkToLight)」のエッジを検索する場合は、Polarityを変更してください。

### Step 4: LineInspectorオブジェクトの実行とリザルトの確認

LineInspectorオブジェクトを実行するには、次のようにしてください。

実行パネルのオブジェクトの<Run>ボタンをクリックしてください。LineInspectorオブジェクトのリザルトが表示されます。この時点で確認すべき主要なリザルトは次のとおりです。

Areaリザルト	検出された欠陥の面積をピクセル単位で返す
PixelXリザルト	欠陥のXY座標値 (ピクセル座標系)を返す
PixelYリザルト	
CameraXリザルト	カメラ座標系における欠陥のXY座標値を返す
CameraYリザルト	これらのリザルトは、カメラキャリブレーションが完了していない場合は“no cal”を返し、値を返しません。
RobotXリザルト	欠陥のXY座標位置をロボット座標で返す
RobotYリザルト	もしキャリブレーションされていなければ、“no cal”が返されます。

#### 6.2.12



#### ArcFinder オブジェクト (円弧/楕円弧検出)

##### ArcFinderオブジェクトの概要

ArcFinderオブジェクトは、画像イメージ中の円弧/楕円弧エッジの場所を特定するのに使われます。

ArcFinderオブジェクトは、複数のエッジ検索を実行し、円弧の場合は半径と中心点、楕円弧の場合は長径, 短径, および角度を特定し、円弧/楕円弧を検出します。

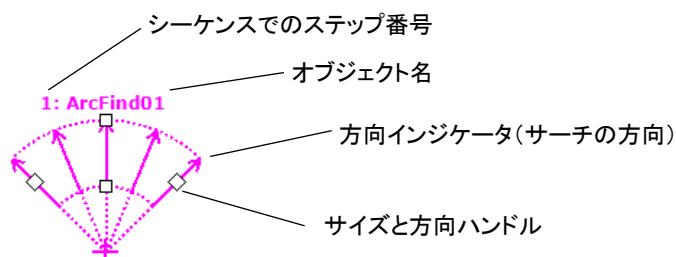
画像イメージ中のオブジェクトのエッジは、グレイスケール値の暗から明への変化、あるいは明から暗への変化で表されます。このような変化は、数個のピクセルにわたって観察されます。

ArcFinderオブジェクトの各エッジサーチは、Polarityプロパティによって定義された、明から暗への変化、または暗から明への変化を検出し、検出された位置をシングルエッジのエッジ位置として定義します。EdgeTypeプロパティを変更してエッジペアを検出することもできます。エッジペアの場合、2つの対極のエッジを検索し、中間点を返します。

検出する円弧/楕円弧タイプは、ArcSearchTypeプロパティによって指定します。

### ArcFinderオブジェクトのレイアウト

ArcFinderオブジェクトは、CorrelationオブジェクトやBlobオブジェクトと異なり、サーチウィンドウは円形であり、始点角、終点角、外半径、内半径の指定もあります。エッジ検索ラインは始点角と終点角の間に均等に広がっています。エッジ検索ラインの数はNumberOfEdgesプロパティで指定します。Directionプロパティで検索方向(内径から外径、または外径から内径)を指定できます。



### ArcFinder オブジェクトのレイアウト

ArcFinderオブジェクトは、検出する円弧のおおよその中心に中心点を移動し、円弧が検索エリア内に入るようにRadiusInnerプロパティとRadiusOuterプロパティを調整することで円弧を検索するように配置します。

### ArcFinderオブジェクトのプロパティ

下の一覧は、ArcFinderオブジェクトプロパティの概略です。各プロパティの詳細は、Vision Guide 7.0プロパティ&リザルトリファレンスを参照してください。

プロパティ	説明
AbortSeqOnFail	オブジェクトの実行に失敗した(不良判定になった)とき、直ちにシーケンス全体を中断し、それ以降のオブジェクトを処理しないように指定 デフォルト: False
Accept	特徴が検出されたと判定するための形状スコアの値を指定 対象物が検出されたとみなされるためには、スコアの値を超えなくてはなりません。 デフォルト: 100
AngleEnd	円弧/楕円弧をサーチする範囲の終了角度を指定 デフォルト: 135
AngleStart	円弧/楕円弧をサーチする範囲の開始角度を指定 デフォルト: 45
ArcSearchType	検出する円弧/楕円弧のタイプを指定
CalRobotPlacePos	設計時とプログラム実行時に RobotPlacePos をキャリブレーションする
Caption	見出しをArcFinderオブジェクトに割りあてる デフォルト: 空欄

プロパティ	説明
CenterPointObject	<p>オブジェクトの中心点として用いられる位置を指定</p> <p>このプロパティをScreenに設定すると画面上の任意の位置にオブジェクトを配置できますが、他のビジョンオブジェクトを指定した場合、そのオブジェクトのPixelX、PixelYリザルトに中心点が設定されます。</p> <p>デフォルト: Screen</p>
CenterPntObjResult	<p>CenterPointObjectプロパティから使用するリザルトを指定</p> <p>Allを指定すると、指定されたビジョンオブジェクトリザルトのすべて (NumberFound)にArcFinderオブジェクトが適用されます。</p> <p>デフォルト: 1</p>
CenterPntOffsetX	<p>サーチウィンドウの中心をCenterPointObjectプロパティで位置決めしたあとに、Xオフセットを設定、または返す</p> <p>デフォルト: 0</p>
CenterPntOffsetY	<p>サーチウィンドウの中心をCenterPointObjectプロパティで位置決めしたあとに、Yオフセットを設定、または返す</p> <p>デフォルト: 0</p>
CenterPntRotOffset	<p>中心点の XY オフセット値 (CenterPntOffsetX, CenterPntOffsetY)を、CenterPointObject の Angle リザルトにしたがって回転するか否かを指定</p> <p>デフォルト: False</p>
CenterX	<p>オブジェクトの中心位置のX座標を指定</p> <p>CenterPointObjectプロパティが他のビジョンオブジェクトに設定されると無効になり、自動的に設定されます。</p>
CenterY	<p>オブジェクトの中心位置のY座標を指定</p> <p>CenterPointObjectプロパティが他のビジョンオブジェクトに設定されると無効になり、自動的に設定されます。</p>
CheckClearanceFor ClearanceCondition	<p>クリアランス (隙間)を確認するオブジェクトを設定</p> <p>クリアランス (隙間)の判断方法を指定</p>
ContrastTarget	<p>エッジサーチのためのコントラストを設定</p> <p>デフォルト: 0 (best contrast)</p>
ContrastVariation	<p>ContrastTargetの許容されるコントラストの変化量を選択</p> <p>デフォルト: 0</p>

プロパティ	説明
CoordObject	リザルトをコピーするCoordinatesオブジェクトを指定 コピー処理はオブジェクトが実行される時に行われ、Decisionの分岐機能などで実行されなかった場合、コピー処理は行われません デフォルト: None
CurrentResult	オブジェクトウィンドウのリザルトリストに表示するリザルト、または、1つのサーチウィンドウ内にある対象物などの複数のリザルトを、システムが検出する設定のときにデータを返すリザルトを定義
Description	ユーザー指定の説明を設定 デフォルト: 空欄
Direction	エッジ検索の方向を設定 デフォルト: InsideOut
EdgeSort	検出されたエッジ結果のソート方法を設定
EdgeThreshold	この値より低いエッジは無視するという、しきい値を設定 デフォルト: 2
EdgeType	検索するエッジタイプ (シングル, ペア)を設定 デフォルト: 1 – Single
Enabled	このオブジェクトを実行するかを指定 デフォルト: True
FailColor	オブジェクトの検出結果が不良時の色を指定 デフォルト: Red
FittingThreshold	円弧のフィッティングに使用するエッジリザルトを指定
Frame	どの位置に置いたフレームを使用するかを指定 デフォルト: none
FrameResult	Frameの何番目のリザルトを使用するかを指定 デフォルト: 1
Graphics	表示するグラフィックを指定 デフォルト: 1 – All
LabelBackColor	オブジェクトラベルの背景の色を設定 デフォルト: Transparent (透明)
Name	ArcFinderオブジェクトに独自のオブジェクト名を割りあてる デフォルト: ArcFind01
NumberOfEdges	検出するエッジ数を設定 デフォルト: 5

プロパティ	説明
PassColor	オブジェクトの検出良時の色を選択 デフォルト: LightGreen
PassType	オブジェクト検出時の良否判定の条件を定義 デフォルト: SomeFound
Polarity	ArcFinderオブジェクトが明から暗へ (LightToDark)の移行を検索するのか、暗から明 へ(DarkToLight)の移行を検索するのかを指定 デフォルト: 1 – LightToDark
RadiusInner	検出範囲の内径を指定
RadiusOuter	検出範囲の外径を指定
ScoreWeightContrast	コントラストによってスコアの割合を設定 デフォルト: 50
ScoreWeightStrength	エッジの強度によってスコアの割合を設定 デフォルト: 50
SearchWidth	エッジサーチの幅を定義 範囲: 3 ~ 99 デフォルト: 3
ShowExtension	被検出エッジラインの両端を延長して表示するか を定義
StrengthTarget	サーチするエッジの強度を設定 デフォルト: 0
StrengthVariation	StrengthTargetの変化量の合計を設定 デフォルト: 0

### ArcFinderオブジェクトのリザルト

下の一覧は、ArcFinderオブジェクトリザルトを簡単に説明した概略です。各リザルトの詳細は、Vision Guide 7.0プロパティ&リザルトリファレンスを参照してください。

リザルト	説明
Angle	検出楕円弧の角度を返す
Angle1	イメージ座標系における検出エッジの始点角度を返す
Angle2	イメージ座標系における検出エッジの終点角度を返す
CameraX	検出円弧/楕円弧エッジの中心X座標をカメラ座標で返す
CameraY	検出円弧/楕円弧エッジの中心Y座標をカメラ座標で返す
CameraXYU	実行時のみ カメラ座標系における検出パーツ位置のCameraX座標, CameraY座標, CameraU座標を返す
ClearanceOK	クリアランス (隙間)の判断結果を返す
Contrast	検出円弧/楕円弧エッジのコントラストを返す
EdgeCameraXYU	検索中に検出したエッジの CameraX, CameraY, Angle 位置 座標を返す


リザルト	説明
EdgePixelXYU	検索中に検出したエッジの PixelX, PixelY, Angle 位置座標を返す
EdgeRobotXYU	検索中に検出したエッジの RobotX, RobotY, Angle 位置座標を返す
FitError	各エッジポイントと検出された円弧との間の距離を二乗平均平方根 (RMS)で返す
Found	オブジェクトが検出されたかどうか、つまり、対象物あるいはパーツがAcceptプロパティの現在の設定を超える形状スコア値を持っているかどうかの値を返す
FoundMajorDiam	検出楕円弧エッジの長径
FoundMinorDiam	検出楕円弧エッジの短径
FoundRadius	検出円弧エッジの半径
MaxError	検出円弧/楕円弧エッジからの最大ずれをピクセル長で返す
NumberFound	検出した円弧の数を返す
Passed	検出結果が良か否かを返す
PixelMajorDiam	ArcFinder で検出された楕円弧の長径長を返す
PixelMinorDiam	ArcFinder で検出された楕円弧の短径長を返す
PixelRadius	検出された円弧の半径を返す (単位: ピクセル)
PixelX	検出円弧/楕円弧エッジの中心X座標をイメージ座標で返す
PixelY	検出円弧/楕円弧エッジの中心Y座標をイメージ座標で返す
PixelXYU	実行時のみ 検出円弧/楕円弧エッジの中心位置のPixelX座標, PixelY座標, PixelU座標をピクセルで返す
RobotX	検出円弧/楕円弧エッジの中心X座標をロボット座標で返す
RobotY	検出円弧/楕円弧エッジの中心Y座標をロボット座標で返す
RobotXYU	ロボット座標系における、検出円弧/楕円弧エッジの中心位置のRobotX座標, RobotY座標, RobotU座標を返す
Strength	検出エッジの強度を返す
Time	オブジェクト処理に要した時間を返す (単位: ミリ秒)

### ArcFinderオブジェクトの使用方法

ここからは、ArcFinderオブジェクトの作成と使用の方法を説明します。

- 新規ArcFinderオブジェクトの作成方法
- サーチウィンドウの位置とサイズの設定
- ArcFinderオブジェクトの関連プロパティの設定
- ArcFinderオブジェクトのテストとリザルトの確認
- プロパティの調整と再テスト


これらのステップを開始する前に、まず新規ビジョンシーケンスを作成するか、使用する

ビジョンシーケンスを選択してください。作業するビジョンシーケンスがない場合は、 <新規シーケンス>ボタンをクリックすると、新規ビジョンシーケンスが作成できます。

すでにビジョンシーケンスを作成済みの場合は、Vision Guideウィンドウのシーケンスツリーをクリックして、作成済みのビジョンシーケンスを選択してください。

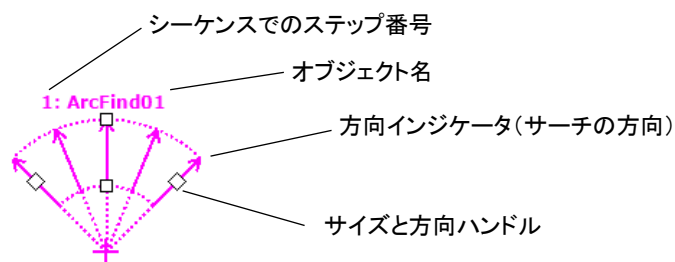
新規ビジョンシーケンスの作成方法、または作成済みビジョンシーケンスの選択方法の詳細は、「5. ビジョンシーケンス」を参照してください。

#### Step 1: 新規ArcFinderオブジェクトの作成

- (1) Vision Guideツールバー -  <Arc Finder (円弧検出)>ボタンをクリックしてください。
- (2) 画像イメージ表示部にマウスを動かしてください。マウスポインターがArcFinderオブジェクトアイコンに変わります。
- (3) アイコンが画像イメージ表示部の目的の位置に来るまで、マウスを動かしてください。それから、マウスを左クリックして、オブジェクトを作成してください。
- (4) オブジェクト名が自動的に作成されるのに注目してください。この例では、このシーケンスで最初に作成されたArcFinderオブジェクトなので、“ArcFind01”というオブジェクト名になります。(オブジェクト名の変更方法については後述します。)

#### Step 2: サーチウィンドウの位置とサイズの設定

画面上には、下図のようなArcFinderオブジェクトが表示されます。



新規ArcFinderオブジェクトのレイアウト

- (1) ArcFinderオブジェクトの名称ラベルをクリックし、マウスを押し続けながら、ArcFinderオブジェクトをドラッグして、サーチウィンドウを好きな位置に合わせてください。
- (2) サイズハンドルをクリックして、マウスをドラッグすると、ArcFinderオブジェクトサーチウィンドウのサイズが変更できますので、適当な大きさに調整してください。



### Step 3: ArcFinderオブジェクトプロパティの設定

ArcFinderオブジェクトプロパティの設定を行います。プロパティを設定するには、関連するプロパティの設定値フィールドをクリックし、新規の値を入力してください。あるいは、ドロップダウンリストが表示されている場合は、リスト中の該当項目をクリックしてください。

次に、ArcFinderオブジェクトで一般的に使われるプロパティのいくつかを示します。その他、さまざまなビジョンオブジェクトの多くで使われるAbortSeqOnFail, Graphicsなどのプロパティの詳細は、Vision Guide 7.0プロパティ&リザルトリファレンスを参照してください。

EdgeType(シングル)	<p>検索するエッジの種類を選択</p> <p>エッジペアを設定した場合、それぞれの方向のエッジを検索し、中間点を位置として検出します。</p>
Nameプロパティ ("ArcFindxx")	<p>新規に作成されたArcFinderオブジェクトに、デフォルトとして与えられるオブジェクト名は“ArcFindxx”です。xxは、1つのビジョンシーケンス内で複数のArcFinderオブジェクトを用いるとき、それらのArcFinderオブジェクトで区別するための番号です。ビジョンシーケンスで1番目に用いられるArcFinderオブジェクトには、“ArcFind01”というオブジェクト名がデフォルトで設定されます。オブジェクト名を変更したい場合は、Nameプロパティの設定値フィールドをクリックして、変更後の名称を入力し、リターンキーを押してください。画面上でArcFinderオブジェクト名を表示する場所は、すべて変更後の名称に変わります。</p>
NumberOfEdges(5)	円弧エッジを検索するために、5個のエッジを検索
Polarity (LightToDark)	<p>エッジ位置を「明から暗へ(LightToDark)」検索</p> <p>「暗から明へ(DarkToLight)」のエッジを検索する場合は、Polarityを変更してください。</p>

### Step 4: ArcFinderオブジェクトの実行とリザルトの確認

ArcFinderオブジェクトを実行するには、次のようにしてください。

実行パネルのオブジェクトの<Run>ボタンをクリックしてください。ArcFinderオブジェクトのリザルトが表示されます。この時点で確認すべき主要なリザルトは次のとおりです。

Angle1リザルト	検出された円弧エッジの始点角度をイメージ座標で返す
Angle2リザルト	検出された円弧エッジの終点角度をイメージ座標で返す
FoundRadiusリザルト	<p>検出された円弧エッジの半径をピクセル長で返す</p> <p>単位: ピクセル</p>
MaxErrorリザルト	<p>検出された円弧エッジからの最大ずれをピクセル長で返す</p> <p>単位: ピクセル</p>

## 6. ビジョンオブジェクト

---

PixelXリザルト PixelYリザルト	検出された円弧エッジの中心XY座標位置をイメージ座標で返す
CameraXリザルト CameraYリザルト	検出された円弧エッジの中心XY座標位置をカメラ座標で返す もしキャリブレーションされていなければ、“no cal”が返されます。
RobotXリザルト RobotYリザルト	検出された円弧エッジの中心XY座標位置をロボット座標で返す もしキャリブレーションされていなければ、“no cal”が返されます。

### 6.2.13 ArcInspectorオブジェクト (円弧/楕円弧検査)

#### ArcInspectorオブジェクトの概要

ArcInspectorオブジェクトは、円弧/楕円弧沿いの欠陥を検索するのに使われます。

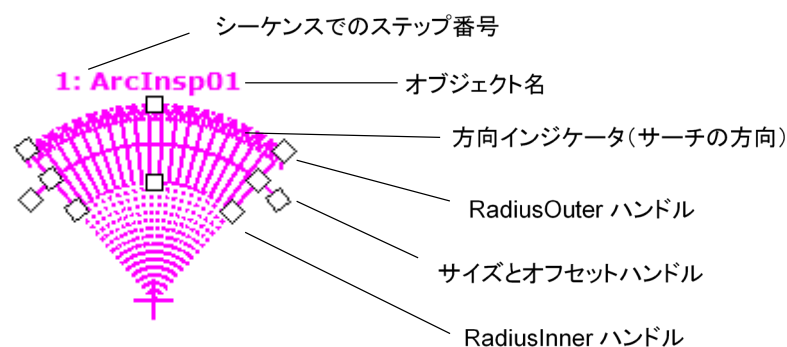
ArcInspectorオブジェクトは、複数のエッジ検索を実行して検査する円弧/楕円弧の異常を検出します。

画像イメージ中のオブジェクトのエッジは、グレースケール値の暗から明への変化、あるいは明から暗への変化で表されます。このような変化は、数個のピクセルにわたって観察されます。

ArcInspector オブジェクトの各エッジ検索では、Polarity プロパティによって定義された、明から暗への変化、または暗から明への変化を検出し、検出された位置をシングルエッジのエッジ位置として定義します。EdgeType プロパティを変更してエッジペアを検出することもできます。エッジペアの場合、2つの対極のエッジを検索し、中間点を返します。

#### ArcInspectorオブジェクトのレイアウト

ArcInspectorオブジェクトは、ArcFinderオブジェクトに似た形をしています。ArcInspectorオブジェクトは円弧/楕円弧の中心から検査領域の外半径に向かって、ラインに沿ってエッジを検索します。各エッジ検索ラインは方向インジケータが示す方向のこのラインに沿って、変化(明から暗へ、または暗から明へ)を検索します。検査に使用するエッジ数はNumberOfEdgesプロパティで設定します。AngleStartプロパティで検査する円弧の始点角を指定し、AngleEndプロパティで終点角を指定します。エッジ検索ラインは(AngleStart + InspectStartOffset)と(AngleEnd – InspectEndOffset)の間で均等に配置されます。



#### ArcInspector オブジェクトのレイアウト

ArcInspectorは、CenterXとCenterYを円弧の中心に配置し、RadiusInnerとRadiusOuterを調整して検査する円弧の半径が検索領域内に入るように検索領域を配置することで、円弧上の欠陥を検索します。ArcInspectorオブジェクトはDirectionの設定に従って、内半径から外径半径(デフォルト)、またはその逆の方向に向かって欠陥を検索します。

ArcFinderオブジェクトで円弧を検出し、ArcInspectorで検査することもできます。

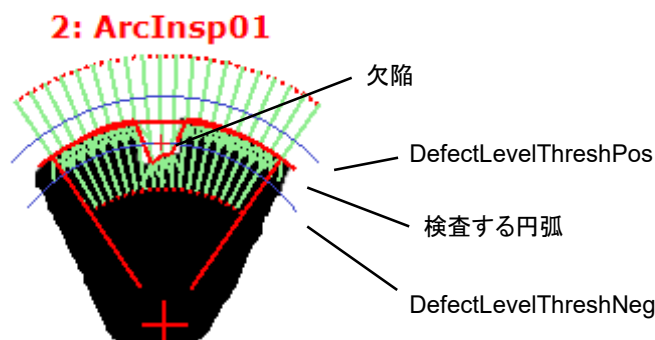
ArcObjectプロパティでArcFinderを使用するように指定します。

## ArcInspectorサーチ

下図は欠陥を持つ円形オブジェクトの一部です。



ArcInspectorは下図のように欠陥を検出します。検査する円弧からのそれぞれのエッジ検索位置はDefectLevelThreshPosまたはDefectLevelThreshNegプロパティの値を超えていなければなりません。また、欠陥領域はMinArea以上、MaxArea未満でなければなりません。



## ArcInspectorオブジェクトのプロパティ

下の一覧は、ArcInspectorオブジェクトプロパティの概略です。各プロパティの詳細は、Vision Guide 7.0プロパティ&リザルトリファレンスを参照してください。

プロパティ	説明
AbortSeqOnFail	オブジェクトの実行に失敗した (不良判定になった)とき、直ちにシーケンス全体を中断し、それ以降のオブジェクトを処理しないように指定 デフォルト: False
Accept	特徴が検出されたと判定するための形状スコアの値を指定 対象物が検出されたとみなされるためには、スコアの値を超えなくてはなりません。 デフォルト: 100
AngleEnd	円弧/楕円弧をサーチする範囲の終了角度を指定 デフォルト: 135
AngleStart	円弧/楕円弧をサーチする範囲の開始角度を指定 デフォルト: 45
ArcObject	検査する円弧/楕円弧を検索するArcFinderを指定 デフォルト: None
ArcObjResult	ArcObject プロパティから使用するリザルトを指定 デフォルト: 1
ArcSearchType	検査する円弧/楕円弧のタイプを指定 ArcObjectを指定する場合、指定されたArcFinderに設定されているArcTypeと合わせてください。

プロパティ	説明
CalRobotPlacePos	設計時とプログラム実行時に RobotPlacePos をキャリブレーションする
Caption	見出しをArcInspectorオブジェクトに割りあてる デフォルト: 空欄
CenterPointObject	オブジェクトの中心点として用いられる位置を指定 このプロパティをScreenに設定すると画面上の任意の位置にオブジェクトを配置できますが、他のビジョンオブジェクトを指定した場合、そのオブジェクトのPixelX、PixelYリザルトに中心点が設定されます。 デフォルト: Screen
CenterPntObjResult	CenterPointObjectプロパティから使用するリザルトを指定 Allを指定すると、指定されたビジョンオブジェクトリザルトのすべて (NumberFound)にArcInspectorオブジェクトが適用されます。 デフォルト: 1
CenterPntOffsetX	サーチウィンドウの中心をCenterPointObjectプロパティで位置決めしたあとに、Xオフセットを設定、または返す デフォルト: 0
CenterPntOffsetY	サーチウィンドウの中心をCenterPointObjectプロパティで位置決めしたあとに、Yオフセットを設定、または返す デフォルト: 0
CenterPntRotOffset	中心点の XY オフセット値 (CenterPntOffsetX, CenterPntOffsetY)を、CenterPointObject の Angle リザルトにしたがって回転するか否かを指定 デフォルト: False
CenterX	オブジェクトの中心位置のX座標を指定 CenterPointObjectプロパティが他のビジョンオブジェクトに設定されると無効になり、自動的に設定されます。
CenterY	オブジェクトの中心位置のY座標を指定 CenterPointObjectプロパティが他のビジョンオブジェクトに設定されると無効になり、自動的に設定されます。
ContrastTarget	エッジサーチのためのコントラストを設定 デフォルト: 0 (best contrast)
ContrastVariation	ContrastTargetの許容されるコントラストの変化量を選択 デフォルト: 0
CoordObject	リザルトをコピーするCoordinatesオブジェクトを指定 コピー処理はオブジェクトが実行される時に行われ、Decisionの分岐機能などで実行されなかった場合、コピー処理は行われません デフォルト: None

プロパティ	説明
CurrentResult	オブジェクトウィンドウのリザルトリストに表示するリザルト、または、1つのサーチウィンドウ内にある対象物などの複数のリザルトを、システムが検出する設定のときにデータを返すリザルトを定義
Description	ユーザー指定の説明を設定 デフォルト: 空欄
DefectAreaExtended	欠陥領域補間を拡張するかを設定 デフォルト: False
DefectLevelThreshNeg	直線より下にある欠陥のしきい値を設定 デフォルト: 2
DefectLevelThreshPos	直線より上にある欠陥のしきい値を設定 デフォルト: 2
Direction	エッジ検索の方向を設定 デフォルト: InsideOut
EdgeSort	検出されたエッジ結果のソート方法を設定
EdgeThreshold	この値より低いエッジは無視するという、しきい値を設定 デフォルト: 2
EdgeType	検索するエッジタイプ (シングル, ペア)を設定 デフォルト: 1 – Single
EllipseAngle	ArcInspector の検査ベースラインの楕円弧の角度を指定
EllipseMajorDiam	ArcInspector の検査ベースラインの楕円弧の長径長を指定
EllipseMinorDiam	ArcInspector の検査ベースラインの楕円弧の短径長を指定
Enabled	このオブジェクトを実行するかを指定 デフォルト: True
FailColor	オブジェクトの検出結果が不良時の色を指定 デフォルト: Red
Frame	どの位置に置いたフレームを使用するかを指定 デフォルト: none
FrameResult	Frameの何番目のリザルトを使用するかを指定 デフォルト: 1
Graphics	表示するグラフィックを指定 デフォルト: 1 – All
InspectStartOffset	検査を開始する円弧の始点からオフセットを設定 デフォルト: 5
InspectEndOffset	検査を終了するラインの終点からオフセットを設定 デフォルト: 5
LabelBackColor	オブジェクトラベルの背景の色を設定 デフォルト: Transparent (透明)

プロパティ	説明
MaxArea	オブジェクトの面積上限を設定 検出可能な欠陥はMaxAreaプロパティの設定値より小さく、MinAreaプロパティの設定値より大きいもののみです。 デフォルト: 100,000
MinArea	オブジェクトの面積下限を設定 検出可能な欠陥はMaxAreaプロパティの設定値より小さく、MinAreaプロパティの設定値より大きいもののみです。 デフォルト: 25
MissingEdgeType	未検出のエッジの処理方法を定義 デフォルト: Interpolate
Name	ArcInspectorオブジェクトに独自のオブジェクト名を割りあてる デフォルト: ArcInsp01
NumberOfEdges	検出するエッジ数を設定 デフォルト: 20
NumberToFind	作業中のサーチウィンドウで検出する対象物の数を定義 デフォルト: 1
PassColor	オブジェクトの検出良時の色を選択 デフォルト: LightGreen
PassType	オブジェクト検出時の良否判定の条件を定義 デフォルト: AllNotFound
Polarity	ArcInspectorオブジェクトが明から暗へ (LightToDark)の移行を検索するのか、暗から明へ(DarkToLight)の移行を検索するのかを指定 デフォルト: 1 – LightToDark
Radius	オブジェクトの中心点から、オブジェクトの検索リング外側までの距離を定義
RadiusInner	検出範囲の内径を指定
RadiusOuter	検出範囲の外径を指定
ScoreWeightContrast	コントラストによるスコアの割合を設定 デフォルト: 50
ScoreWeightStrength	エッジの強度によるスコアの割合を設定 デフォルト: 50
SearchWidth	エッジサーチの幅を定義 範囲: 3 ~ 99 デフォルト: 3
SizeToFind	検出する欠陥のサイズを選択 デフォルト: 1 - Largest
StrengthTarget	サーチするエッジの強度を設定 デフォルト: 0
StrengthVariation	StrengthTargetの変化量の合計を設定 デフォルト: 0

## ArcInspector Objectのリザルト

下の一覧は、ArcInspectorオブジェクトリザルトを簡単に説明した概略です。各リザルトの詳細は、Vision Guide 7.0プロパティ&リザルトリファレンスを参照してください。

リザルト	説明
Area	欠陥の面積をピクセルで返す
CameraX	カメラ座標系における欠陥のX座標位置を返す
CameraY	カメラ座標系における欠陥のY座標位置を返す
CameraXYU	実行時のみ カメラ座標系における検出欠陥位置のCameraX座標, CameraY座標, CameraU座標を返す
Contrast	検出エッジの平均コントラストを返す
DefectLevel	欠陥の高さを返す
Length	欠陥の長さをmm単位で返す
PixelLength	欠陥の長さをピクセル単位で返す
NumberFound	検出された欠陥の数を返す (最小0個から、最大はNumberToFindプロパティで設定した検出個数になります。)
Passed	検出結果が良か否かを返す
PixelX	検出円弧エッジの中心X座標をイメージ座標で返す
PixelY	検出円弧エッジの中心Y座標をイメージ座標で返す
PixelXYU	実行時のみ 検出欠陥位置のPixelX座標, PixelY座標, PixelU座標をピクセルで返す
RobotX	ロボット座標系における欠陥のX座標値を返す
RobotY	ロボット座標系における欠陥のY座標値を返す
RobotU	ロボット座標系における欠陥のU座標値を返す
RobotXYU	実行時のみ ロボット座標系における、検出欠陥位置のRobotX座標, RobotY座標, RobotU座標を返す
Strength	検出エッジの強度を返す
Time	オブジェクト処理に要した時間を、ミリ秒単位で返す
TotalArea	検出した欠陥すべての領域の合計をピクセル単位で返す




## ArcInspectorオブジェクトの使用方法

ここからは、ArcInspectorオブジェクトの作成と使用の方法を説明します。

- 新規ArcInspectorオブジェクトの作成方法
- サーチウィンドウの位置とサイズの設定
- ArcInspectorオブジェクトの関連プロパティの設定
- ArcInspectorオブジェクトのテストとリザルトの確認
- プロパティの調整と再テスト


これらのステップを開始する前に、まず新規ビジョンシーケンスを作成するか、使用する

ビジョンシーケンスを選択してください。作業するビジョンシーケンスがない場合は、 <新規シーケンス>ボタンをクリックすると、新規ビジョンシーケンスが作成できます。

すでにビジョンシーケンスを作成済みの場合は、Vision Guideウィンドウのシーケンスツリーをクリックして、作成済みのビジョンシーケンスを選択してください。

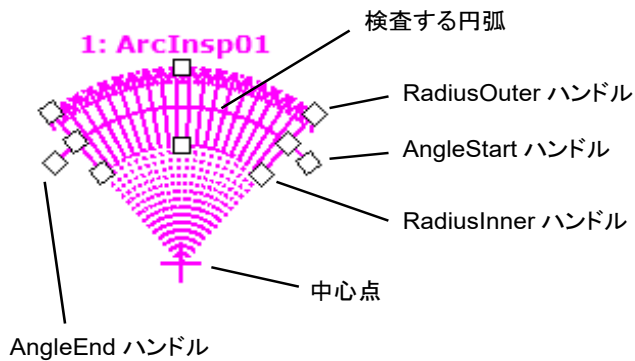
新規ビジョンシーケンスの作成方法、または作成済みビジョンシーケンスの選択方法の詳細は、「5. ビジョンシーケンス」を参照してください。

### Step 1: 新規ArcInspectorオブジェクトの作成

- (1) Vision Guideツールバー - <全ツール> -  <Arc Inspector (円弧検査)>ボタンをクリックしてください。
- (2) 画像イメージ表示部にマウスを動かしてください。マウスポインターがArcInspectorオブジェクトアイコンに変わります。
- (3) アイコンが画像イメージ表示部の目的の位置にくるまで、マウスを動かし続けてください。それから、マウスを左クリックして、オブジェクトを作成してください。
- (4) オブジェクト名が自動的に作成されます。この例では、このシーケンスで最初に作成されたArcInspectorオブジェクトなので、“ArcInsp01”というオブジェクト名になります。(オブジェクト名の変更方法については後述します。)

### Step 2: サーチウィンドウの位置とサイズの設定

画面上には、下図のようなArcInspectorオブジェクトが表示されます。



#### 新規ArcInspector オブジェクトのレイアウト

1. ArcInspectorオブジェクトの名称ラベルをクリックし、マウスを押しながら、中心点が検査する円弧の中心に近くなるようにArcInspectorオブジェクトをドラッグします。
2. RadiusOuter、RadiusInner、AngleStart、AngleEndサイズハンドルをクリックして、マウスをドラッグすると、ArcInspectorオブジェクトサーチウィンドウのサイズが変更できますので、適当な大きさに調整してください。

### Step 3: ArcInspectorオブジェクトプロパティーの設定

ArcInspectorオブジェクトプロパティーの設定を行います。プロパティーを設定するには、関連するプロパティーの設定値フィールドをクリックし、新規の値を入力してください。あるいは、ドロップダウンリストが表示されている場合は、リスト中の該当項目をクリックしてください。

次に、ArcInspectorオブジェクトで一般的に使われるプロパティーのいくつかを示します。その他、さまざまなビジョンオブジェクトの多くで使われる AbortSeqOnFail、Graphics などのプロパティーの詳細は、Vision Guide 7.0 プロパティー&リザルトリファレンスを参照してください。

EdgeType (Single)	<p>検索するエッジの種類を選択</p> <p>エッジペアを設定した場合、それぞれの方向のエッジを検索し、中間点を位置として検出します。</p>
Name property ("ArcInspxx")	<p>新規に作成されたArcInspectorオブジェクトに、デフォルトとして与えられるオブジェクト名は“ArcInspxx”です。xxは、1つのビジョンシーケンス内で複数のArcInspectorオブジェクトを用いるとき、それらのArcInspectorオブジェクトで区別するための番号です。ビジョンシーケンスで1番目に用いられるArcInspectorオブジェクトには、“ArcInsp01”というオブジェクト名がデフォルトで設定されます。オブジェクト名を変更したい場合は、Nameプロパティーの設定値フィールドをクリックして、変更後の名称を入力し、リターンキーを押してください。</p> <p>画面上でArcInspectorオブジェクト名を表示する場所は、すべて変更後の名称に変わります。</p>

NumberOfEdges	欠陥の検出に使用するエッジ検索の数を指定 デフォルトは15、最大で99
Polarity	デフォルトのエッジ検索の極性は「明から暗へ (LightToDark)」 「暗から明へ(DarkToLight)」のエッジを検索する場合は、 Polarityを変更してください。
Direction	エッジ検索の方向にInsideOut (RadiusInnerから RadiusOuter)、またはOutsideIn (RadiusOuterからRadiusInner) を指定
DefectLevelThreshPos	検査する円弧の始点側の、検出したエッジから円弧までの最 短距離を指定
DefectLevelThreshNeg	検査する円弧の終点側の、検出したエッジから円弧までの最 短距離を指定
MinArea	欠陥の最小面積をピクセルで指定
MaxArea	欠陥の最大面積をピクセルで指定

#### Step 4: ArcInspectorオブジェクトの実行とリザルトの確認

ArcInspectorオブジェクトを実行するには、次のようにしてください。

実行パネルのオブジェクトの<Run>ボタンをクリックしてください。

ArcInspectorオブジェクトのリザルトが表示されます。この時点で確認すべき主要なリザルトは次のとおりです。

Areaリザルト	検出された欠陥の面積をピクセル単位で返す
PixelXリザルト	欠陥のXY座標値をイメージ座標で返す
PixelYリザルト	
CameraXリザルト	カメラ座標系における欠陥のXY座標値を返す
CameraYリザルト	これらのリザルトは、カメラキャリブレーションが完了していな い場合は “no cal” を返し、値を返しません。
RobotXリザルト	欠陥のXY座標位置をロボット座標で返す
RobotYリザルト	もしキャリブレーションされていなければ、“no cal” が返され ます。

### 6.2.14 DefectFinder オブジェクト (差分検査)

#### DefectFinderオブジェクトの概要

DefectFinderオブジェクトは、テンプレート画像イメージと入力画像イメージとの差を特定するのに使われます。

欠陥検索では、はじめにサーチエリアとテンプレート画像イメージの絶対差分イメージが計算処理され、差分画像上でブローブ解析を実行し、欠陥を検出します。

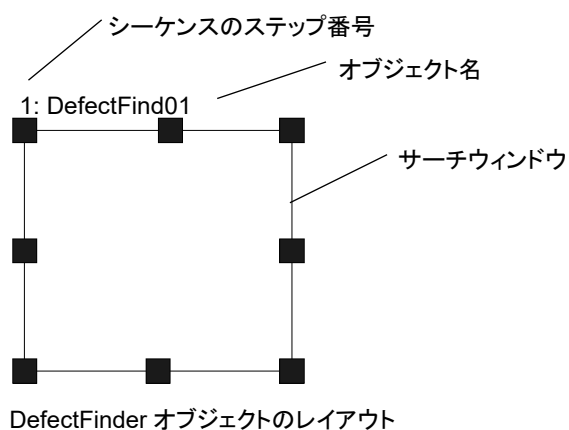
以下の特徴が求められます。

- 領域と周
- 重心
- 主軸と慣性モーメント
- 連結性
- 外接長方形
- ピクセル座標系、カメラ座標系、ロボット座標系での重心の座標位置
- 欠陥(ブローブ)の穴、ざらつき、コンパクトさ

#### DefectFinderオブジェクトのレイアウト

DefectFinderオブジェクトのレイアウトは、Blobオブジェクトのような長方形です。しかし、DefectFinderオブジェクトにはモデル(テンプレート)の「ティーチ」が必要になります。モデルのティーチにはサーチウィンドウで定義された領域全体が使用されます。Correlationオブジェクトのようなモデルウィンドウはありません。

サーチウィンドウは、DefectFinderが欠陥 (イメージ差)を検索する領域を定義すると共に、テンプレート画像の範囲を定義します。DefectFinderオブジェクトの例を下に示します。



## DefectFinderオブジェクトのプロパティー

下の一覧は、DefectFinderオブジェクトプロパティーの概略です。各プロパティーの詳細は、Vision Guide 7.0プロパティー&リザルトリファレンスを参照してください。

プロパティー	説明
AbortSeqOnFail	オブジェクトの実行に失敗した (不良判定になった)とき、直ちにシーケンス全体を中断し、それ以降のオブジェクトを処理しないように指定 デフォルト: False
CalRobotPlacePos	設計時とプログラム実行時に RobotPlacePos をキャリブレーションする
Caption	見出しをDefectFinderオブジェクトに割りあてる デフォルト: 空欄
CenterPointObject	オブジェクトの中心点として用いられる位置を指定 このプロパティーを“Screen”に設定すると画面上の任意の位置にオブジェクトを配置できますが、他のビジョンオブジェクトを指定した場合、そのオブジェクトのPixelX, PixelYリザルトに中心点が設定されます。 デフォルト: Screen
CenterPntObjResult	CenterPointObjectプロパティーから使用するリザルトを指定 Allを指定すると、指定されたビジョンオブジェクトリザルトのすべて (NumberFound)にDefectFinderオブジェクトが適用されます。 デフォルト: 1
CenterPntOffsetX	サーチウィンドウの中心をCenterPointObjectプロパティーで位置決めしたあとに、Xオフセットを設定、または返す デフォルト: 0
CenterPntOffsetY	サーチウィンドウの中心をCenterPointObjectプロパティーで位置決めしたあとに、Yオフセットを設定、または返す デフォルト: 0
CenterPntRotOffset	中心点の XY オフセット値 (CenterPntOffsetX, CenterPntOffsetY)を、CenterPointObject の Angle リザルトにしたがって回転するか否かを指定 SearchWinType が RotatedRectangle に設定されている場合、サーチウィンドウは Angle リザルトにしたがって回転します。 デフォルト: False
CheckClearanceFor	クリアランス (隙間)を確認するオブジェクトを設定
ClearanceCondition	クリアランス (隙間)の判断方法を指定
CoordObject	リザルトをコピーする Coordinates オブジェクトを指定 コピー処理はオブジェクトが実行される時に行われ、Decision の分岐機能などで実行されなかった場合、コピー処理は行われません デフォルト: None

プロパティ	説明
CurrentResult	オブジェクトウィンドウのリザルトリストに表示するリザルト、または、1つのサーチウィンドウ内にある対象物などの複数のリザルトを、システムが検出する設定のときにデータを返すリザルトを定義 デフォルト: 1
Description	ユーザー指定の説明を設定 デフォルト: 空欄
Enabled	このオブジェクトを実行するかを指定 デフォルト: True
FailColor	オブジェクトの検出不良時の色を選択 デフォルト: Red
Frame	どの位置に置いたフレームを使用するかを指定 デフォルト: None
FrameResult	Frameの何番目のリザルトを使用するかを指定 デフォルト: 1
Graphics	表示するグラフィックを指定 デフォルト: 1 - All
KernelHeight	登録画像とのイメージ差を計算するときのピクセルずれの許容量(高さ方向)
KernelWidth	テンプレート画像とのイメージ差を計算するときのピクセルずれの許容量(横方向)
LabelBackColor	オブジェクトラベルの背景の色を設定 デフォルト: Transparent (透明)
LuminanceCorrection	輝度変換の前処理の使用を設定 デフォルト: None
MaxArea	欠陥の上限領域を規定 欠陥が取得するAreaリザルトは、MaxAreaプロパティに設定された値より低くなければなりません。 デフォルト: 100,000
MinArea	欠陥の下限領域を規定 欠陥が取得するAreaリザルトは、MinAreaプロパティに設定された値より高くなければなりません。 デフォルト: 25
MinMaxArea	実行時のみ 1ステートメントでMinAreaとMaxAreaを設定、または返す
Name	DefectFinderオブジェクトに独自のオブジェクト名を割りあてる デフォルト: DefFind01
NumberToFind	サーチウィンドウの中で検出するオブジェクトの数を指定 デフォルト: 1
PassColor	オブジェクトの検出良時の色を選択 デフォルト: LightGreen

プロパティ	説明
PassType	オブジェクト検出時の良否判定の条件を定義 デフォルト: AllNotFound
Polarity	検出する欠陥の極性を指定します デフォルト: Both
RejectOnEdge	Trueに設定すると、サーチウィンドウのエッジにかかって検出された欠陥を無視します。 デフォルト: False
SaveTeachImage	モデルをティーチするときに、画像ファイルに保存するかを設定
SearchWin	実行時のみ サーチウィンドウの左端、上端、高さ、幅のパラメーターを設定、または返す
SearchWinAngle	被検索領域の角度を定義
SearchWinCenterX	被検索領域の中心のX座標値を定義
SearchWinCenterY	被検索領域の中心のY座標値を定義
SearchWinHeight	被検索領域の高さを定義 (単位: ピクセル) デフォルト: 100
SearchWinLeft	被検索領域の最左端を定義 (単位: ピクセル)
SearchWinTop	被検索領域の最上端を定義 (単位: ピクセル)
SearchWinType	被検査領域のタイプ (Rectangle, RotatedRectangle, Circle)を定義
SearchWinWidth	被検索領域の幅を定義 (単位: ピクセル) デフォルト: 100
ShowModel	登録済みイメージ画像を表示します。 検出マスクを設定
SizeToFind	検出する欠陥のサイズを選択します。 デフォルト: 1 - Largest
Sort	オブジェクトのリザルトを並び替える順番を選択 デフォルト: 0 - None
ThresholdHigh	ThresholdLowプロパティとともに、対象物 (欠陥)、背景、画像イメージのエッジを表すグレイレベル範囲を定義 ThresholdHighプロパティは、画像イメージの対象物領域のグレイレベル範囲の上限を定義します。ThresholdLowとThresholdHighで定義されたグレイレベル範囲にあてはまる画像イメージがThresholdColorに設定されます。 デフォルト: 128
ThresholdLow	ThresholdHighプロパティとともに、対象物 (欠陥)、背景、画像イメージのエッジを表すグレイレベル範囲を定義 ThresholdLowプロパティは、画像イメージの対象物領域のグレイレベル範囲の下限を定義します。ThresholdLowとThresholdHighで定義されたグレイレベル範囲にあてはまる画像イメージがThresholdColorに設定されます。 デフォルト: 0

## DefectFinderオブジェクトのリザルト

下の一覧は、DefectFinderオブジェクトリザルトを簡単に説明した概略です。各リザルトの詳細は、Vision Guide 7.0プロパティ&リザルトリファレンスを参照してください。

リザルト	説明
Angle	検出された欠陥の回転量を角度で返す
Area	欠陥の面積をピクセルで返す
CameraX	カメラ座標系における検出欠陥のX座標位置を返す
CameraY	カメラ座標系における検出欠陥のY座標位置を返す
CameraXYU	実行時のみカメラ座標における検出欠陥位置のCameraX座標, CameraY座標, CameraU座標を返す
ClearanceOK	クリアランス (隙間)の判断結果を返す
Compactness	欠陥の緊密さを返す
Extrema	実行時のみ欠陥極値のMinXピクセル座標, MaxXピクセル座標, MinYピクセル座標, MaxYピクセル座標を返す
Found	欠陥が検出されたかどうかを返す
FoundOnEdge	検出された欠陥がサーチウィンドウのエッジに接している状態で検出されるとき、Trueを返す
Holes	欠陥内で検出される穴の数を返す
MajorDiameter	検出された欠陥を楕円形に近似した場合の長径を返す
MaxFeretDiameter	検出された欠陥の最大フェレ径を返す
MaxX	欠陥の外接長方形の最大Xピクセル座標を返す
MaxY	欠陥の外接長方形の最大Yピクセル座標を返す
MinorDiameter	検出された欠陥を楕円形に近似した場合の短径を返す
MinX	欠陥の外接長方形の最小Xピクセル座標を返す
MinY	欠陥の外接長方形の最小Yピクセル座標を返す
NumberFound	検出された欠陥の数を返す (最低0個から、最高はNumberToFindプロパティで設定したオブジェクト検出個数になります。)
Passed	検出結果が良か否かを返す
Perimeter	検出された欠陥の、外側エッジの端から端までのピクセルの数
PixelX	検出パーツ位置のX座標をピクセルで返す
PixelY	検出パーツ位置のY座標をピクセルで返す
PixelXYU	実行時のみ 検出パーツ位置のPixelX座標, PixelY座標, PixelU座標をイメージ座標で返す
RobotX	ロボット座標系における検出パーツのX座標位置を返す
RobotY	ロボット座標系における検出パーツのY座標位置を返す
RobotU	ロボット座標系における検出パーツのU座標位置を返す



リザルト	説明
RobotXYU	実行時のみ ロボット座標系における、検出パーツ位置のRobotX座標, RobotY座標, RobotU座標を返す
Roughness	欠陥の粗さを返す
ShowAllResults	特定のビジョンオブジェクトのリザルトすべてを、表形式でダイアログに表示 リザルトの比較が簡単にできます。
Time	オブジェクト処理に要した時間を返す (単位: ミリ秒)
TotalArea	検出したリザルトすべての欠陥領域の合計を返す

DefectFinderオブジェクトはテンプレート画像とのイメージ差をBlob処理してリザルトを返します。リザルトの詳細はBlobオブジェクトの章を参照してください。


### DefectFinderオブジェクトの使用方法

以上、Vision GuideのDefectFinderオブジェクトの使用方法を理解するための基本的知識として、2値検査がどのように動作するのか確認しました。ここでは、DefectFinderオブジェクトの使用方法を説明します。

DefectFinderオブジェクトを使うのに必要なステップは次のとおりです。

- 新規DefectFinderオブジェクトの作成
- サーチウィンドウの位置とサイズの設定
- DefectFinderオブジェクトの関連プロパティの設定
- テンプレート画像の登録
- DefectFinderオブジェクトのテストとリザルトの確認
- プロパティの調整と再テスト


これらのステップを開始する前に、まず新規ビジョンシーケンスを作成するか、使用する

ビジョンシーケンスを選択してください。作業するビジョンシーケンスがない場合は、 <新規シーケンス>ボタンをクリックすると新規ビジョンシーケンスが作成できます。

すでにビジョンシーケンスを作成済みの場合は、Vision Guideウィンドウのシーケンスツリーをクリックして、作成済みのビジョンシーケンスを選択してください。

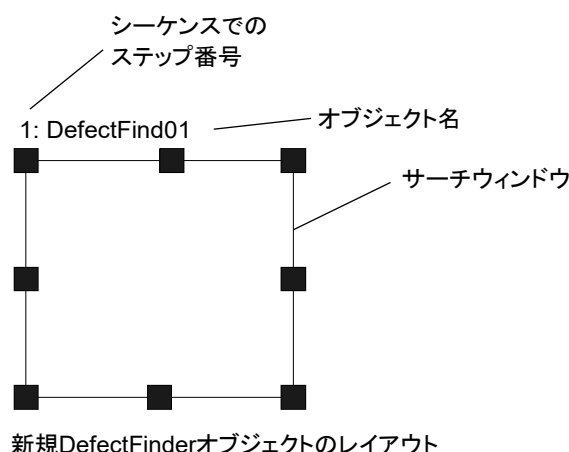
新規ビジョンシーケンスの作成方法、または作成済みビジョンシーケンスの選択方法の詳細は、「ビジョンシーケンス」を参照してください。

### Step 1: 新規DefectFinderオブジェクトの作成

- (1) Vision Guideツールバー - <全ツール> -  <Defect Finder (欠陥検査)>ボタンをクリックしてください。
- (2) 画像イメージ表示部にマウスを動かしてください。マウスポインターがDefectFinderアイコンに変わります。
- (3) アイコンが画像イメージ表示部の目的の位置に来るまで、マウスを動かし続けてください。それから、マウスを左クリックして、オブジェクトを作成してください。
- (4) オブジェクト名は自動的に作成されますので注意してください。この例では、このシーケンスで最初に作成されたDefectFinderオブジェクトなので、“DefFind01”という名称になります。(名称の変更方法については後述します。)

## Step 2: サーチウィンドウの位置とサイズの設定

画面上には、下図のようなDefectFinderオブジェクトが表示されます。



- (1) DefectFinderオブジェクトの名称ラベルをクリックし、マウスを押し続けながら、DefectFinderオブジェクトをドラッグして、サーチウィンドウを好きな位置に合わせてください。
- (2) サイズハンドルをクリックして、マウスをドラッグすると、DefectFinderオブジェクトサーチウィンドウのサイズが変更できますので、適当な大きさに調整してください。(サーチウィンドウは、ブローブの被検索領域になります。)



注意

- ビジョンシーケンスの実行結果は、外らん光や外部機器ノイズにより影響を受けることがあります。  
外らん光や外部機器ノイズにより、影響を受けると、取り込み画像が想定外の結果となり、検出位置結果がサーチエリア (検出エリア)内の不特定位置になる可能性があります。  
サーチエリアは、極力小さく、また誤検出に配慮した画像処理シーケンスの作成を行ってください。

## Step 3: DefectFinderオブジェクトプロパティーの設定

このステップでは、DefectFinderオブジェクトのプロパティーを設定することができます。次に、DefectFinderオブジェクトに特有で、よく使用されるプロパティーを示します。

その他、さまざまなビジョンオブジェクトの多くで使われるAbortSeqOnFail, Graphicsなどのプロパティーの詳細は、Vision Guide 7.0プロパティー&リザルトリファレンス、または、本章で説明した「DefectFinderオブジェクトのプロパティー」リストを参照してください。



注意

- ビジョンシーケンスの実行結果は、外らん光や外部機器ノイズにより影響を受けることがあります。  
外らん光や外部機器ノイズにより、影響を受けると、取り込み画像が想定外の結果となり、検出位置結果がサーチエリア (検出エリア)内の不特定位置になる可能性があります。  
MaxArea, MinArea, RejectOnEdgeなどのプロパティー値を適切に設定し、誤検出リスクを軽減してください。

Nameプロパティ	新規に作成されたDefectFinderオブジェクトにデフォルトとして与えられる名称は、“DefFindxx”です。xxは、1つのビジョンシーケンス内で複数のDefectFinderオブジェクトを用いるとき、それらのDefectFinderオブジェクトを区別するための番号です。 ビジョンシーケンスで1番目に用いられるDefectFinderオブジェクトには、“DefFind01”という名称がデフォルトで設定されます。 名称を変更したい場合は、Nameプロパティの設定値フィールドをクリックして、変更後の名称を入力し、リターンキーを押してください。画面上でそのDefectFinderオブジェクト名を表示する場所の全部が、変更後の名称を表示するようになります。
KernelWidth, KernelHeight プロパティ	登録済みテンプレート画像とのイメージ差を計算するとき、どの程度のピクセルずれを許容量するかを設定 値を大きく設定すると外らんやテンプレート画像とのずれに強くなりますが、小さな欠陥を検出できなくなります。検出する欠陥サイズ、および入力画像のずれ量に合わせて設定してください。
MinArea, MaxArea プロパティ	“Found”とみなされる (Foundリザルトの値が “True”だった) DefectFinderオブジェクトの面積を定義 デフォルトの範囲は、25~100,000 (MinArea ~ MaxArea)と非常に広く設定されています。つまり、MinAreaプロパティとMaxAreaプロパティを調整しないまま、新規作成のDefectFinderオブジェクトを実行すると、細かい欠陥まで検出されたとして報告されます。通常は、検出対象の欠陥サイズに合わせて、これらのプロパティを調整します。
RejectOnEdge プロパティ	サーチウィンドウのエッジにかかって検出された欠陥を除外
PassType プロパティ	DefectFinderオブジェクトの検出結果の良否判定方法を設定 通常は欠陥が検出されない状態が良判定です。“AllNotFound”に設定します。

以上で、DefectFinderオブジェクトのテストができるようになりました。必要となるその他のプロパティは、テスト後に再び設定します。

### Step 4: テンプレート画像の登録と確認

DefectFinderオブジェクトのテンプレート画像を登録するには、実行パネルにある、<ティーチ>ボタンをクリックしてください。登録されたテンプレート画像を確認するには、プロパティリストのShowModelプロパティをクリックします。

## Step 5: DefectFinderオブジェクトのテストとリザルトの確認

DefectFinderオブジェクトのテストを実行するには、実行パネルのオブジェクトの<Run>ボタンをクリックしてください。DefectFinderオブジェクトのリザルトが表示されます。ここで確認する主要なリザルトは次に示すとおりです。けれども、役に立つリザルトがこれから他にも出てきます。

Foundリザルト	欠陥が検出されたかどうかの値を返す  検出された欠陥が、MinAreaプロパティとMaxAreaプロパティで定義された面積の範囲内にはないときは、Foundリザルトは“False”を返します。
Passedリザルト	DefectFinderオブジェクトの検出結果の良否を返す
Areaリザルト	検出された欠陥の面積 (単位: ピクセル)
Angleリザルト	欠陥の方向角度  副軸の角度から計算し、+90° ~ -90°までの範囲の値になります。
Timeリザルト	DefectFinderオブジェクトの実行に要した時間
PixelX, PixelY	検出欠陥の重心のXY座標位置 (単位: ピクセル)
MinX, MinY, MaxX, MaxY	これらの4つの値で、欠陥の外接長方形を返します。

## NOTE



RobotXYU, RobotX, RobotY, RobotUのリザルトとCameraX, CameraY, CameraXYUのリザルトは、この時点で“no cal”を返します。これは、キャリブレーションが実行されていないので、ロボット座標系あるいはカメラ座標系に関して座標リザルトを計算することができなかった、ということを意味しています。詳細は、「7. ビジョンキャリブレーション」を参照してください。

## Step 6: プロパティの調整と再テスト

DefectFinderオブジェクトを数回実行することにより、欠陥検出上の問題が見つかったり、プロパティ設定の微調整をする必要が出てくるかもしれません。一般的に生じる問題や、微調整の方法について次に説明します。

## 問題：

DefectFinderオブジェクトのFoundリザルトが“True”を返すときは、次の事項を確認し、調整してください。

- Polarityプロパティで定義された値を見てください。暗い背景の明るいオブジェクトか、明るい背景の暗いオブジェクトのいずれかです。Polarityオブジェクトはサーチウィンドウに表示されている対象物と一致します。
- Areaリザルトを確認し、この値とMinAreaプロパティとMaxAreaプロパティで定義された値とを比較してください。  
Areaリザルトが、MinAreaプロパティとMaxAreaプロパティで定義された範囲に入っていない場合は、MinAreaプロパティとMaxAreaプロパティを調整して、Blobオブジェクトを再実行してください。
- KernelWidth, KernelHeightプロパティを調整してください。  
設定値を大きくすることで、小さな欠陥の誤検出を回避できます。

- ヒストグラムを使用して、画像イメージのグレイスケール値を調べます。ヒストグラムツールはThresholdHighとThresholdLowプロパティを設定するのに適しています。ヒストグラムについての詳細は、「8. ヒストグラムツール」に記載しています。

### 微調整：

アプリケーションによっては、DefectFinderオブジェクトの微調整が必要となる場合があります。DefectFinderオブジェクトの微調整に関する主要なプロパティについて、次に説明します。

- MinArea, MaxArea – DefectFinderオブジェクトを数回実行してみると、Areaリザルトに返される概略値がわかってきます。MinAreaプロパティとMaxAreaプロパティの値を変更するときは、これらの概略値を使ってください。一般的に、MinAreaプロパティとMaxAreaプロパティの値は、Foundリザルトを制限するように設定し、検出対象のブローブだけについて“True”のFoundリザルトが返るようにすると好都合です。(このように設定することによって、検出したい欠陥と面積が異なっているブローブを排除することができます。)

- ThresholdHigh, ThresholdLow – これらのプロパティは、欠陥を識別するために、グレイレベルのしきい値を設定するパラメーターを調整します。

「Vision Guide 7.0プロパティ&リザルトリファレンス」の「ThresholdHighプロパティ」と「ThresholdLowプロパティ」を参照してください。

プロパティの調整が完了し、DefectFinderオブジェクトのテストを行って満足な欠陥検出結果を得ることができたら、ビジョンオブジェクトの作成ステップは完了です。

他のビジョンオブジェクトを作成したり、ビジョンシーケンス全体の設定やテストを行う次のステップに進むことができます。

### 6.2.15 Frame オブジェクト (フレーム)

#### Frameオブジェクトの概要

Frameオブジェクトは、ビジョンオブジェクトの一種のダイナミックな基準となります。

新規のFrameオブジェクトを定義すると、その他のビジョンオブジェクトはそのFrameオブジェクトを基準にして位置決めすることができます。そのため、Frameオブジェクトはさまざまな状況で役に立ちます。

また、Frameオブジェクトを利用した場合、大まかな位置が検出され、フレームが定義されると、Frameオブジェクトに基づくその他のビジョンオブジェクトでは、サーチウィンドウを大きくする必要がないので、ビジョンの処理時間を短縮することができます。

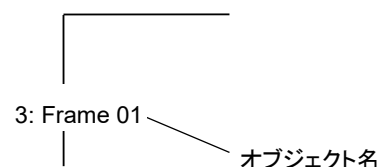
Frameオブジェクトは、パーツ上に何らかの共通参照パターン (例えば、プリント回路基板上の基準点など)があり、そのパターンが、その他のビジョンオブジェクトサーチウィンドウによって基準位置として用いられる場合に、最大の利用価値があります。

### Frameオブジェクトのレイアウト

Frameオブジェクトは、交差する2本のLineオブジェクトのような外観をしています。

Frameオブジェクトの位置を変更したいときは、Frameオブジェクト名をクリックし、オブジェクトを変更したい位置にドラッグしてください。

Frameオブジェクトの位置と方向は、ほとんどの場合が他のオブジェクトの位置に基づいて設定されます。



Frameオブジェクトのレイアウト

### Frameオブジェクトのプロパティー

下の一覧は、Frameオブジェクトプロパティーの概略です。各プロパティーの詳細は、「Vision Guide 7.0プロパティー&リザルトリファレンス」を参照してください。

プロパティー	説明
AbortSeqOnFail	オブジェクトの実行に失敗した (不良判定になった)とき、直ちにシーケンス全体を中断し、それ以降のオブジェクトを処理しないように指定 デフォルト: False
Caption	見出しをFrameオブジェクトに割りあてる デフォルト: 空欄
CurrentResult	オブジェクトウィンドウのリザルトリストに表示するリザルト、または、1つのサーチウィンドウ内にある対象物などの複数のリザルトを、システムが検出する設定のときにデータを返すリザルトを定義 デフォルト: 1
Description	ユーザー指定の説明を設定 デフォルト: 空欄
Enabled	このオブジェクトを実行するかを指定 デフォルト: True
FailColor	オブジェクトの検出不良時の色を選択 デフォルト: Red
Graphics	表示するグラフィックを指定 デフォルト: 1 – All
LabelBackColor	オブジェクトラベルの背景の色を設定 デフォルト: Transparent (透明)
Name	Frameオブジェクトに独自のオブジェクト名を割りあてる デフォルト: Frame01

プロパティ	説明
OriginAngleEnabled	ツーポイントFrameの場合のようにOriginPointプロパティとYaxisPointプロパティ間のベクトルの回転に基づくのではなく、フレームに原点オブジェクトの角度を回転させるシングルポイントフレーム使用を可能にします。 デフォルト: False
OriginPntObjResult	OriginPointプロパティで指定されたビジョンオブジェクトから使用するリザルトを指定  “All”を指定すると、指定されたビジョンオブジェクトのすべてにFrameオブジェクトが適用されます。 デフォルト: 1 (最初のリザルトを使います)
OriginPoint	Frameオブジェクトの原点として用いるビジョンオブジェクトを定義 デフォルト: Screen
PassColor	オブジェクトの検出良時の色を選択 デフォルト: LightGreen
PassType	オブジェクト検出時の良否判定の条件を定義 デフォルト: SomeFound
ShowExtensions	フレームの延長線を表示するかを設定 デフォルト: False
YAxisPoint	FrameオブジェクトのY軸上のポイントとして用いるビジョンオブジェクトを定義 (Frameオブジェクトの方向を定義します。) デフォルト: Screen
YAxisPntObjResult	YaxisPointプロパティで指定されたビジョンオブジェクトから使用するリザルトを指定 デフォルト: 1 (最初のリザルトを使います。)

### Frameオブジェクトのリザルト

Frameオブジェクトリザルトの一覧を次に示し、簡単に説明します。

Frameオブジェクトで使われる各リザルトの詳細は、「Vision Guide 7.0プロパティ&リザルトリファレンス」を参照してください。

リザルト	説明
Angle	検出されたパーツの回転量を角度で返す
Found	オブジェクトが検出されたかどうかを返す

### ツーポイントフレーム

ツーポイントのFrameオブジェクトは、原点 (OriginPointプロパティで定義される)とY軸方向 (YAxisPointプロパティで定義される)を必要とします。原点とY軸方向との組み合わせによって、その他のビジョンオブジェクトが基準とする一種のローカル座標系が定義されます。



Frameオブジェクトの強力な利点は、Frameオブジェクトが移動すると、そのフレーム内で定義されるビジョンオブジェクトのすべてが、Frameオブジェクトと一緒に移動することです。(つまり、OriginPointと定義されたビジョンオブジェクトのXY変化とYaxisPointプロパティの移動で作成された回転に基づいて、サーチウィンドウは調整されます。)したがって、サーチウィンドウを小さくしたままで作業ができるため、ビジョンの処理時間を短縮することができます。

### Frame オブジェクトの定義

新規のFrameオブジェクトが作成されると、そのFrameオブジェクトには、フレームの原点とY軸上のポイントの基準点として、2つのビジョンオブジェクトが必要になります。

原点とY軸方向は、OriginPointプロパティとYAxisPointプロパティによって定義されます。XY位置リザルトを持つビジョンオブジェクトであれば、フレームの原点とY軸上のポイントを定義するのに用いることができます。

つまり、Blob, Correlation, Edge, Polar, Pointのオブジェクトのすべては、FrameオブジェクトのOriginPointプロパティとYAxisPointプロパティを定義するのに用いることができます。

### シングルポイントフレーム

シングルポイントフレームは、Frameオブジェクトのオプション使用方法です。この使用で、OriginPointプロパティは、ビジョンオブジェクトをXY位置原点参照として使えるように定義します。


OriginAngleEnabledプロパティをFalseに設定したとき、原点ポイントとして使われるビジョンオブジェクトのXY位置変更に基づいて、フレームは位置を調節します。回転を考慮する必要はありません。これは、1つのオブジェクト(blobやcorrelationなど)がパーツのXY位置を検出して、フレーム中の残りのオブジェクトはそれに従ってXとYを合わせる場合、簡単にXY移動ができ便利です。

いくつかの場合、フレームの中で回転を説明する必要があるかもしれません。Blobオブジェクトは、パーツのX, Y, U位置 (XY座標+回転)を検出するのに使われると考えてください。そして、他の種類のビジョンオブジェクトは、パーツ上の対象物を検出すると考えてください。Blobオブジェクトはフレームの回転を含むシングルポイントフレームを定義するのに使えます。そして、フレーム中の他のオブジェクトは、X, Yに移動して、Blobオブジェクトが返した回転に基づいて回転します。したがって、1つのビジョンオブジェクトだけでXY移動とパーツの回転の両方を定義しました。これがシングルポイントフレームにYaxisPointプロパティを必要としない理由です。

### Frameオブジェクトの使用方法


ここからは、Frameオブジェクトの作成方法と使用方法を説明します。

- 新規Frameオブジェクトの作成方法
- サーチウィンドウの位置とサイズの設定
- Frameオブジェクトの関連プロパティの設定
- Frameオブジェクトのテストとリザルトの確認
- プロパティの調整と再テスト

これらのステップを開始する前に、まず新規ビジョンシーケンスを作成するか、使用するビジョンシーケンスを選択してください。作業するビジョンシーケンスがない場合は、 <新規シーケンス>ボタンをクリックすると、新規ビジョンシーケンスが作成できます。すでにビジョンシーケンスを作成済みの場合は、Vision Guideウィンドウのシーケンスツリーをクリックして、作成済みのビジョンシーケンスを選択してください。

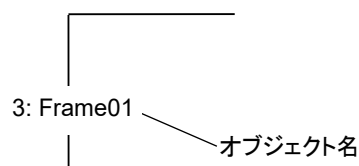
新規ビジョンシーケンスの作成方法、または作成済みビジョンシーケンスの選択方法の詳細は、「ビジョンシーケンス」を参照してください。

### Step 1: 新規Frameオブジェクトの作成

- (1) Vision Guideツールバー - <全ツール> -  <Frame (フレーム)>ボタンをクリックしてください。
- (2) 画像イメージ表示部にマウスを動かしてください。マウスポインターがFrameオブジェクトアイコンに変わります。
- (3) アイコンが画像イメージ表示部の目的の位置に来るまで、マウスを動かし続けてください。それから、マウスを左クリックして、オブジェクトを作成してください。
- (4) オブジェクト名が自動的に作成されるのに注目してください。この例では、このシーケンスで最初に作成されたFrameオブジェクトなので、“Frame01”というオブジェクト名になります。(オブジェクト名の変更方法については後述します。)

### Step 2: Frameオブジェクトの位置決め

画面上には、下図のようなFrameオブジェクトが表示されます。



#### 新規Frameオブジェクト

Frameオブジェクトには大きなウィンドウはありません。Frameオブジェクトの名称ラベルや軸の周囲をクリックして、マウスを押し続けながら、Frameオブジェクト全体を画面上の新しい位置にドラッグしてください。位置が決まったら、マウスを離してください。Frameオブジェクトが画面上の新しい位置にきます。

### Step 3: Frameオブジェクトプロパティの設定

Frameオブジェクトプロパティの設定を行います。プロパティを設定するには、関連するプロパティの設定値フィールドをクリックし、新規の値を入力してください。あるいは、ドロップダウンリストが表示されている場合は、リスト中の該当項目をクリックしてください。

次に、Frameオブジェクトで一般的に使われるプロパティのいくつかを示します。その他、さまざまなビジョンオブジェクトの多くで使われるAbortSeqOnFail, Graphicsなどのプロパティの詳細は、「Vision Guide 7.0プロパティ&リザルトリファレンス」、および本章の「Frameオブジェクトのプロパティ」を参照してください。

<b>Nameプロパティ</b>	新規に作成されたFrameオブジェクトに、デフォルトとして与えられるオブジェクト名は”Framexx”です。xxは、1つのビジョンシーケンス内で複数のFrameオブジェクトを用いるとき、それらのFrameオブジェクトで区別するための番号です。ビジョンシーケンスで1番目に用いられるFrameオブジェクトには、“Frame01”というオブジェクト名がデフォルトで設定されます。オブジェクト名を変更したい場合は、Nameプロパティの設定値フィールドをクリックして、変更後の名称を入力し、リターンキーを押してください。画面上でFrameオブジェクト名を表示する場所は、すべて変更後の名称に変わります。
<b>OriginPoint プロパティ</b>	通常、このプロパティをシーケンスの前の方で行われるオブジェクトの1つに設定  実行時のFrameの原点を決定します。
<b>YAxisPoint プロパティ</b>	通常、このプロパティをシーケンスの前の方で行われるオブジェクトの1つに設定  実行時のFrameのY軸方向を決定します。

### Step 4: Frameオブジェクトの実行とリザルトの確認

Frameオブジェクトを実行するには、次のようにしてください。

実行パネルのオブジェクトの<Run>ボタンをクリックしてください。OriginPointプロパティまたはYAxisPointプロパティのどちらかがScreenでない場合、それぞれのオブジェクトが最初に実行されます。例えば、OriginPointプロパティがBlobオブジェクトなら、フレームの原点位置を決定するためにブローブが最初に実行されます。

Frameオブジェクトのリザルトが表示されます。この時点で確認すべき主要なリザルトは次のとおりです。

<b>Angle</b>	フレームの角度
--------------	---------

### 6.2.16 Line オブジェクト (ライン)

#### Lineオブジェクトの概要

Lineオブジェクトは、2点間の直線を定義するのに使われます。

2つの点は、画面上の位置あるいは他のビジョンオブジェクトの位置を基準とすることができます。下に示すのは、Lineオブジェクトが作成されるいくつかの状況です。

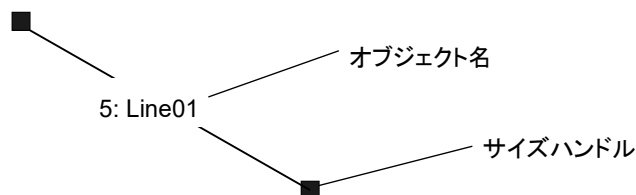
- 2つのBlobオブジェクト間
- 2つのCorrelationオブジェクト間
- 2つのPointオブジェクト間
- BlobオブジェクトとCorrelationオブジェクトの間
- 1つのBlobオブジェクトの2つのリザルト間
- Blobオブジェクトのリザルト1とCorrelationオブジェクトのリザルト3の間
- PointオブジェクトとCorrelationオブジェクトの間
- XY位置リザルトを持つあらゆる組み合わせのビジョンオブジェクト間

Lineオブジェクトは次のような状況で使用する就非常に便利です。

- ビジョンオブジェクト間の距離 (または、複数のリザルトが使われるときのビジョンオブジェクトのリザルト)を測定したいとき  
距離は、アプリケーションが必要とする最短距離から最長距離までになるように確認することもできます。
- 2つのビジョンオブジェクト間の回転量を計算したいとき  
(RobotURIザルトというロボット座標で返された直線の角度を使います。)
- 直線の中心点、または2直線の交点を計算するブロックを作成したいとき

#### Lineオブジェクトのレイアウト

Lineオブジェクトのレイアウトは、文字どおりラインのレイアウトになっています。つまり、始点、終点、オブジェクト名からなる1本のラインです。Lineオブジェクトの位置を変更したいときは、Lineオブジェクト名(またはライン上)をクリックし、ラインを変更したい位置にドラッグしてください。Lineオブジェクトのサイズを調整するには、ラインの始点あるいは終点(サイズハンドルで示しています)をクリックし、望みのサイズになる位置にドラッグしてください。



Lineオブジェクトのレイアウト

### Lineオブジェクトのプロパティー

Lineオブジェクトプロパティーのリストを次に示し、簡単に説明します。

Lineオブジェクトで使われる各プロパティーの詳細は、「Vision Guide 7.0プロパティー&リザルトリファレンス」を参照してください。

プロパティー	説明
AbortSeqOnFail	オブジェクトの実行に失敗した (不良判定になった)とき、直ちにシーケンス全体を中断し、それ以降のオブジェクトを処理しないように指定 デフォルト: False
AngleBase	基準角度を設定 デフォルト: 0
AngleMode	角度の出力形式を指定 デフォルト: 1-Default
AngleStart	角度サーチの中心を指定
LabelBackColor	オブジェクトラベルの背景の色を設定 デフォルト: Transparent (透明)
Caption	見出しをLineオブジェクトに割りあてる デフォルト: 空欄
CurrentResult	オブジェクトウィンドウのリザルトリストに表示するリザルト、または、1つのサーチウィンドウ内にある対象物などの複数のリザルトを、システムが検出する設定のときにデータを返すリザルトを定義
Description	ユーザー指定の説明を設定 デフォルト: 空欄
Directed	線の方向を使って角度を設定するかを指定 デフォルト: True
Enabled	このオブジェクトを実行するかを指定 デフォルト: True
EndPointObjResult	どのリザルトをEndPointObjectから用いるのかを特定 デフォルト: 1
EndPointObject	どのビジョンオブジェクトを使って、直線の終点を定義するかを特定 デフォルト: Screen
EndPointType	どのタイプの終点を使って直線の終点を定義するかを特定 デフォルト: 0 - Point
FailColor	オブジェクトの検出不良時の色を選択 デフォルト: Red
Frame	どの位置に置いたフレームを使用するかを指定 デフォルト: none
Graphics	表示するグラフィックを指定 デフォルト: 1 - All

プロパティ	説明
MaxLength	Lineオブジェクトの長さの上限を定義 Lineが検出されるには、MaxLengthプロパティに設定されている値より下のLengthリザルトを持たなければなりません。 デフォルト: 9999
MaxPixelLength	Lineオブジェクトのピクセル単位長さの上限を定義 Lineが検出されるには、MaxPixelLengthプロパティに設定されている値より下のPixelLengthリザルトを持たなければなりません。 デフォルト: 9999
MinLength	Lineオブジェクトの長さの下限を定義 Lineが検出されるには、MinLengthプロパティに設定されている値より上のLengthリザルトを持たなければなりません。 デフォルト: 0
MinPixelLength	Lineオブジェクトのピクセル単位長さの下限を定義 Lineが検出されるには、MinPixelLengthプロパティに設定されている値より上のPixelLengthリザルトを持たなければなりません。 デフォルト: 0
Name	Lineオブジェクトに独自のオブジェクト名を割りあてる デフォルト: Line01
PassColor	オブジェクトの検出良時の色を選択 デフォルト: LightGreen
PassType	オブジェクト検出時の良否判定の条件を定義 デフォルト: SomeFound
ShowExtensions	“True”に設定すると、画像イメージ表示部の端までの延長線をグラフィック表示 デフォルト: False
StartPntObjResult	どのリザルトをStartPointObjectから用いるのかを指定 デフォルト: 1
StartPointObject	どのビジョンオブジェクトを使って、直線の始点を定義するかを指定 デフォルト: Screen
StartPointType	どのタイプの始点を使って直線の始点を定義するかを指定 デフォルト: “0 – Point”
X1	直線の始点のX座標位置
X2	直線の終点のX座標位置
Y1	直線の始点のY座標位置
Y2	直線の終点のY座標位置

## Lineオブジェクトのリザルト

下の一覧は、Lineオブジェクトリザルトを簡単に説明した概略です。各リザルトの詳細は、「Vision Guide 7.0プロパティ&リザルトリファレンス」を参照してください。


リザルト	説明
Angle	時計の3時方向を0°とする位置を基準に、始点と終点で形成される直線の角度を返す
CameraX1	カメラ座標系における直線の始点のX座標位置を返す
CameraX2	カメラ座標系における直線の終点のX座標位置を返す
CameraY1	カメラ座標系における直線の始点のY座標位置を返す
CameraY2	カメラ座標系における直線の終点のY座標位置を返す
Found	オブジェクトが検出されたかどうかの値を返す  Lineオブジェクトを定義するビジョンオブジェクトが検出されなかった場合は、Lineオブジェクトは検出されません。
Length	直線の長さをミリメートルの単位で返す  あらかじめ、カメラのキャリブレーションをしておいてください。キャリブレーションが完了していない場合は、長さのリザルトとして”no cal”の値が返されます。  LineオブジェクトがMaxLengthとMinLengthの制約で失敗したとき、Lengthリザルトはリザルトリストに赤で表示されます。
NumberFound	検出された直線の数返す
PixelLength	直線の長さをピクセル単位で返す  LineオブジェクトがMaxPixelLengthとMinPixelLengthの制約で失敗したとき、PixelLengthリザルトはリザルトリストに赤で表示されます。
PixelLine	実行時のみ  X1, Y1, X2, Y2の4つの直線座標をピクセル単位で返す
PixelX1	直線の始点のX座標位置をピクセル単位で返す
PixelX2	直線の終点のX座標位置をピクセル単位で返す
PixelY1	直線の始点のY座標位置をピクセル単位で返す
PixelY2	直線の終点のY座標位置をピクセル単位で返す
RobotX1	検出エッジラインの始点X座標をロボット座標で返す
RobotX2	検出エッジラインの終点X座標をロボット座標で返す
RobotY1	検出エッジラインの始点Y座標をロボット座標で返す
RobotY2	検出エッジラインの終点Y座標をロボット座標で返す
RobotU	ロボット座標系において、始点と終点で形成される直線の角度を返す

### Lineオブジェクトの使用方法

ここからは、Lineオブジェクトの作成方法と使用方法を説明します。


- 新規Lineオブジェクトの作成
- Lineオブジェクトの位置決め
- Lineオブジェクトのプロパティの設定
- Lineオブジェクトの実行とリザルトの確認

次に説明するステップを開始する前に、新規ビジョンシーケンスを作成するか、使用する

ビジョンシーケンスを選択してください。作業するビジョンシーケンスがない場合は、 <新規シーケンス>ボタンをクリックすると、新規ビジョンシーケンスが作成できます。すでにビジョンシーケンスを作成済みの場合は、Vision Guideウィンドウのシーケンスツリーをクリックして、作成済みのビジョンシーケンスを選択してください。

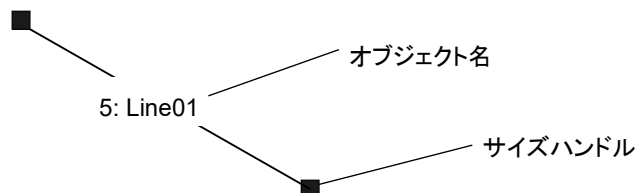
新規ビジョンシーケンスの作成方法、または作成済みビジョンシーケンスの選択方法の詳細は、「5. ビジョンシーケンス」を参照してください。

### Step 1: 新規Lineオブジェクトの作成

1. Vision Guideツールバー - <全ツール> -  <Line (ライン)>ボタンをクリックしてください。
2. 画像イメージ表示部にマウスを動かしてください。マウスポインターがLineオブジェクトアイコンに変わります。
3. アイコンが画像イメージ表示部の目的の位置に来るまで、マウスを動かし続けてください。それから、マウスを左クリックして、オブジェクトを作成してください。
4. オブジェクト名が自動的に作成されるのに注目してください。この例では、このシーケンスで最初に作成されたLineオブジェクトなので、“Line01”というオブジェクト名になります。(オブジェクト名の変更方法については後述します。)

### Step 2: Lineオブジェクトの位置決め

画面上には、下図のようなLineオブジェクトが表示されています。



#### 新規 Line オブジェクト

Lineオブジェクトにはウィンドウがありません。どちらかのサイズハンドルをクリックし、直線の端を新しい位置にドラッグすることで、長さや回転を変更できます。また、Lineオブジェクトの名称ラベル、または直線周辺をクリックして、マウスを押し続けながら、Lineオブジェクト全体を画面上の新しい位置にドラッグしてください。位置が決まったら、マウスを離してください。Lineオブジェクトが画面上の新しい位置にきます。



### Step 3: Lineオブジェクトのプロパティの設定

Lineオブジェクトプロパティの設定を行います。プロパティを設定するには、変更するプロパティの設定値フィールドをクリックし、新規の値を入力してください。あるいは、ドロップダウンリストが表示されている場合は、リスト中の該当項目をクリックしてください。

次に、Lineオブジェクトで一般的に使われるプロパティのいくつかを示します。その他、さまざまなビジョンオブジェクトの多くで使われるAbortSeqOnFail, Graphicsなどのプロパティの詳細は、「Vision Guide 7.0プロパティ&リザルトリファレンス」、および本章の「Lineオブジェクトプロパティ」を参照してください。

<b>Name プロパティ</b>	新規に作成されたLineオブジェクトに、デフォルトとして与えられるオブジェクト名は“Linexx”です。xxは、1つのビジョンシーケンス内で複数のLineオブジェクトを用いるとき、それらのLineオブジェクトで区別するための番号です。ビジョンシーケンスで1番目に用いられるLineオブジェクトには、“Line01”というオブジェクト名がデフォルトで設定されます。オブジェクト名を変更したい場合は、Nameプロパティの設定値フィールドをクリックして、変更後の名称を入力し、リターンキーを押してください。画面上でLineオブジェクト名を表示する場所は、すべて変更後の名称に変わります。
<b>StartPointObject プロパティ</b>	シーケンスの前の方で行われるオブジェクトの1つに設定 実行時のLineオブジェクトの始点を決定します。
<b>EndPointObject プロパティ</b>	シーケンスの前の方で行われるオブジェクトの1つに設定 実行時のLineオブジェクトの終点を決定します。

角度の出力方式として、AngleMode プロパティを、“2-UseAngleBase”にすることで、Directed および AngleBase プロパティの設定により角度の出力形式を指定することができます。Directed プロパティは Line オブジェクトの配置方向に依存した角度であるかどうかを指定します。AngleBase で指定した角度を基準として、角度を出力します。たとえば、ティーチング時に指定した角度を中心にして計測を行うとき、AngleBase にこの角度を指定することで、この角度を中心として計測された角度が出力されます。AngleMode プロパティを“1-Default”に設定した場合は 0 から 360 度の範囲で角度が出力されます。

(この場合は、下記のように設定した場合と同じ動作となります。)

AngleMode プロパティ: 2-UseAngleBase  
AngleBase: 0  
Directed: True)

## Step 4: Lineオブジェクトの実行とリザルトの確認

Lineオブジェクトを実行するには、次のようにしてください。

実行パネルのオブジェクトの<Run>ボタンをクリックしてください。StartPointObjectプロパティまたはEndPointObjectプロパティのどちらかがScreenでない場合、それぞれのオブジェクトが最初に実行されます。例えば、StartPointObjectがBlobオブジェクトなら、直線の始点の位置を定義するためにブローブが最初に実行されます。

Lineオブジェクトのリザルトが表示されます。この時点で確認すべき主要なリザルトは次のとおりです。

Lengthリザルト	直線のmm単位の長さ  定義されるLengthの直線を含むシーケンスに関連したキャリブレーションが必ずあります。
PixelX1リザルト	直線の両端のXY位置 (単位: ピクセル)
PixelY1リザルト	
PixelX2リザルト	
PixelY2リザルト	
PixelLengthリザルト	直線のピクセル単位の長さ

### 6.2.17 Point オブジェクト (ポイント)

#### Pointオブジェクトの概要

Pointオブジェクトは、点を定義するときに使います。Pointオブジェクトは、オブジェクト単独で使うよりも、Lineオブジェクトの中点を定義するために使うなど、通常、他のオブジェクトと一緒に使われます。

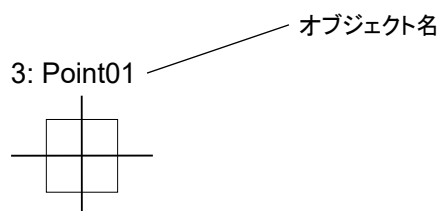
Pointオブジェクトは、下記のように、PolarオブジェクトやLineオブジェクトから参照した位置を定義するときに、よく使われます。

**Polarオブジェクト** Polarオブジェクトの中心 (CenterXおよびCenterYプロパティ)を定義するときに、Pointオブジェクトが使われます。

**Lineオブジェクト** 直線の始点, 中心点, 終点, 2直線の交点を定義するときに、Pointオブジェクトが使われます。

#### Pointオブジェクトのレイアウト

Pointオブジェクトのレイアウトは、十字線のような外観をしています。Pointオブジェクトをクリックし、変更したい位置にドラッグして、位置を変更することができます。ほとんどの場合、他のオブジェクトに付随しているので、関連するオブジェクトの位置に基づいて位置決めされます。



Point オブジェクトのレイアウト

## Pointオブジェクトのプロパティ

下の一覧は、Pointオブジェクトのプロパティを簡単に説明した概略です。各リザルトの詳細は、「Vision Guide 7.0 プロパティ&リザルトリファレンス」を参照してください。

プロパティ	説明
AbortSeqOnFail	オブジェクトの実行に失敗した (不良判定になった)とき、直ちにシーケンス全体を中断し、それ以降のオブジェクトを処理しないように指定 デフォルト: False
AngleObject	出力する角度をもつオブジェクトを指定 デフォルト: Screen
AngleObjectResult	AngleObject プロパティが使用するリザルトを指定
CalRobotPlacePos	設計時とプログラム実行時に RobotPlacePos をキャリブレーションする
Caption	見出しをPointオブジェクトに割りあてる デフォルト: 空欄
CenterPntObjResult	CenterPointObjectプロパティから使用するリザルトを指定 Allを指定すると、指定されたビジョンオブジェクトリザルトのすべて (NumberFound)にPointオブジェクトが適用されます。
CenterPntOffsetX	サーチウィンドウの中心をCenterPointObjectプロパティで位置決めしたあとに、Xオフセットを設定、または返す
CenterPntOffsetY	サーチウィンドウの中心をCenterPointObjectプロパティで位置決めしたあとに、Yオフセットを設定、または返す
CenterPntRotOffset	中心点のXYオフセット値 (CenterPntOffsetX, CenterPntOffsetY) を、CenterPointObjectのAngleリザルトにしたがって回転するか否かを指定
CenterPointObject	オブジェクトの中心点として用いられる位置を指定 このプロパティを“Screen”に設定すると画面上の任意の位置にオブジェクトを配置できますが、他のビジョンオブジェクトを指定した場合、そのオブジェクトのPixelX, PixelYリザルトに中心点が設定されます。
CoordObject	リザルトをコピーする Coordinates オブジェクトを指定 コピー処理はオブジェクトが実行される時に行われ、Decisionの分岐機能などで実行されなかった場合、コピー処理は行われません デフォルト: None
CurrentResult	オブジェクトウィンドウのリザルトリストに表示するリザルト、または、1つのサーチウィンドウ内にある対象物などの複数のリザルトを、システムが検出する設定のときにデータを返すリザルトを定義
Description	ユーザー指定の説明を設定 デフォルト: 空欄
Enabled	このオブジェクトを実行するかを指定 デフォルト: True

プロパティ	説明
FailColor	オブジェクトの検出不良時の色を選択 デフォルト: 1 – Red
Frame	使用するフレームを指定 デフォルト: None
FrameResult	Frameの何番目のリザルトを使用するかを指定
Graphics	表示するグラフィックスを指定 デフォルト: 1 – All
LabelBackColor	オブジェクトラベルの背景の色を設定 デフォルト: Transparent (透明)
LineObj1Result	LineObject1プロパティに設定されたオブジェクトのどのリザルトを使用するかを設定
LineObj2Result	LineObject2プロパティに設定されたオブジェクトのどのリザルトを使用するかを設定
LineObject1	Pointオブジェクトで、直線の中点を定義するときに、そのLineオブジェクトを指定  また、2つの直線の交点を定義するときに、1番目のLineオブジェクトを指定します。 デフォルト: none
LineObject2	Pointオブジェクトで、2つの直線の交点を定義するときは、2番目のLineオブジェクトを指定 デフォルト: none
Name	Pointオブジェクトに独自のオブジェクト名を割りあてる デフォルト: Point01
PassColor	オブジェクトの検出良時の色を選択 デフォルト: LightGreen
PassType	オブジェクト検出時の良否判定の条件を定義 デフォルト: SomeFound
PointType	LineObject1やLineObject2で、直線の中点や交点を指定 デフォルト: Screen
X	ピクセル座標系におけるPointオブジェクトのX座標
Y	ピクセル座標系におけるPointオブジェクトのY座標

### Pointオブジェクトのリザルト

下の一覧は、Pointオブジェクトリザルトを簡単に説明した概略です。各リザルトの詳細は、「Vision Guide 7.0 プロパティ&リザルトリファレンス」を参照してください。

リザルト	説明
Angle	検出されたポイントの角度を返す
CameraX	カメラ座標系における、検出オブジェクトのX座標を返す
CameraY	カメラ座標系における、検出オブジェクトのY座標を返す
CameraXYU	実行時のみ カメラ座標系における検出オブジェクト位置のCameraX座標, CameraY座標, CameraU座標を返す
ColorValue	ピクセルのグレイスケール値, カラー値を返す
NumberFound	検出されたポイントの数を返す
Found	オブジェクトが検出されたかどうかを返す (Pointオブジェクトは、付随するビジョンオブジェクトが検出されなかったとき、検出されません。)
Passed	検出結果が良か否かを返す
PixelX	ピクセル座標系における、検出オブジェクトのX座標を返す
PixelY	ピクセル座標系における、検出オブジェクトのY座標を返す
PixelXYU	実行時のみ 検出オブジェクト位置のPixelX座標, PixelY座標, PixelU座標をピクセルで返す
RobotX	ロボット座標系における、検出オブジェクトのX座標を返す
RobotY	ロボット座標系における、検出オブジェクトのY座標を返す
RobotU	ロボット座標系における、検出オブジェクトの回転量を返す (Pointオブジェクトは回転しないため、回転量は常に“0”です。)
RobotXYU	実行時のみ ロボット座標系における、検出オブジェクト位置のX座標, Y座標, 回転量を返す (Pointオブジェクトは回転しないため、回転量は常に“0”です。)

### Pointオブジェクトの基礎知識

Pointオブジェクトは、画面上の位置やLineオブジェクト、または他のビジョンオブジェクトに付随して得られた位置に対して、ユーザーがしるしをつけるために考えられました。

ここでは、Pointオブジェクトを理解するための、2つの基本事項について説明します。

1. Pointオブジェクトの位置を定義する
2. 他のオブジェクトの参照位置にPointオブジェクトを使用する

### Pointオブジェクトの位置を定義する

Pointオブジェクトの位置は、Pointオブジェクトを操作する画面上の位置、あるいは、Lineオブジェクトの中点や、2つのLineオブジェクト間の交点を基準として決定することができます。

PointTypeプロパティによって、Pointオブジェクト位置が基準とする点が定義されます。つまり、PointTypeプロパティは、“Screen”、“Midpoint”、“Intersection”の選択項目の中からいずれかを選んで設定することができます。

### PointTypeプロパティが“Screen”に設定されている場合

Pointオブジェクトの初期作成時には、PointTypeにはデフォルトの値である“Screen”が設定され、Pointオブジェクトの位置は、Pointオブジェクトが置かれている画面上の位置を基準とします。

PointTypeプロパティが“Screen”に設定されている場合には、Pointオブジェクトの位置をマウスで移動したり、XプロパティやYプロパティの値を変更しない限り、Pointオブジェクトの位置は変更されません。

### PointTypeプロパティを“Midpoint”に設定する方法

Pointオブジェクトの位置を直線の中点として定義したい場合は、PointTypeプロパティを“Midpoint”に設定してください。

ただし、PointTypeプロパティを“Midpoint”に設定する前に、どの直線を使うか、まず定義しなければなりませんので注意してください。Vision Guide 7.0は、どの直線で中点を計算するかについて情報を持っていません。

LineObject1プロパティは、中点の計算対象となるLineオブジェクトを定義し、その中点をPointオブジェクトの位置として設定するように定義します。

ビジョンシーケンスステップ実行リスト内の、Pointオブジェクトの前にある有効Lineオブジェクトは、LineObject1プロパティのドロップダウンリストから選択することができます。

実際には、ビジョンシーケンスステップ実行リスト内で、どのLineオブジェクトがPointオブジェクトの前にあるかが自動的にチェックされ、これらのLineオブジェクトだけがLineObject1プロパティのドロップダウンリスト中に表示されます。これによって、システム操作が簡単にできます。

LineObject1プロパティに特定のLineオブジェクトを指定せずに、PointTypeプロパティを“Midpoint”に設定しようとすると、最初にLineObject1プロパティに特定の直線を選択するよう指示するエラーメッセージが表示されます。

### PointTypeプロパティを“Intersection”に設定する方法

Pointオブジェクトの位置を2直線の交点として定義したい場合は、まず、2つの直線を定義する必要があります。2つの直線の定義は、LineObject1プロパティとLineObject2プロパティで行います。

LineObject1プロパティとLineObject2プロパティには、それぞれ異なるLineオブジェクトを定義しなければなりません。各プロパティにLineオブジェクトが定義されると、

PointTypeプロパティに“Intersection”を設定することができます。この設定をすることによって、Pointオブジェクトの位置は、LineObject1プロパティとLineObject2プロパティで定義された2つの直線の交点となるように規定されます。

まず、ビジョンシーケンスステップ実行リスト内のPointオブジェクトの前にある有効なLineオブジェクトが、LineObject1プロパティのドロップダウンリストから選択されて、2直線交点を形成する1番目の直線として規定されます。

次に、ビジョンシーケンスステップ実行リスト内のPointオブジェクトの前にある残りの有効なLineオブジェクトが、LineObject2プロパティのドロップダウンリストから選択されて、2直線の交点を形成する2番目の直線となります。

Vision Guide 7.0は自動的に、LineObject1プロパティ、またはLineObject2プロパティの値として設定するのに有効なLineオブジェクトだけを、関連するドロップダウンリスト中に表示します。

LineObject1プロパティとLineObject2プロパティの両方に特定のLineオブジェクトを規定せずに、PointTypeプロパティに“Intersection”を設定しようとすると、エラーメッセージが表示され、PointTypeプロパティに“Intersection”を設定する前に、まずLineObject1プロパティあるいはLineObject2プロパティに特定の直線を定義するように指示します。

メッセージ内容は、どのLineオブジェクトが定義されていないかによって異なります。



直線の交点によって位置を定義する方法は、物体の中心などを計算するのに便利です。例えば、長方形の物体を想定してみてください。4つの頂点を検出すると、長方形の中心で交わる2つの対角線を作成することができます。この中心点に位置するPointオブジェクトは、長方形の中心に位置します。

### 他のオブジェクトの参照位置にPointオブジェクトを使用する

Pointオブジェクトの主要な目的は、LineオブジェクトやPolarオブジェクトなど、他のオブジェクトの基準位置として機能することです。

つまり、Pointオブジェクトの位置は、LineオブジェクトやPolarオブジェクトの基準の位置となります。

このような強力な機能によって、2直線の交点として定義されるPointオブジェクトが物体の中心を計算し、それをLineオブジェクトの終点として使い、2つの物体の中心間の距離を計算する、というような場面を実現することができます。

### Pointオブジェクトの位置をPolarオブジェクトの基準位置として用いる方法

Polarオブジェクトは、ポーラサーチの基準となるCenterPoint位置を必要とします。Polarオブジェクトを初めてお使いの場合は、画面上の点をCenterPointプロパティの基準点としてご使用になるかもしれません。しかし、検索対象物は移動するため、Polarオブジェクトは検索対象物のCenterPointと連動して移動することになり、すぐにお気づきになるでしょう。

このような場面で、Pointオブジェクトのような他のビジョンオブジェクトからのXY位置をPolarオブジェクトに適用できる機能が大きなパワーを発揮します。

PolarオブジェクトのCenterPointに、Pointオブジェクトの位置を適用したい場合がよくありますが、適用できるのは、主に次の2つの状況においてです。

1. CenterPointが直線の中心点として定義されている場合
2. CenterPointが2直線の交点として定義されている場合

例えば、直線の中心点を検出してから、この中心点からポーラサーチを開始したい場合や、ポーラサーチのCenterPointの基準点を、2直線の交点としたい場合です。

Pointオブジェクトと一緒にPolarオブジェクトを使用するのは、非常に一般的な使用方法です。

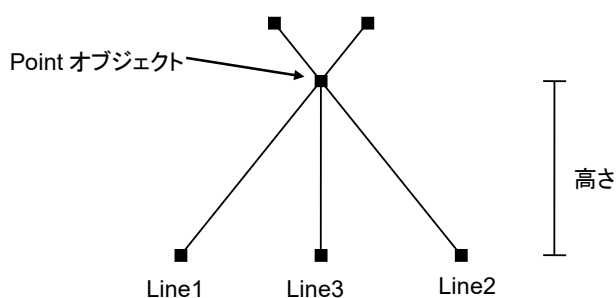
### Pointオブジェクトの位置をLineオブジェクトの基準点位置として用いる方法

直線は始点と終点を必要とします。多くの場合、直線の始点と終点の位置は、BlobオブジェクトやCorrelationオブジェクトなどの他のビジョンオブジェクトのXY位置リザルトに基づいています。

しかし、直線の基準位置として、Pointオブジェクトの位置を利用することもできます。直線の始点と終点の両方、または、直線の始点か終点を1つだけPointオブジェクトに基づいて定義することもできます。

Pointオブジェクトの一般的な使用方法のひとつに、Pointオブジェクトを2直線の交点の基準とする方法があります。

下図は、さまざまな高さで交わる2つのLineオブジェクトの例を示しています。直線3は、交点の高さを計算するのに用いられます。Pointオブジェクトは、Line1とLine2の交点の位置に表されています。



Line1とLine2の交点として定義されるPointオブジェクト




## Pointオブジェクトの使用方法

次の数ページにわたり、Pointオブジェクトの作成方法と使用方法を、下記の項目について説明します。

- 新規Pointオブジェクトの作成方法
- 画面上のPointオブジェクトの位置決め
- Pointオブジェクトの関連プロパティの設定
- Pointオブジェクトのテストとリザルトの確認


次に説明するステップを開始する前に、新規ビジョンシーケンスを作成するか、使用するビジョンシーケンスを選択してください。

作業するビジョンシーケンスがない場合は  <新規シーケンス>ボタンをクリックすると、新規ビジョンシーケンスが作成できます。

また、すでにビジョンシーケンスを作成済みの場合は、Vision Guideウィンドウのシーケンスツリーをクリックして、作成済みのビジョンシーケンスを選択することができます。

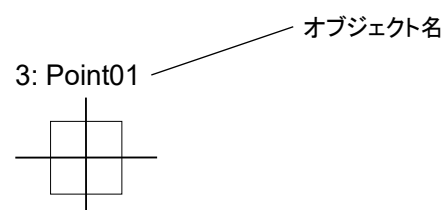
新規ビジョンシーケンスの作成方法、または作成済みビジョンシーケンスの選択方法の詳細は、「5. ビジョンシーケンス」を参照してください。

### Step 1: 新規Pointオブジェクトの作成

- (1) Vision Guideツールバー - <全ツール> -  <Point (ポイント)>ボタンをクリックしてください。
- (2) <Point (ポイント)>オブジェクトツールバーボタンの上に、Pointアイコンが表示されます。
- (3) Pointアイコンをクリックして、Vision Guideウィンドウの画像イメージ表示部にドラッグしてください。
- (4) オブジェクト名は自動的に作成されます。この例では、このシーケンスで最初に作成されたPointオブジェクトなので、“Pnt01”というオブジェクト名になります。(オブジェクト名の変更方法については後述します。)

### Step 2: Pointオブジェクトの位置決め

画面上には、下図のようなPointオブジェクトが表示されます。



新規Pointオブジェクトのレイアウト

Pointオブジェクトは、文字どおりの点であり、高さも厚さもないので、サイズ調整をすることはできません。しかし、マウスを操作したり、位置の値をXプロパティとYプロパティに設定することによって、Pointオブジェクトの位置を設定することができます。

Pointオブジェクトを作成するときには、PointTypeプロパティが“0-Screen”に設定されているので、次の手順により、Pointオブジェクトを少し移動させることができます。

- (1) Pointオブジェクトのラベル名をクリックし、マウスボタンを押し続けながら、Pointオブジェクトを画面上の新規の場所にドラッグしてください。  
適切な位置でマウスを放すと、Pointオブジェクトは、画面上の新しい位置に設定されます。

### Step 3: Pointオブジェクトプロパティの設定

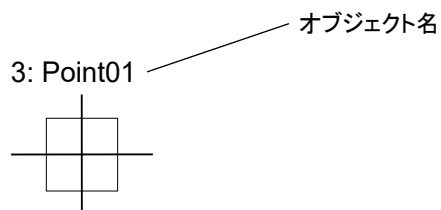
Pointオブジェクトプロパティの設定を行います。

プロパティを設定するには、関連するプロパティの設定値フィールドをクリックし、新規の値を入力してください。または、ドロップダウンリストが表示されている場合は、リスト中の該当項目をクリックしてください。

次に、Pointオブジェクトで一般的に使われるプロパティのいくつかを示します。その他、さまざまなビジョンオブジェクトで使われるAbortSeqOnFail, Graphicsなどのプロパティの詳細は、「Vision Guide 7.0プロパティ&リザルトリファレンス、または本章の「Pointオブジェクトのプロパティ」リストを参照してください。

### Pointオブジェクトのレイアウト

Pointオブジェクトのレイアウトは、画面上に十字線で表示されます。Pointオブジェクトは、位置を変更することだけができます。Pointオブジェクトをクリックし、変更したい位置にドラッグしてください。Pointオブジェクトは、ほとんどの場合が他のオブジェクトに付属しているので、関連オブジェクトの位置に基づいて位置が計算されます。



### Point オブジェクトのレイアウト

Pointオブジェクトをテストするときは、これらのプロパティを設定する必要はありません。テストでは、Pointオブジェクトの位置を、直線の中心点にも2直線の交点にも設定しないため、デフォルト値が適切であるからです。ただし、実際にPointオブジェクトで作業するのが初めての方は、この説明を参考にしてください。

**Nameプロパティ** 新規に作成されたPointオブジェクトにデフォルトとして与えられるオブジェクト名は“Pointxx”です。xxは、1つのビジョンシーケンス内で複数のPointオブジェクトを用いるとき、それらのPointオブジェクトを区別するための番号です。ビジョンシーケンスで1番目に用いられるPointオブジェクトには、“Point01”というオブジェクト名がデフォルトで設定されます。名称を変更したい場合は、Nameプロパティの設定値フィールドをクリックして、変更後の名称を入力し、リターンキーを押してください。画面上でPointオブジェクト名を表示する場所がすべて、変更後の名称を表示するようになります。

LineObject1 ("None")	<p>PointオブジェクトのPointTypeプロパティを“Midpoint”(直線の中心点を規定する)に設定すると、どの直線を使用するかを指定</p> <p>また、PointTypeプロパティが“Intersection”(2直線の交点を規定する)に設定されている場合は、このプロパティでは、2直線の1番目の直線を指定します。デフォルトは“None”に設定されています。</p>
LineObject2 ("None")	<p>PointオブジェクトのPointTypeプロパティを“Intersection”(2直線の交点を規定する)に設定すると、2直線の2番目の直線を指定</p> <p>また、交点を定義するための1番目の直線は、LineObject1プロパティによって指定されます。デフォルトは“None”に設定されています。</p>
PointType (Screen)	<p>Pointオブジェクトの位置を定義</p> <p>画面上の位置、LineObject1プロパティによって指定される直線の中心点、または、LineObject1プロパティとLineObject2プロパティによって指定される2直線の交点を基準とすることができます。</p> <p>デフォルト: Screen</p>

この例では、Lineオブジェクトが規定されていないので、LineObject1プロパティ、LineObject2プロパティ、PointTypeプロパティは、デフォルトの値を変更することができません。

なお、直線の中心点をPointオブジェクトの位置としたい場合は、LineObject1プロパティにLineオブジェクトを規定します。

また、2直線の交点をPointオブジェクトの位置としたい場合は、LineObject1プロパティに1番目のLineオブジェクトを、LineObject2プロパティに2番目のLineオブジェクトを規定します。

詳細は、前述の「Pointオブジェクトの位置を定義する」を参照してください。

AngleObjectで指定したオブジェクトの角度を、Pointオブジェクトから出力される角度とすることができます。これは、Pointオブジェクトの出力として、XY位置だけでなく、Pointオブジェクトの前段のオブジェクトの角度を、Pointオブジェクトの角度とすることで、RobotXYUなどのXY位置と角度Uの情報をまとめて取得することができるようになります。

### Step 4: Pointオブジェクトの実行とリザルトの確認

Pointオブジェクトは、実行パネルのオブジェクトの<Run>ボタンをクリックするだけで簡単に実行できます。

(1) 実行パネルのオブジェクトの<Run>ボタンをクリックしてください。

Pointオブジェクトが実行されると、リザルトが表示されますので確認してください。この時点で確認すべき主要なリザルトは、次のとおりです。

PixelXリザルト	PointオブジェクトのXY位置 (ピクセル座標系)
PixelYリザルト	PointオブジェクトのPointTypeプロパティーが中点に設定されている場合、これらのリザルトは、LineObject1プロパティーによって設定されたLineオブジェクトの中点のXY位置 (ピクセル座標系)を返す
CameraXリザルト	カメラ座標系におけるPointオブジェクトのXY座標を定義
CameraYリザルト	カメラキャリブレーションが完了していない場合は値を返さず、“no cal”を返します。
RobotXリザルト	ロボット座標系におけるPointオブジェクトのXY座標を定義
RobotYリザルト	ロボットは、このXY座標を動作先にできます。その他の変換や処理手順は必要ありません。なお、カメラキャリブレーションが完了していない場合は値を返さず、“no cal”を返します。

## 6.2.18 BoxFinder オブジェクト (長方形検出)

### BoxFinderオブジェクトの概要

BoxFinderオブジェクトは、画像イメージ中の長方形(正方形を含む)エッジの場所を特定するのに使われます。

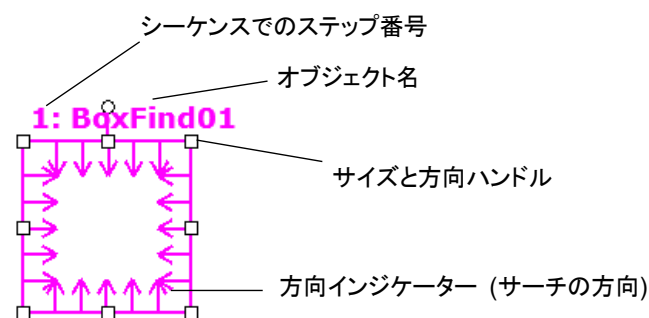
BoxFinderオブジェクトは、複数のEdgeオブジェクトを自動的に処理してエッジ位置を特定し、各エッジ位置から特定される長方形を求めます。

画像イメージ中のオブジェクトのエッジは、グレースケール値の暗から明への変化、あるいは明から暗への変化で表されます。このような変化は、数個のピクセルにわたって観察されます。

BoxFinderオブジェクトの各エッジサーチは、Polarityプロパティーによって定義された、明から暗への変化、または暗から明への変化を検出し、検出されたエッジ位置を結ぶ最適な直線を特定します。EdgeTypeプロパティーを変更してエッジペアを検出することもできます。エッジペアの場合、2つの対極のエッジを検索し、中間点を返します。

### BoxFinderオブジェクトのレイアウト

BoxFinderオブジェクトは、LineFinderのようにサーチウィンドウ内にエッジのサーチ方向を示す方向インジケターがあります。LineFinderとの違いは、サーチウィンドウの4辺に各方向の方向インジケターがあるところです。エッジ検索ラインの数はNumberOfEdgesプロパティーで指定します。Directionプロパティーで検索方向を指定できます。



### BoxFinder オブジェクトのレイアウト

BoxFinderオブジェクトは、(垂直方向、水平方向だけでなく)あらゆる方向を検索するように置くことができます。BlobオブジェクトのSearchWinType=AngledRectangleと同様に、BoxFinderオブジェクトのサーチウィンドウ回転ハンドルを使って、BoxFinderオブジェクトを目的のエッジを検出する方向に動かすことができます。

## BoxFinderオブジェクトのプロパティー

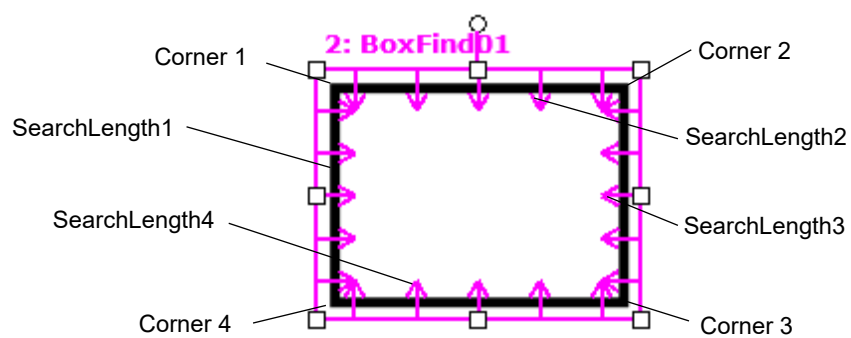
下の一覧は、BoxFinderオブジェクトプロパティーの概略です。各プロパティーの詳細は、「Vision Guide 7.0プロパティー&リザルトリファレンス」を参照してください。

プロパティー	説明
AbortSeqOnFail	オブジェクトの実行に失敗した(不良判定になった)とき、直ちにシーケンス全体を中断し、それ以降のオブジェクトを処理しないように指定 デフォルト: False
Accept	特徴が検出されたと判定するための形状スコアの値を指定 対象物が検出されたとみなされるためには、スコアの値を超えてはなりません。 デフォルト: 100
Caption	見出しをBoxFinderオブジェクトに割りあてる デフォルト: 空欄
CenterPointObject	オブジェクトの中心点として用いられる位置を指定 このプロパティーを“Screen”に設定すると画面上の任意の位置にオブジェクトを配置できますが、他のビジョンオブジェクトを指定した場合、そのオブジェクトのPixelX, PixelYリザルトに中心点が設定されます。 デフォルト: Screen
CenterPntObjResult	CenterPointObjectプロパティーから使用するリザルトを指定 Allを指定すると、指定されたビジョンオブジェクトリザルトのすべて (NumberFound) にBoxFinderオブジェクトが適用されます。 デフォルト: 1
CenterPntOffsetX	サーチウィンドウの中心をCenterPointObjectプロパティーで位置決めしたあとに、Xオフセットを設定、または返す デフォルト: 0
CenterPntOffsetY	サーチウィンドウの中心をCenterPointObjectプロパティーで位置決めしたあとに、Yオフセットを設定、または返す デフォルト: 0
CenterPntRotOffset	中心点の XY オフセット値 (CenterPntOffsetX, CenterPntOffsetY)を、CenterPointObject の Angle リザルトにしたがって回転するか否かを指定 デフォルト: False
ContrastTarget	エッジサーチのためのコントラストを設定 デフォルト: 0 (best contrast)
ContrastVariation	ContrastTargetの許容されるコントラストの変化量を選択 デフォルト: 0

プロパティ	説明
CoordObject	リザルトをコピーするCoordinatesオブジェクトを指定 コピー処理はオブジェクトが実行される時に行われ、Decisionの分岐機能などで実行されなかった場合、コピー処理は行われません デフォルト: None
CurrentResult	オブジェクトウィンドウのリザルトリストに表示するリザルト、または、1つのサーチウィンドウ内にある対象物などの複数のリザルトを、システムが検出する設定のときにデータを返すリザルトを定義
Description	ユーザー指定の説明を設定 デフォルト: 空欄
Direction	エッジ検索の方向を設定 デフォルト: InsideOut
EdgeSort	検出されたエッジ結果のソート方法を設定 デフォルト: Score
EdgeThreshold	しきい値(この値より低いエッジは無視する)を設定 デフォルト: 2
EdgeType	検索するエッジタイプ (シングル, ペア)を設定 デフォルト: 1 – Single
Enabled	このオブジェクトを実行するかを指定 デフォルト: True
FailColor	オブジェクトの検出結果が不良時の色を指定 デフォルト: Red
FittingThreshold	直線のフィッティングに使用するエッジリザルトを指定 デフォルト: 10
Frame	どの位置に置いたフレームを使用するかを指定 デフォルト: none
FrameResult	Frameの何番目のリザルトを使用するかを指定 デフォルト: 1
Graphics	表示するグラフィックを指定 デフォルト: 1 – All
LabelBackColor	オブジェクトラベルの背景の色を設定 デフォルト: Transparent (透明)
MissingEdgeType	未検出エッジの処理方法を設定 デフォルト: Interpolate
Name	BoxFinderオブジェクトに独自のオブジェクト名を割りあてる デフォルト: BoxFind01
NumberOfEdges	検出するエッジ数を設定 デフォルト: 5

プロパティ	説明
PassColor	オブジェクトの検出が良のときの色を選択 デフォルト: LightGreen
PassType	オブジェクト検出の良否判定の条件を定義 デフォルト: SomeFound
Polarity	BoxFinderオブジェクトが明から暗へ(LightToDark)の移行を検索するのか、暗から明へ(DarkToLight)の移行を検索するのかを指定 デフォルト: 1 – LightToDark
ScoreWeightContrast	コントラストによってスコアの割合を設定 デフォルト: 50
ScoreWeightStrength	エッジの強度によってスコアの割合を設定 デフォルト: 50
SearchLength	エッジをサーチする範囲の長さを定義 下記のSearchLength1~4の値を一括で設定できる
SearchLength1	図「BoxFinderオブジェクトのプロパティ、リザルトの位置関係」のSearchLength1の長さを設定
SearchLength2	図「BoxFinderオブジェクトのプロパティ、リザルトの位置関係」のSearchLength2の長さを設定
SearchLength3	図「BoxFinderオブジェクトのプロパティ、リザルトの位置関係」のSearchLength3の長さを設定
SearchLength4	図「BoxFinderオブジェクトのプロパティ、リザルトの位置関係」のSearchLength4の長さを設定
SearchWidth	エッジサーチの幅を定義 範囲: 3 ~ 99 デフォルト: 3
SearchWin	実行時のみ1コールでサーチウィンドウの左端, 上端, 高さ, 幅のパラメーターを設定、または返す
SearchWinAngle	被検索領域の角度を定義
SearchWinCenterX	被検索領域の中心のX座標値を定義
SearchWinCenterY	被検索領域の中心のY座標値を定義
SearchWinHeight	被検索領域の高さを定義 (単位: ピクセル)
SearchWinLeft	被検索領域の最左端を定義 (単位: ピクセル)
SearchWinTop	被検索領域の最上端を定義 (単位: ピクセル)
SearchWinWidth	被検索領域の幅を定義 (単位: ピクセル)
StrengthTarget	サーチするエッジの強度を設定 デフォルト: 0
StrengthVariation	StrengthTargetの変化量の合計を設定 デフォルト: 0





BoxFinder オブジェクトのプロパティ, リザルトの位置関係

## BoxFinderオブジェクトのリザルト

下の一覧は、BoxFinderオブジェクトリザルトを簡単に説明した概略です。各リザルトの詳細は、「Vision Guide 7.0プロパティ&リザルトリファレンス」を参照してください。

リザルト	説明
Angle	検出された長方形の角度を返す
CameraX	検出された長方形エッジの中心X座標をカメラ座標で返す
CameraY	検出された長方形エッジの中心Y座標をカメラ座標で返す
CameraXYU	実行時のみ カメラ座標系における検出された長方形エッジの中心位置のCameraX座標, CameraY座標, CameraU座標を返す
CameraX1	図「BoxFinderオブジェクトのプロパティ, リザルトの位置関係」のCorner1のX座標をカメラ座標系で返す
CameraY1	図「BoxFinderオブジェクトのプロパティ, リザルトの位置関係」のCorner1のY座標をカメラ座標系で返す
CameraX2	図「BoxFinderオブジェクトのプロパティ, リザルトの位置関係」のCorner2のX座標をカメラ座標系で返す
CameraY2	図「BoxFinderオブジェクトのプロパティ, リザルトの位置関係」のCorner2のY座標をカメラ座標系で返す
CameraX3	図「BoxFinderオブジェクトのプロパティ, リザルトの位置関係」のCorner3のX座標をカメラ座標系で返す
CameraY3	図「BoxFinderオブジェクトのプロパティ, リザルトの位置関係」のCorner3のY座標をカメラ座標系で返す
CameraX4	図「BoxFinderオブジェクトのプロパティ, リザルトの位置関係」のCorner4のX座標をカメラ座標系で返す
CameraY4	図「BoxFinderオブジェクトのプロパティ, リザルトの位置関係」のCorner4のY座標をカメラ座標系で返す
Contrast	検出された長方形エッジのコントラストを返す
FitError	各エッジポイントと検出された長方形との間の距離を二乗平均平方根(RMS)で返す
Found	オブジェクトが検出されたかどうか、つまり、対象物あるいはパーツがAcceptプロパティの現在の設定を超える形状スコア値を持っているかどうかの値を返す
MaxError	検出された長方形エッジからの最大ずれをピクセル長で返す
Passed	検出結果が良か否かを返す
Perimeter	検出された長方形の外周のピクセルの数を返す
PixelX	検出された長方形エッジの中心X座標をイメージ座標で返す
PixelY	検出された長方形エッジの中心Y座標をイメージ座標で返す
PixelX1	図「BoxFinderオブジェクトのプロパティ, リザルトの位置関係」のCorner1のX座標をイメージ座標で返す
PixelY1	図「BoxFinderオブジェクトのプロパティ, リザルトの位置関係」のCorner1のY座標をイメージ座標で返す


リザルト	説明
PixelX2	図「BoxFinderオブジェクトのプロパティ、リザルトの位置関係」のCorner2のX座標をイメージ座標で返す
PixelY2	図「BoxFinderオブジェクトのプロパティ、リザルトの位置関係」のCorner2のY座標をイメージ座標で返す
PixelX3	図「BoxFinderオブジェクトのプロパティ、リザルトの位置関係」のCorner3のX座標をイメージ座標で返す
PixelY3	図「BoxFinderオブジェクトのプロパティ、リザルトの位置関係」のCorner3のY座標をイメージ座標で返す
PixelX4	図「BoxFinderオブジェクトのプロパティ、リザルトの位置関係」のCorner4のX座標をイメージ座標で返す
PixelY4	図「BoxFinderオブジェクトのプロパティ、リザルトの位置関係」のCorner4のY座標をイメージ座標で返す
PixelXYU	検出された長方形エッジの中心位置のPixelX座標, PixelY座標, PixelU座標をピクセルで返す
RobotX	検出された長方形エッジの中心X座標をロボット座標で返す
RobotY	検出された長方形エッジの中心Y座標をロボット座標で返す
RobotU	検出された長方形エッジの中心U座標をロボット座標で返す
RobotXYU	実行時のみ ロボット座標系における、検出された長方形エッジの中心位置のRobotX座標, RobotY座標, RobotU座標を返す
RobotX1	図「BoxFinderオブジェクトのプロパティ、リザルトの位置関係」のCorner1のX座標をロボット座標で返す
RobotY1	図「BoxFinderオブジェクトのプロパティ、リザルトの位置関係」のCorner1のY座標をロボット座標で返す
RobotX2	図「BoxFinderオブジェクトのプロパティ、リザルトの位置関係」のCorner2のX座標をロボット座標で返す
RobotY2	図「BoxFinderオブジェクトのプロパティ、リザルトの位置関係」のCorner2のY座標をロボット座標で返す
RobotX3	図「BoxFinderオブジェクトのプロパティ、リザルトの位置関係」のCorner3のX座標をロボット座標で返す
RobotY3	図「BoxFinderオブジェクトのプロパティ、リザルトの位置関係」のCorner3のY座標をロボット座標で返す
RobotX4	図「BoxFinderオブジェクトのプロパティ、リザルトの位置関係」のCorner4のX座標をロボット座標で返す
RobotY4	図「BoxFinderオブジェクトのプロパティ、リザルトの位置関係」のCorner4のY座標をロボット座標で返す
ShowAllResults	特定のビジョンオブジェクトのリザルトすべてを、表形式でダイアログに表示 リザルトの比較が簡単にできます。
Strength	検出エッジの強度を返す

リザルト	説明
Time	オブジェクト処理に要した時間を返す (単位: ミリ秒)

### BoxFinderオブジェクトの使用方法

BoxFinderオブジェクトの作成と使用の方法を説明します。


- 新規BoxFinderオブジェクトの作成方法
- サーチウィンドウの位置とサイズの設定
- BoxFinderオブジェクトの関連プロパティの設定
- BoxFinderオブジェクトのテストとリザルトの確認
- プロパティの調整と再テスト

これらのステップを開始する前に、まず新規ビジョンシーケンスを作成するか、使用するビジョンシーケンスを選択してください。作業するビジョンシーケンスがない場合は、 <新規シーケンス>ボタンをクリックし、新規ビジョンシーケンスを作成します。

ビジョンシーケンスを作成済みの場合は、Vision Guideウィンドウのシーケンスツリーをクリックして、作成済みのビジョンシーケンスを選択します。

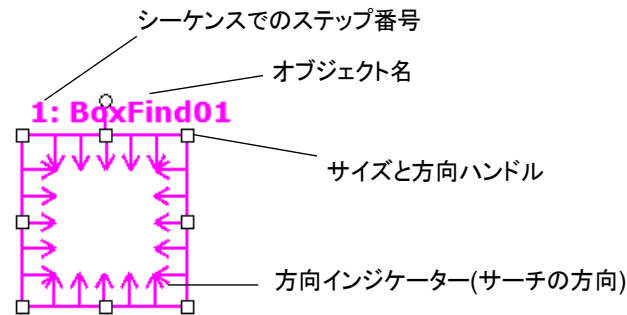
新規ビジョンシーケンスの作成方法、または作成済みビジョンシーケンスの選択方法の詳細は、「ビジョンシーケンス」を参照してください。

### Step 1: 新規BoxFinderオブジェクトの作成

- (1) Vision Guideツールバー - <全ツール> -  <Box Finder (長方形検出)>ボタンをクリックします。
- (2) 画像イメージ表示部にマウスを動かします。  
マウスポインターがBoxFinderオブジェクトアイコンに変わります。
- (3) アイコンが画像イメージ表示部の目的の位置に来るまで、マウスを動かします。
- (4) マウスを左クリックして、オブジェクトを作成します。
- (5) オブジェクト名が自動的に作成されます。  
この例では、このシーケンスで最初に作成されたBoxFinderオブジェクトなので、“BoxFind01”というオブジェクト名になります。(オブジェクト名の変更方法については後述します。)

## Step 2: サーチウィンドウの位置とサイズの設定

画面上に、下図のようなBoxFinderオブジェクトが表示されます。



### 新規BoxFinderオブジェクトのレイアウト

- (1) BoxFinderオブジェクトの名称ラベルをクリックし、マウスを押し続けながら、BoxFinderオブジェクトをドラッグして、サーチウィンドウを好きな位置に合わせます。
- (2) サイズハンドルをクリックし、マウスをドラッグすると、BoxFinderオブジェクトサーチウィンドウのサイズが変更できます。適当な大きさに調整します。

## Step 3: BoxFinderオブジェクトプロパティーの設定

BoxFinderオブジェクトプロパティーの設定を行います。

プロパティーを設定するには、関連するプロパティーの設定値フィールドをクリックし、新規の値を入力します。または、ドロップダウンリストが表示されている場合は、リストから選択します。

次に、BoxFinderオブジェクトで一般的に使われるプロパティーのいくつかを示します。その他、さまざまなビジョンオブジェクトの多くで使われるAbortSeqOnFail, Graphicsなどのプロパティーの詳細は、「Vision Guide 7.0プロパティー&リザルトリファレンス」を参照してください。

**Nameプロパティー**      新規に作成されたBoxFinderオブジェクトに、デフォルトとして与えられるオブジェクト名は“BoxFind\*\*”です。\*\*は、1つのビジョンシーケンス内で複数のBoxFinderオブジェクトを用いるとき、それらのBoxFinderオブジェクトを区別するための番号です。

ビジョンシーケンスで1番目に用いられるBoxFinderオブジェクトには、“BoxFind01”というオブジェクト名がデフォルトで設定されます。

名称を変更したい場合は、Nameプロパティーの設定値フィールドをクリックして、変更後の名称を入力し、リターンキーを押してください。画面上でBoxFinderオブジェクト名を表示する場所がすべて、変更後の名称を表示するようになります。

**EdgeType(シングル)**      検索するエッジの種類を選択

エッジペアを設定した場合、それぞれの方向のエッジを検索し、中間点を位置として検出します。

**NumberOfEdges(5)**      エッジを検索するために、サーチウィンドウの各辺で5個のエッジを検索

Polarity (LightToDark)	エッジ位置を「明から暗へ(LightToDark)」検索 「暗から明へ(DarkToLight)」のエッジを検索する場合は、 Polarityを変更してください。
---------------------------	---

#### Step 4: BoxFinderオブジェクトの実行とリザルトの確認

BoxFinderオブジェクトを実行する方法を説明します。

実行パネルのオブジェクトの<Run>ボタンをクリックします。

BoxFinderオブジェクトのリザルトが表示されます。この時点で確認すべき主要なリザルトは次のとおりです。

Angleリザルト	検出された長方形の角度をイメージ座標で返す
MaxErrorリザルト	検出された長方形エッジからの最大ずれを返す (単位: ピクセル)
PixelXリザルト PixelYリザルト	検出された長方形エッジの中心XY座標位置をイメージ座標で返す
CameraXリザルト CameraYリザルト	検出された長方形エッジの中心XY座標位置をカメラ座標で返す XY座標位置がキャリブレーションされていない場合は、“no cal” が返されます。
RobotXリザルト RobotYリザルト	検出された長方形エッジの中心XY座標位置をロボット座標で返す XY座標位置がキャリブレーションされていない場合は、“no cal” が返されます。

##### 6.2.19



#### CornerFinder オブジェクト (コーナー検出)

##### CornerFinderオブジェクトの概要

CornerFinderオブジェクトは、画像イメージ中の2直線の交点(コーナー)の場所を特定するのに使われます。

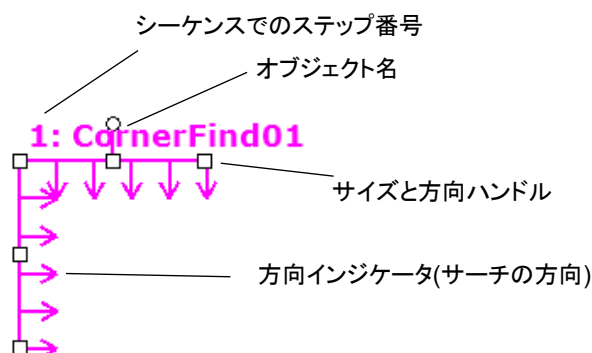
CornerFinderオブジェクトは、複数のEdgeオブジェクトを自動的に処理してエッジ位置を特定し、各エッジ位置から特定される2直線の交点を求めます。

画像イメージ中のオブジェクトのエッジは、グレイスケール値の暗から明への変化、あるいは明から暗への変化で表されます。このような変化は、数個のピクセルにわたって観察されます。

CornerFinderオブジェクトの各エッジサーチは、Polarityプロパティーによって定義された、明から暗への変化、または暗から明への変化を検出し、検出されたエッジ位置を結ぶ最適な直線を特定します。EdgeTypeプロパティーを変更してエッジペアを検出することもできます。エッジペアの場合、2つの対極のエッジを検索し、中間点を返します。

### CornerFinderオブジェクトのレイアウト

CornerFinderオブジェクトには、LineFinderのようにサーチウィンドウ内にエッジのサーチ方向を示す方向インジケータがあります。LineFinderとの違いはサーチウィンドウの2辺に各方向の方向インジケータがあるところです。エッジ検索ラインの数はNumberOfEdgesプロパティで指定します。Directionプロパティで検索方向を指定できます。



### CornerFinder オブジェクトのレイアウト

CornerFinderオブジェクトは、(垂直方向, 水平方向だけでなく)あらゆる方向を検索するように置くことができます。BlobオブジェクトのSearchWinType=AngledRectangleと同様に、CornerFinderオブジェクトのサーチウィンドウ回転ハンドルを使って、CornerFinderオブジェクトを目的のエッジを検出する方向に動かすことができます。

### CornerFinderオブジェクトのプロパティ

下の一覧は、CornerFinderオブジェクトプロパティの概略です。各プロパティの詳細は、「Vision Guide 7.0プロパティ&リザルトリファレンス」を参照してください。

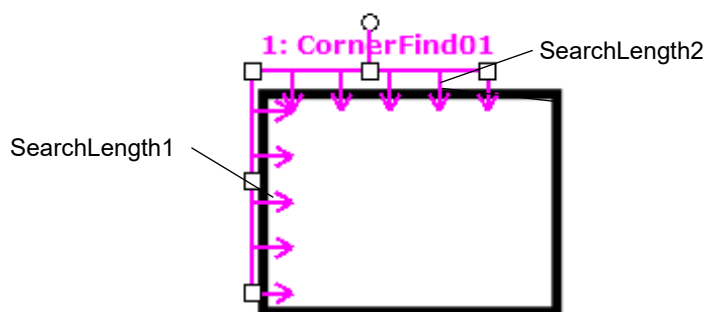
プロパティ	説明
AbortSeqOnFail	オブジェクトの実行に失敗した(不良判定になった)とき、直ちにシーケンス全体を中断し、それ以降のオブジェクトを処理しないように指定 デフォルト: False
Accept	特徴が検出されたと判定するための形状スコアの値を指定 対象物が検出されたとみなされるためには、スコアの値を超えなくてはなりません。 デフォルト: 100
Caption	見出しをCornerFinderオブジェクトに割りあてる デフォルト: 空欄
CenterPointObject	オブジェクトの中心点として使用する位置を指定 このプロパティを“Screen”に設定すると画面上の任意の位置にオブジェクトを配置できます。他のビジョンオブジェクトを指定した場合、そのオブジェクトのPixelX, PixelYリザルトに中心点が設定されます。 デフォルト: Screen

プロパティ	説明
CenterPntObjResult	CenterPointObjectプロパティから使用するリザルトを指定 Allを指定すると、指定されたビジョンオブジェクトリザルトのすべて (NumberFound)にCornerFinderオブジェクトが適用されます。 デフォルト: 1
CenterPntOffsetX	サーチウィンドウの中心をCenterPointObjectプロパティで位置決めしたあとに、Xオフセットを設定、または返す デフォルト: 0
CenterPntOffsetY	サーチウィンドウの中心をCenterPointObjectプロパティで位置決めしたあとに、Yオフセットを設定、または返す デフォルト: 0
CenterPntRotOffset	中心点の XY オフセット値 (CenterPntOffsetX, CenterPntOffsetY)を、CenterPointObject の Angle リザルトにしたがって回転するか否かを指定 デフォルト: False
ContrastTarget	エッジサーチのためのコントラストを設定 デフォルト: 0 (best contrast)
ContrastVariation	ContrastTargetの許容されるコントラストの変化量を選択 デフォルト: 0
CoordObject	リザルトをコピーするCoordinatesオブジェクトを指定 コピー処理はオブジェクトが実行される時に行われ、Decisionの分岐機能などで実行されなかった場合、コピー処理は行われません デフォルト: None
CurrentResult	オブジェクトウィンドウのリザルトリストに表示するリザルト または、1つのサーチウィンドウ内にある対象物などの複数のリザルトを、システムが検出する設定のときにデータを返すリザルトを定義
Description	ユーザー指定の説明を設定 デフォルト: 空欄
Direction	エッジ検索の方向を設定 デフォルト: InsideOut
EdgeSort	検出されたエッジ結果のソート方法を設定 デフォルト: Score
EdgeThreshold	しきい値(この値より低いエッジは無視する)を設定 デフォルト: 2
EdgeType	検索するエッジタイプ(シングル, ペア)を設定 デフォルト: 1 – Single
Enabled	このオブジェクトを実行するかを指定 デフォルト: True



プロパティ	説明
FailColor	オブジェクトの検出結果が不良時の色を指定 デフォルト: Red
FittingThreshold	直線のフィッティングに使用するエッジリザルトを指定 デフォルト: 10
Frame	どの位置に置いたフレームを使用するかを指定 デフォルト: none
FrameResult	Frameの何番目のリザルトを使用するかを指定 デフォルト: 1
Graphics	表示するグラフィックを指定 デフォルト: 1 – All
LabelBackColor	オブジェクトラベルの背景の色を設定 デフォルト: Transparent (透明)
MissingEdgeType	未検出エッジの処理方法を設定 デフォルト: Interpolate
Name	CornerFinderオブジェクトに独自のオブジェクト名を割りあてる デフォルト: CornerFind01
NumberOfEdges	検出するエッジ数を設定 デフォルト: 5
PassColor	オブジェクトの検出が、良のときの色を選択 デフォルト: LightGreen
PassType	オブジェクト検出時の良否判定の条件を定義 デフォルト: SomeFound
Polarity	CornerFinderオブジェクトが、明から暗へ(LightToDark)の移行を検索するのか、暗から明へ(DarkToLight)の移行を検索するのかを指定 デフォルト: 1 – LightToDark
ScoreWeightContrast	コントラストによってスコアの割合を設定 デフォルト: 50
ScoreWeightStrength	エッジの強度によってスコアの割合を設定 デフォルト: 50
SearchLength	エッジをサーチする範囲の長さを定義 下記のSearchLength1,2の値を一括で設定できる
SearchLength1	図「CornerFinderオブジェクトのプロパティの位置関係」のSearchLength1の長さを設定
SearchLength2	図「CornerFinderオブジェクトのプロパティの位置関係」のSearchLength2の長さを設定
SearchWidth	エッジサーチの幅を定義 範囲: 3 ~ 99 デフォルト: 3

プロパティ	説明
SearchWin	実行時のみ 1コールでサーチウィンドウの左端, 上端, 高さ, 幅のパラメータを設定、または返す
SearchWinAngle	被検索領域の角度を定義
SearchWinCenterX	被検索領域の中心のX座標値を定義
SearchWinCenterY	被検索領域の中心のY座標値を定義
SearchWinHeight	被検索領域の高さを定義 (単位: ピクセル)
SearchWinLeft	被検索領域の最左端を定義 (単位: ピクセル)
SearchWinTop	被検索領域の最上端を定義 (単位: ピクセル)
SearchWinWidth	被検索領域の幅を定義 (単位: ピクセル)
StrengthTarget	サーチするエッジの強度を設定 デフォルト: 0
StrengthVariation	StrengthTargetの変化量の合計を設定 デフォルト: 0



CornerFinder オブジェクトのプロパティの位置関係

### CornerFinderオブジェクトのリザルト

下の一覧は、CornerFinderオブジェクトリザルトを簡単に説明した概略です。各リザルトの詳細は、「Vision Guide 7.0プロパティ&リザルトリファレンス」を参照してください。

リザルト	説明
Angle	検出されたコーナーの角度を返す
CameraX	検出コーナーのX座標をカメラ座標で返す
CameraY	検出コーナーのY座標をカメラ座標で返す
CameraXYU	実行時のみ カメラ座標系における検出コーナー位置のCameraX座標, CameraY座標, CameraU座標を返す
Contrast	検出エッジのコントラストを返す
FitError	各エッジポイントと検出されたコーナーとの間の距離を二乗平均平方根(RMS)で返す


リザルト	説明
Found	オブジェクトが検出されたかどうか、つまり、対象物あるいはパーツがAcceptプロパティの現在の設定を超える形状スコア値を持っているかどうかの値を返す
MaxError	検出直線エッジからの最大ずれをピクセル長で返す
Passed	検出結果が良か否かを返す
PixelX	検出コーナーのX座標をイメージ座標で返す
PixelY	検出コーナーのY座標をイメージ座標で返す
PixelXYU	実行時のみ 検出コーナー位置のPixelX座標, PixelY座標, PixelU座標をピクセルで返す
RobotX	検出コーナーのX座標をロボット座標で返す
RobotY	検出コーナーのY座標をロボット座標で返す
RobotU	検出コーナーのU座標をロボット座標で返す
RobotXYU	実行時のみ ロボット座標系における、検出コーナー位置のRobotX座標, RobotY座標, RobotU座標を返す
ShowAllResults	特定のビジョンオブジェクトのリザルトすべてを、表形式でダイアログに表示 リザルトの比較が簡単にできます。
Strength	検出エッジの強度を返す
Time	オブジェクト処理に要した時間を返す (単位: ミリ秒)

### CornerFinderオブジェクトの使用方法

CornerFinderオブジェクトの作成と使用の方法を説明します。

- 新規CornerFinderオブジェクトの作成方法
- サーチウィンドウの位置とサイズの設定
- CornerFinderオブジェクトの関連プロパティの設定
- CornerFinderオブジェクトのテストとリザルトの確認
- プロパティの調整と再テスト


これらのステップを開始する前に、まず新規ビジョンシーケンスを作成するか、使用する

ビジョンシーケンスを選択してください。作業するビジョンシーケンスがない場合は、 <新規シーケンス> ボタンをクリックし、新規ビジョンシーケンスを作成します。

ビジョンシーケンスを作成済みの場合は、Vision Guideウィンドウのシーケンスツリーをクリックして、作成済みのビジョンシーケンスを選択します。

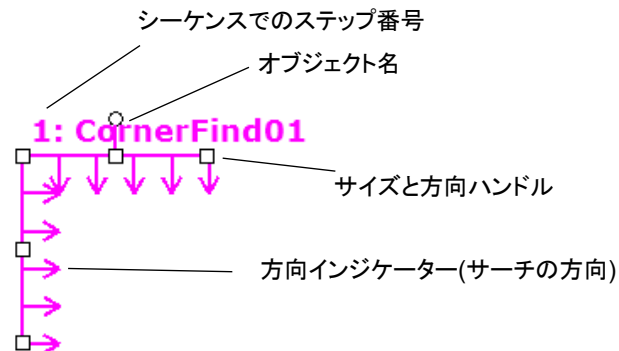
新規ビジョンシーケンスの作成方法、または作成済みビジョンシーケンスの選択方法の詳細は、「ビジョンシーケンス」を参照してください。

### Step 1: 新規CornerFinderオブジェクトの作成

- (1) Vision Guideツールバー - <全ツール> -  <CornerFinder (コーナー検出)> ボタンをクリックします。
- (2) 画像イメージ表示部にマウスを動かします。  
マウスポインターがCornerFinderオブジェクトアイコンに変わります。
- (3) アイコンが画像イメージ表示部の目的の位置に来るまで、マウスを動かします。
- (4) マウスを左クリックして、オブジェクトを作成します。
- (5) オブジェクト名が自動的に作成されます。  
この例では、このシーケンスで最初に作成されたCornerFinderオブジェクトなので、“CornerFind01”というオブジェクト名になります。(オブジェクト名の変更方法については後述します。)

## Step 2: サーチウィンドウの位置とサイズの設定

画面には、下図のようなCornerFinderオブジェクトが表示されます。



### 新規CornerFinderオブジェクトのレイアウト

- (1) CornerFinderオブジェクトの名称ラベルをクリックし、マウスを押し続けながら、CornerFinderオブジェクトをドラッグして、サーチウィンドウを好きな位置に合わせます。
- (2) サイズハンドルをクリックして、マウスをドラッグし、CornerFinderオブジェクトサーチウィンドウのサイズを変更します。適当な大きさに調整します。

## Step 3: CornerFinderオブジェクトプロパティーの設定

CornerFinderオブジェクトプロパティーの設定を行います。

プロパティーを設定するには、関連するプロパティーの設定値フィールドをクリックし、新規の値を入力します。または、ドロップダウンリストが表示されている場合は、リストから選択します。

次に、CornerFinderオブジェクトで一般的に使われるプロパティーのいくつかを示します。その他、さまざまなビジョンオブジェクトの多くで使われるAbortSeqOnFail, Graphicsなどのプロパティーの詳細は、Vision Guide 7.0プロパティー&リザルトリファレンスを参照してください。

**Nameプロパティー** 新規に作成されたCornerFinderオブジェクトに、デフォルトとして与えられるオブジェクト名は“CornerFind\*\*”です。\*\*は、1つのビジョンシーケンス内で複数のCornerFinderオブジェクトを用いるとき、それらのCornerFinderオブジェクトを区別するための番号です。

ビジョンシーケンスで1番目に用いられるCornerFinderオブジェクトには、“CornerFind01”というオブジェクト名がデフォルトで設定されます。

名称を変更したい場合は、Nameプロパティーの設定値フィールドをクリックして、変更後の名称を入力し、リターンキーを押してください。画面上でCornerFinderオブジェクト名を表示する場所がすべて、変更後の名称を表示するようになります。

**EdgeType(シングル)** 検索するエッジの種類を選択

エッジペアを設定した場合、それぞれの方向のエッジを検索し、中間点を位置として検出します。

NumberOfEdges(5)	エッジを検索するために、サーチウィンドウの各辺で5個のエッジを検索
Polarity (LightToDark)	エッジ位置を「明から暗へ(LightToDark)」検索 「暗から明へ(DarkToLight)」のエッジを検索する場合は、 “Polarity”を変更してください。

### Step 4: CornerFinderオブジェクトの実行とリザルトの確認

CornerFinderオブジェクトを実行する方法を説明します。

実行パネルのオブジェクトの<Run>ボタンをクリックします。

CornerFinderオブジェクトのリザルトが表示されます。この時点で確認すべき主要なリザルトは次のとおりです。

Angleリザルト	検出されたコーナーの角度をイメージ座標で返す
MaxErrorリザルト	検出された直線からの最大ずれを返す(単位: ピクセル)
PixelXリザルト PixelYリザルト	検出されたコーナーのXY座標位置をイメージ座標で返す
CameraXリザルト CameraYリザルト	検出されたコーナーのXY座標位置をカメラ座標で返す XY座標位置がキャリブレーションされていない場合は、“no cal” が返されます。
RobotXリザルト RobotYリザルト	検出されたコーナーのXY座標位置をロボット座標で返す XY座標位置がキャリブレーションされていない場合は、“no cal” が返されます。

## 6.2.20 Contour オブジェクト (輪郭抽出)

### Contourオブジェクトの概要

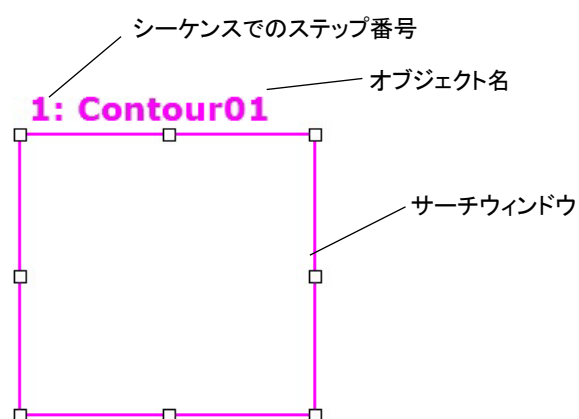
Contourオブジェクトは、ワークの輪郭に沿う軌跡を出力します。ロボットのハンドをワークの輪郭に沿わせて動かしたい場合に、画像情報から簡単に経路を取得できます。

Contourオブジェクトには、軌跡の取得方法が3種類あり、用途に合わせて使い分けが可能です。取得方法は、**ContourMode**プロパティーを変えることで切り替わり、各モードでGUIや有効なプロパティーも異なります。

各モードの特徴を簡単に説明します。

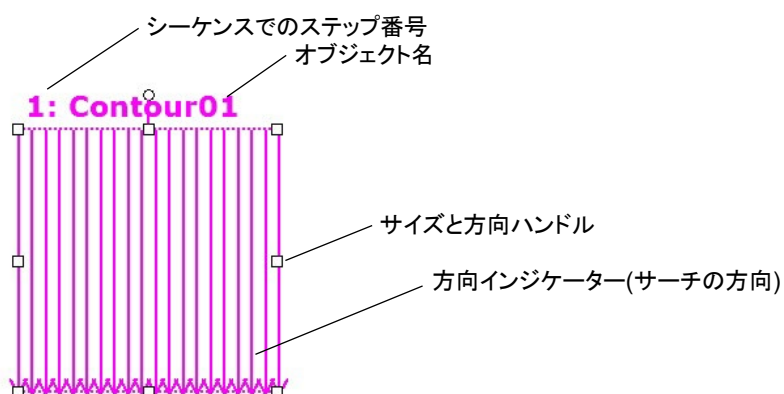
#### 1. Blob形式

配置したサーチウィンドウ内にある対象ワークのブローブを検出し、その輪郭を取得します。複雑な形状を持つワークから輪郭を取得する場合に使用します。



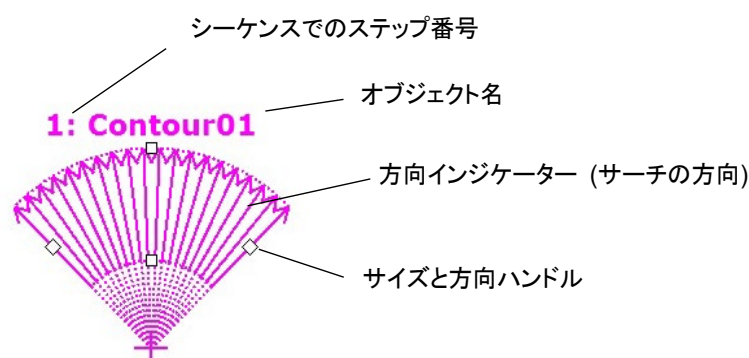
#### 2. Line形式

横に並ぶ複数のエッジ探索ラインで輪郭を取得します。対象ワークの一部から、起伏の少ない輪郭を簡単に取得する場合に便利です。



## 3. Arc形式

放射状に並ぶ複数のエッジ探索ラインで輪郭を取得します。対象ワークの一部から、起伏の少ない円弧上の輪郭を簡単に取得する場合に便利です。



## Contour オブジェクトのプロパティー

下の一覧は、Contourオブジェクトのプロパティーを簡単に説明した概略です。各リザルトの詳細は、「Vision Guide 7.0 プロパティー&リザルトリファレンス」を参照してください。

また、ContourModeの値により有効なプロパティーが異なります。一覧には、各プロパティーが有効になるContourModeの値を記載しています。

プロパティー	説明
AbortSeqOnFail	オブジェクトの実行に失敗した(不良判定になった)とき、直ちにシーケンス全体を中断し、それ以降のオブジェクトを処理しないように指定 デフォルト: False 対応形式: Blob, Line, Arc
Accept	特徴が検出されたと判定するための形状スコアの値を指定 対象物が検出されたとみなされるためには、スコアの値を超えなくてはなりません。 デフォルト: 100 対応形式: Line, Arc
AngleEnd	円弧をサーチする範囲の終了角度を指定 デフォルト: 135 対応形式: Arc
AngleStart	円弧をサーチする範囲の開始角度を指定 デフォルト: 45 対応形式: Arc
Caption	見出しをContourオブジェクトに割りあてる デフォルト: 空欄 対応形式: Blob, Line, Arc



プロパティ	説明
CenterPointObject	<p>オブジェクトの中心点として用いられる位置を指定 このプロパティを“Screen”に設定すると画面上の任意の位置にオブジェクトを配置できますが、他のビジョンオブジェクトを指定した場合、そのオブジェクトのPixelX, PixelYリザルトに中心点が設定されます。</p> <p>対応形式: Blob, Line, Arc</p>
CenterPntObjResult	<p>CenterPointObjectプロパティから使用するリザルトを指定 Allを指定すると、指定されたビジョンオブジェクトリザルトのすべて (NumberFound)にContourオブジェクトが適用されます。</p> <p>デフォルト: 1</p> <p>対応形式: Blob, Line, Arc</p>
CenterPntOffsetX	<p>サーチウィンドウの中心をCenterPointObjectプロパティで位置決めしたあとに、Xオフセットを設定、または返す</p> <p>対応形式: Blob, Line, Arc</p>
CenterPntOffsetY	<p>サーチウィンドウの中心をCenterPointObjectプロパティで位置決めしたあとに、Yオフセットを設定、または返す</p> <p>対応形式: Blob, Line, Arc</p>
CenterPntRotOffset	<p>中心点のXYオフセット値 (CenterPntOffsetX, CenterPntOffsetY)を、CenterPointObjectのAngleリザルトにしたがって回転するか否かを指定</p> <p>SearchWinTypeがRotatedRectangleに設定されている場合、サーチウィンドウはAngleリザルトにしたがって回転します。</p>
CenterX	<p>オブジェクトの中心点として用いられる位置のX座標を指定 CenterPointプロパティが他のビジョンオブジェクトに設定されると、自動的に設定されます。</p> <p>対応形式: Arc</p>
CenterY	<p>オブジェクトの中心点として用いられる位置のY座標を指定 CenterPointプロパティが他のビジョンオブジェクトに設定されると、自動的に設定されます。</p> <p>対応形式: Arc</p>
ContourMode	Contour オブジェクトのエッジ検出方式を定義
ContrastTarget	<p>エッジサーチのためのコントラストを設定 デフォルト: 0 (best contrast)</p> <p>対応形式: Line, Arc</p>
ContrastVariation	<p>ContrastTargetの許容されるコントラストの変化量を選択 デフォルト: 0</p> <p>対応形式: Line, Arc</p>

プロパティ	説明
CoordObject	リザルトをコピーするCoordinatesオブジェクトを指定 コピー処理はオブジェクトが実行される時に行われ、Decisionの分岐機能などで実行されなかった場合、コピー処理は行われません デフォルト: None
CurrentResult	オブジェクトウィンドウのリザルトリストに表示するリザルト または、1つのサーチウィンドウ内にある対象物などの複数のリザルトを、システムが検出する設定のときにデータを返すリザルトを定義  対応形式: Blob, Line, Arc
Description	ユーザー指定の説明を設定 デフォルト: 空欄
Direction	エッジ検索の方向を設定 デフォルト: InsideOut  対応形式: Arc
EdgeThreshold	しきい値(この値より低いエッジは無視する)を設定 デフォルト: 2  対応形式: Line, Arc
EdgeType	検索するエッジタイプ(シングル, ペア)を設定 デフォルト: 1 – single  対応形式: Line, Arc
Enabled	このオブジェクトを実行するかを指定 デフォルト: True  対応形式: Blob, Line, Arc
EndPntObjResult	どのリザルトをEndPointObjectから用いるのかを特定 デフォルト: 1  対応形式: Blob, Line, Arc
EndPointObject	どのビジョンオブジェクトを使って、直線の終点を定義するかを特定 デフォルト: Screen  対応形式: Blob, Line, Arc
EndPointType	どのタイプの終点を使って直線の終点を定義するかを特定 デフォルト: 0 – Point  対応形式: Blob, Line, Arc
FailColor	オブジェクトの検出結果が不良時の色を指定 デフォルト: Red  対応形式: Blob, Line, Arc

プロパティ	説明
FillHoles	2値画像の穴を埋めるかを選択 デフォルト: False 対応形式: Blob
FittingThreshold	直線、または円弧のフィッティング閾値を定義 対応形式: Line, Arc
Frame	どの位置に置いたフレームを使用するかを指定 デフォルト: none 対応形式: Blob, Line, Arc
FrameResult	Frameの何番目のリザルトを使用するかを指定 デフォルト: 1 対応形式: Blob, Line, Arc
Graphics	表示するグラフィックを指定 デフォルト: 1 – All 対応形式: Blob, Line, Arc
LabelBackColor	オブジェクトラベルの背景の色を設定 デフォルト: Transparent(透明) 対応形式: Blob, Line, Arc
LineDirection	輪郭点を出力する方向を定義 デフォルト: LeftToRight 対応形式: Line
MaxArea	オブジェクトの上限領域を規定 デフォルト: 100,000 対応形式: Blob
MinArea	オブジェクトの下限領域を規定 デフォルト: 25 対応形式: Blob
MinMaxArea	実行時のみ 1ステートメントでMinAreaとMaxAreaを設定、または返す
Name	Contourオブジェクトに独自のオブジェクト名を割りあてる デフォルト: Contour01 対応形式: Blob, Line, Arc
NumberOfEdges	検出するエッジ数を設定 デフォルト: 20 対応形式: Line, Arc
NumberToFind	出力する輪郭点の最大個数を定義 デフォルト: 1 対応形式: Blob, Line, Arc

プロパティ	説明
PassColor	オブジェクトの検出が、良のときの色を選択 デフォルト: LightGreen 対応形式: Blob, Line, Arc
PassType	オブジェクト検出時の良否判定の条件を定義 デフォルト: SomeFound 対応形式: Blob, Line, Arc
Polarity	Contourオブジェクトが、明るい背景の暗いオブジェクト (DarkOnLight)を検索するのか、暗い背景の明るいオブジェクト (LightOnDark)を検索するのか、明から暗へ (LightToDark)の移行を検索するのか、暗から明へ (DarkToLight)の移行を検索するのかを指定 デフォルト_Blob: 1 – DarkOnLight デフォルト_Line /Arc: 1 – LightToDark 対応形式: Blob, Line, Arc
RadiusInner	検出範囲の内径を指定 対応形式: Arc
RadiusOuter	検出範囲の外径を指定 対応形式: Arc
RejectOnEdge	“True”に設定すると、サーチウィンドウのエッジにかかって検出されたブローブを無視します デフォルト: False 対応形式: Blob
RotationDirection	輪郭点の回転方向を指定 デフォルト: 0 – CW 対応形式: Blob, Arc
RuntimeContour	オブジェクト実行時に輪郭点を検出するかを指定 デフォルト: True (実行時に検出) 対応形式: Blob, Line, Arc
SamplingPitch	輪郭点を削減する度合いを設定 デフォルト: 0 (削減しない) 対応形式: Blob, Line, Arc
SaveTeachImage	モデルをティーチするときに、画像ファイルに保存するかを設定
ScoreWeightContrast	コントラストによるスコアの割合を設定 デフォルト: 50 対応形式: Line, Arc
ScoreWeightStrength	エッジの強度によるスコアの割合を設定 デフォルト: 50 対応形式: Line, Arc

プロパティ	説明
SearchWidth	エッジサーチの幅を定義 範囲: 3 ~ 99 デフォルト: 3 対応形式: Line, Arc
SearchWin	実行時のみ 1コールでサーチウィンドウの左端, 上端, 高さ, 幅のパラメータを設定、または返す 対応形式: Blob, Line
SearchWinAngle	被検索領域の角度を定義 対応形式: Blob, Line
SearchWinCenterX	被検索領域の中心のX座標値を定義 対応形式: Blob, Line
SearchWinCenterY	被検索領域の中心のY座標値を定義 対応形式: Blob, Line
SearchWinHeight	被検索領域の高さを定義 (単位: ピクセル) デフォルト: 100 対応形式: Blob, Line
SearchWinLeft	被検索領域の最左端を定義 (単位: ピクセル) 対応形式: Blob, Line
SearchWinTop	被検索領域の最上端を定義 (単位: ピクセル) 対応形式: Blob, Line
SearchWinType	被検査領域のタイプ (Rectangle, RotatedRectangle, Circle)を定義 対応形式: Blob
SearchWinWidth	被検索領域の幅を定義 (単位: ピクセル) デフォルト: 100 対応形式: Blob
SizeToFind	検出する欠陥のサイズを選択 デフォルト: 1 – Largest 対応形式: Blob
ShowModel	輪郭の教示モデルを表示 対応形式: Blob
Sort	オブジェクトのリザルトを並び替える順番を選択 対応形式: Blob
StartPntObjResult	どのリザルトをStartPointObjectから用いるのかを指定 デフォルト: 1 対応形式: Blob, Line, Arc

プロパティ	説明
StartPointObject	どのビジョンオブジェクトを使って、直線の始点を定義するかを指定 デフォルト: Screen 対応形式: Blob, Line, Arc
StartPointType	どのタイプの始点を使って直線の始点を定義するのかを指定 デフォルト: “0 – Point” 対応形式: Blob, Line, Arc
StrengthTarget	サーチするエッジの強度を設定 デフォルト: 0 対応形式: Line, Arc
StrengthVariation	StrengthTargetの変化量の合計を設定 デフォルト: 0 対応形式: Line, Arc
ThresholdAuto	対象物(オブジェクト), 背景, 画像イメージのエッジを表すグレイレベルのしきい値を自動設定するか否かを設定 デフォルト: 無効 対応形式: Blob
ThresholdBlockSize	ThresholdMethodプロパティをLocalAdaptiveに設定したときに使用する、しきい値を決定するために近傍領域を参照する範囲を定義 デフォルト: 1/16ROI 対応形式: Blob
ThresholdColor	しきい値の範囲内にあるピクセルの色を定義 デフォルト: 黒色 対応形式: Blob
ThresholdHigh	ThresholdLowプロパティとともに、対象物(オブジェクト), 背景, 画像イメージのエッジを表すグレイレベル範囲を定義 ThresholdHighプロパティは、画像イメージの対象物領域のグレイレベル範囲の上限を定義 ThresholdLowとThresholdHighで定義されたグレイレベル範囲にあてはまる画像イメージは、どの部分も <u>ピクセルの重みづけ1</u> が割りあてられます。(つまり、対象物の一部分です。) ThresholdAutoが“True”で、かつThresholdColorが“White”の場合、本プロパティ値は“255”に設定され、変更できなくなります。 デフォルト: 128 対応形式: Blob

プロパティ	説明
ThresholdLevel	ThresholdMethodプロパティをLocalAdaptiveに設定したときに使用する、近傍領域との輝度差の割合を定義 デフォルト: 15% 対応形式: Blob
ThresholdLow	ThresholdHighプロパティとともに、対象物(オブジェクト), 背景, 画像イメージのエッジを表すグレイレベル範囲を定義 ThresholdLowプロパティは、画像イメージの対象物領域のグレイレベル範囲の下限を定義 ThresholdLowとThresholdHighで定義されたグレイレベル範囲にあてはまる画像イメージは、どの部分もピクセルの <u>重みづけ1</u> が割りあてられます。(つまり、対象物の一部分です。) ThresholdAutoが“True”で、かつThresholdColorが“Black”の場合、本プロパティ値は“0”に設定され、変更できなくなります。 デフォルト: 0 対応形式: Blob
ThresholdMethod	2値化の処理方法を設定
ContourTolerance	輪郭点を削減する時の許容値を設定します。 デフォルト: 0 対応形式: Blob, Line, Arc

### Contour オブジェクトのリザルト

下の一覧は、Contourオブジェクトリザルトを簡単に説明した概略です。各リザルトの詳細は、「Vision Guide 7.0プロパティ&リザルトリファレンス」を参照してください。

リザルト	説明
CameraX	カメラ座標系における輪郭点のX座標位置を返す
CameraY	カメラ座標系における輪郭点のY座標位置を返す
CameraXYU	実行時のみ カメラ座標系における輪郭点のCameraX座標, CameraY座標, CameraU座標を返す
Found	オブジェクトが検出されたかどうかを返す
NumberFound	輪郭点の総数を返す
Passed	検出結果が良か否かを返す
PixelX	輪郭点位置のX座標をピクセルで返す
PixelY	輪郭点位置のY座標をピクセルで返す
PixelXYU	実行時のみ 輪郭点位置のPixelX座標, PixelY座標, PixelU座標をピクセルで返す
RobotX	ロボット座標系における輪郭点のX座標位置を返す
RobotY	ロボット座標系における輪郭点のY座標位置を返す
RobotXYU	実行時のみ ロボット座標系における輪郭点のX座標位置, Y座標位置, U座標位置を返す
ShowAllResults	特定のビジョンオブジェクトのリザルトすべてを、表形式でダイアログに表示  リザルトの比較が簡単にできます。
Time	オブジェクト処理に要した時間を返す (単位: ミリ秒)

### 輪郭抽出の原理

輪郭抽出は、大別して2つの手順があります。各手順の内容は、下記のとおりです。

1. 初期輪郭点作成
  - ブローブ検出 (Blob)
  - 輪郭トレース (Blob)
  - エッジ検出 (Line/Arc)
2. 輪郭点編集
  - 輪郭点削減
  - 輪郭取得



### ブローブ検出 (Blob)

ContourModeの設定値が“Blob”の場合、Blobオブジェクトと同じ機能を用いて、サーチウィンドウ内のブローブを検出します。ブローブの検出原理は、下記を参照してください。

#### 6.2.4 Blob オブジェクト(2値検査)-2値検査の原理

Contourオブジェクトでは、サーチウィンドウ内にあるブローブを全て検出し、次のステップに進みます。

### 輪郭トレース (Blob)

ContourModeの設定値が“Blob”の場合、「ブローブ検出」ステップの後に輪郭トレースのステップが実行されます。「ブローブ検出」ステップで検出されたブローブの1つ目のリザルトのブローブに対して輪郭をトレースして、初期輪郭点を作成します。そのため輪郭を取得したいワークが1つ目のリザルトとして検出されるように、サーチウィンドウの位置調整や、Sort, MinArea, MaxAreaなどのプロパティ調整が必要です。

初期輪郭点は、隙間のない連続した軌跡として出力されます。また、初期輪郭点はStartPointObjectで指定する座標位置に最も近い点から始まり、EndPointObjectで指定する座標位置に最も近い点で終わります。

### エッジ検出 (Line/Arc)

ContourModeの設定値が“Line”、または“Arc”の場合、エッジ探索ラインの検出結果を初期輪郭点とします。初期輪郭点は、StartPointObjectで指定する座標位置に最も近い点から始まり、EndPointObjectで指定する座標位置に最も近い点で終わります。

### 輪郭点削減


初期輪郭点の作成後、不要な点の削減が行われます。輪郭点の削減はSamplingPitchとContourToleranceの値に従います。SamplingPitchの値からは、何個の初期輪郭点ごとに1つの輪郭点を取得するかが決まります。例えば、“10”に設定した場合は、最大で10個の初期輪郭点から1個の輪郭点を抽出します。また、ContourToleranceの値からは、初期輪郭点を削減するときの誤差の許容値が決まります。この値が大きいほど、対象ワークの形に従わない輪郭点の削減も許可されます。

### 輪郭取得

前のステップで削減された輪郭点をリザルトデータとして取得するステップです。取得するとき、RotationDirection / LineDirectionの設定値で軌跡の方向が決まります。

また、Blob形式の場合は、RejectOnEdgeを“False”にすると、対象ワークがサーチウィンドウからはみ出すことがあります。この場合は、サーチウィンドウの枠線は輪郭として取得しません。軌跡の途中でサーチウィンドウの枠に触れた部分で軌跡は終わりとなります。サーチウィンドウにより取得可能な軌跡が複数に分かれた場合は、StartPointObject, EndPointObjectが設定されていない限り、軌跡の中から最も長いものを取得します。

## Step 1: 新規Contourオブジェクトの作成

- (1) Vision Guideツールバー - <全ツール> -  <Contour (輪郭取得)>ボタンをクリックします。
- (2) 画像イメージ表示部にマウスを動かします。  
マウスポインターがContourアイコンに変わります。
- (3) アイコンが画像イメージ表示部の目的の位置に来るまで、マウスを動かし続けます。
- (4) マウスを左クリックして、オブジェクトを作成します。
- (5) オブジェクト名は、自動的に作成されます。  
この例では、このシーケンスで最初に作成されたContourオブジェクトなので、“Contour01”という名称になります。(名称の変更方法については後述します。)

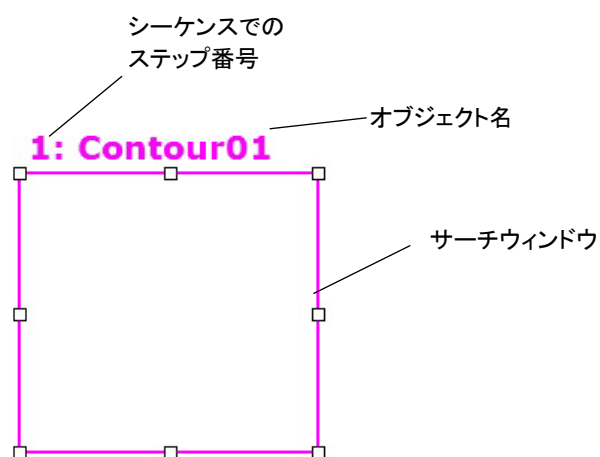
得たい軌跡の特徴に合わせてContourModeを変更します。ContourModeの選択については、下表を参照してください。

## ContourModeの設定値 判断基準

Blob	対象ワークを一周する軌跡を取得したい。 複雑な形状の軌跡を取得したい。
Line	ワークの一部分の軌跡を取得したい。 凹凸の少ない単純な線の軌跡を取得したい。
Arc	ワークの一部分の軌跡を取得したい。 凹凸の少ない円弧上の軌跡を取得したい。

## Step 2: 輪郭検出位置の設定 (ContourMode:Blobの場合)

ContourModeがBlobの場合、画面には下図のようなContourオブジェクトが表示されます。



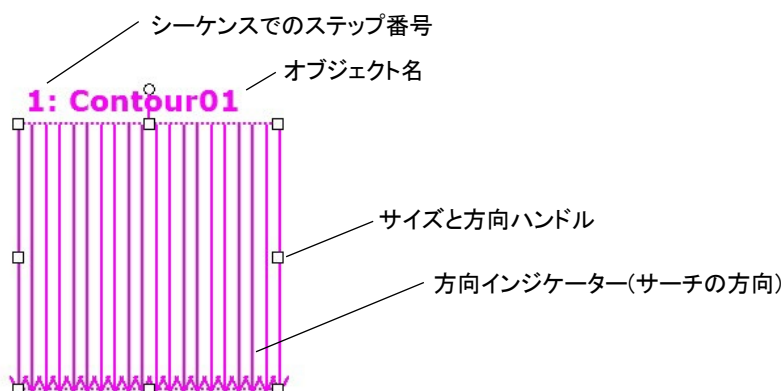
## 新規Contourオブジェクト(ContourMode:Blob)のレイアウト

- (1) Contourオブジェクトの名称ラベルをクリックし、マウスを押し続けながら、Contourオブジェクトをドラッグして、サーチウィンドウの左上コーナーを好きな位置に合わせます。

- (2) サイズハンドルをクリックし、マウスをドラッグすると、Contourオブジェクトサーチウィンドウのサイズが変更できます。適当な大きさに調整します。(サーチウィンドウは、プローブの被検索領域になります。)

### Step 2: 輪郭検出位置の設定 (ContourMode: Lineの場合)

ContourModeがLineの場合、画面には下図のようなContourオブジェクトが表示されます。

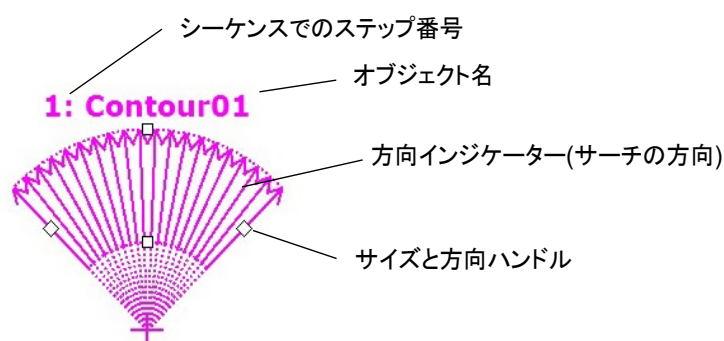


新規Contour オブジェクト(ContourMode:Line)のレイアウト

- (1) Contourオブジェクトの名称ラベルをクリックし、マウスを押し続けながら、Contourオブジェクトをドラッグして、サーチウィンドウの左上コーナーを好きな位置に合わせます。
- (2) サイズハンドルをクリックして、マウスをドラッグすると、Contourオブジェクトサーチウィンドウのサイズが変更できます。適当な大きさに調整します。  
画面には、Edgeオブジェクトが表示されています。

### Step 2: 輪郭検出位置の設定 (ContourMode:Arcの場合)

ContourModeがArcの場合、画面には下図のようなContourオブジェクトが表示されます。



新規Contourオブジェクト(ContourMode:Arc)のレイアウト

- (1) Contourオブジェクトの名称ラベルをクリックし、マウスを押し続けながら、Contourオブジェクトをドラッグして、サーチウィンドウを好きな位置に合わせます。
- (2) サイズハンドルをクリックして、マウスをドラッグすると、Contourオブジェクトサーチウィンドウのサイズが変更できます。適当な大きさに調整します。

## Step 3: Contourオブジェクトプロパティの設定 (ContourMode:Blobの場合)

Contourオブジェクト(ContourMode: Blobの場合)のプロパティの設定を行います。

ContourModeが“Blob”に設定されているときに、よく使用されるプロパティを示します。

その他、さまざまなビジョンオブジェクトの多くで使われるAbortSeqOnFail, Graphicsなどのプロパティの詳細は、「Vision Guide 7.0プロパティ&リザルトリファレンス」、または、「Contourオブジェクトのプロパティ」リストを参照してください。



注意

- ビジョンシーケンスの実行結果は、外らん光や外部機器ノイズにより影響を受けることがあります。  
外らん光や外部機器ノイズにより、影響を受けると、取り込み画像が想定外の結果となり、検出位置結果がサーチエリア (検出エリア)内の不特定位置になる可能性があります。  
MaxArea, MinArea, RejectOnEdgeなどのプロパティ値を適切に設定し、誤検出リスクを軽減してください。

- Nameプロパティ**      新規に作成されたContourオブジェクトに、デフォルトとして与えられる名称は、“Contour xx”です。xxは、1つのビジョンシーケンス内で複数のContourオブジェクトを用いるとき、それらのContourオブジェクトを区別するための番号です。  
ビジョンシーケンスで1番目に用いられるContourオブジェクトには、“Contour 01”という名称がデフォルトで設定されます。  
名称を変更したい場合は、Nameプロパティの設定値フィールドをクリックして、変更後の名称を入力し、リターンキーを押してください。画面上でそのContourオブジェクト名を表示する場所の全部が、変更後の名称で表示されます。
- Polarityプロパティ**      明るい背景の暗いオブジェクトを検出したい場合と、暗い背景の明るいオブジェクトを検出したい場合があります。この選択をするのが、Polarityプロパティの基本的な目的です。  
デフォルトは、DarkOnLight (明るい背景の暗いオブジェクト)の検出に設定されています。  
このデフォルト設定を変更したいときは、Polarityプロパティの値フィールドをクリックします。ドロップダウンリストが表示され、“Dark on Light”と“Light on Dark”の選択項目が表示されます。選択して、クリックしてください。
- MinArea, MaxArea**      輪郭抽出の対象とするブローブの面積を定義します。  
デフォルトの範囲は、25 ~ 100,000 (MinArea ~ MaxArea)です。  
非常に広く設定されています。つまり、MinAreaプロパティとMaxAreaプロパティを調整しないまま、新規作成のBlobオブジェクトを実行すると、ほとんどのブローブは検出されたとして報告されます。通常は、検出対象のブローブに適切な範囲を反映するために、これらのプロパティを調整します。このようにして、範囲の外側に何らかのブローブが検出された場合は、それは検出対象のブローブではないと判断することができます。
- RejectOnEdgeプロパティ**      サーチウィンドウのエッジにかかって検出されたパーツを除外  
通常は、“True”に設定します。

**RuntimeContour**      オブジェクト実行時に輪郭を抽出するかを定義  
ワークの形状が変化する場合は“True”に設定します。ワークの形状変化がない場合は、“False”に設定して、オブジェクト実行前に輪郭情報のティーチを行ってください。ティーチされた形状はShowModelプロパティから確認できます。

以上で、Blobオブジェクトのテストができるようになりました。必要となるその他のプロパティは、テスト後に再び設定することになります。

### Step 3: Contourオブジェクトプロパティの設定 (ContourMode: Lineの場合)

Contourオブジェクト(ContourMode: Lineの場合)のプロパティを設定します。

ContourModeが“Line”に設定されているときに、よく使用されるプロパティを示します。その他、さまざまなビジョンオブジェクトの多くで使われるAbortSeqOnFail, Graphicsなどのプロパティの詳細は、Vision Guide 7.0プロパティ&リザルトリファレンスを参照してください。

**EdgeType(シングル)**      検索するエッジの種類を選択  
エッジペアを設定した場合、それぞれの方向のエッジを検索し、中間点を位置として検出します。

**Nameプロパティ ("Contourxx")**      新規に作成されたContourオブジェクトに、デフォルトとして与えられるオブジェクト名称は“Contour xx”です。xxは、1つのビジョンシーケンス内で複数のContourオブジェクトを用いるとき、それらのContourオブジェクトで区別するための番号です。  
ビジョンシーケンスで1番目に用いられるContourオブジェクトには、“Contour01”というオブジェクト名がデフォルトで設定されます。  
オブジェクト名を変更したい場合は、Nameプロパティの設定値フィールドをクリックして、変更後の名称を入力し、リターンキーを押してください。画面上でContourオブジェクト名を表示する場所は、すべて変更後の名称に変わります。

**NumberOfEdges(1)**      エッジ検索ラインに沿って1個以上のエッジを検索

**Polarity (LightToDark)**      エッジ位置を「明から暗へ(LightToDark)」検索  
「暗から明へ(DarkToLight)」のエッジを検索する場合は、Polarityを変更してください。

### Step 3: Contourオブジェクトプロパティの設定 (ContourMode: Arcの場合)

Contourオブジェクト(ContourMode: Arcの場合)のプロパティを設定します。  
ContourModeが“Arc”に設定されているときに、よく使用されるプロパティを示します。  
その他、さまざまなビジョンオブジェクトの多くで使われるAbortSeqOnFail, Graphicsなどのプロパティの詳細は、「Vision Guide 7.0プロパティ&リザルトリファレンス」を参照してください。

EdgeType(シングル)	検索するエッジの種類を選択  エッジペアを設定した場合、それぞれの方向のエッジを検索し、中間点を位置として検出します。
Nameプロパティ ("Contourxx")	新規に作成されたContourオブジェクトに、デフォルトとして与えられるオブジェクト名称は“Contourxx”です。xxは、1つのビジョンシーケンス内で複数のContourオブジェクトを用いるとき、それらのContourオブジェクトで区別するための番号です。 ビジョンシーケンスで1番目に用いられるContourオブジェクトには、“Contour01”というオブジェクト名がデフォルトで設定されます。 オブジェクト名を変更したい場合は、Nameプロパティの設定値フィールドをクリックして、変更後の名称を入力し、リターンキーを押してください。画面上でContourオブジェクト名を表示する場所は、すべて変更後の名称に変わります。
NumberOfEdges(5)	円弧エッジを検索するために、5個のエッジを検索
Polarity (LightToDark)	エッジ位置を「明から暗へ(LightToDark)」検索  「暗から明へ(DarkToLight)」のエッジを検索する場合は、Polarityを変更してください。

### Step 4: 輪郭点に関わるプロパティの設定

軌跡の正確さ、輪郭点の個数などを調整するプロパティを設定します。  
使用されるプロパティを示します。その他、プロパティの詳細は、「Vision Guide 7.0プロパティ&リザルトリファレンス」を参照してください。

SamplingPitch	輪郭点を削減する最大個数を定義  “5”を設定した場合、個数が初期検出個数から最大で1/5程度になります。
ContourTolerance	輪郭点を削減するときの許容値を定義  この値が小さいほどワーク形状に反する輪郭点の削減を許可しません。オブジェクト実行結果を確認しながら、必要な精度に合わせて調整してください。
StartPointObject, EndPointObject	軌跡の開始位置と終了位置を指定  設定する場合は、Pointオブジェクトを作成し、Contourオブジェクトより前に実行されるよう、ビジョンシーケンス内の順序を調整します。Pointオブジェクトは、軌跡を開始/終了したい位置付近に配置します。

RotationDirection, LineDirection	軌跡の方向を指定 ContourModeがBlobとArcの場合は軌跡の回転方向(CW/ CCW)を設定します。ContourModeがLineの場合は、エッジ探 索ラインが垂直下方向を向いている場合を基準として、左右方 向(LeftToRight/ RightToLeft)を指定します。
-------------------------------------	---

### Step 5: Contourオブジェクトのテストとリザルトの確認

Contourオブジェクトのテストを実行するには、実行パネルのオブジェクトの<Run>ボタンをクリックします。Contourオブジェクトのリザルトが表示されます。ここで確認する主要なリザルトは、次に示すとおりです。

PixelX, PixelY	検出された輪郭点位置 (単位: ピクセル)
CameraX, CameraY	カメラ座標を基準とした輪郭点位置 (単位: mm)
RobotX, RobotY	ロボット座標を基準とした輪郭点位置 (単位: mm)
Timeリザルト	Contourオブジェクトの実行に要した時間

#### NOTE



RobotX, RobotYのリザルトとCameraX, CameraYのリザルトは、この時点で“no cal”を返します。これは、「キャリブレーションが実行されていないので、ロボット座標系あるいはカメラ座標系に関して座標リザルトを計算することができなかった。」ということを示しています。詳細は、「7. ビジョンキャリブレーション」を参照してください。

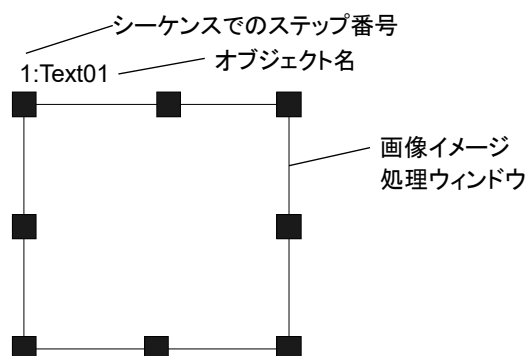
## 6.2.21 Textオブジェクト

### Text オブジェクトの概要

Textオブジェクトは、ビジョンオブジェクトの実行結果を、文字として画面上に描画します。

### Textオブジェクトのレイアウト

Textオブジェクトには、下に示すように画像イメージ処理ウィンドウがあります。



### Textオブジェクトのプロパティ

Textオブジェクトで使われるプロパティの一覧を次に示し、簡単に説明します。各プロパティの詳細は、Vision Guide 7.0 プロパティ&リザルトリファレンスを参照してください。

プロパティ	説明
AbortSeqOnFail	オブジェクトの実行に失敗した(不良判定になった)とき、直ちにシーケンス全体を中断し、それ以降のオブジェクトを処理しないように指定 デフォルト: False
CenterPointObject	オブジェクトの中心点として用いられる位置を指定 このプロパティを“Screen”に設定すると画面上の任意の位置にオブジェクトを配置できますが、他のビジョンオブジェクトを指定した場合、そのオブジェクトのPixelX, PixelYリザルトに中心点が設定されます。
CenterPntObjResultCenterPointObject	CenterPointObjectプロパティからどのリザルトを使うのかを指定 Allを指定すると、指定されたビジョンオブジェクトリザルトのすべて(NumberFound)にTextオブジェクトが適用されます。 デフォルト: 1
CenterPntOffsetX	サーチウィンドウの中心を CenterPointObject プロパティで位置決めしたあとに、X オフセットを設定、または返す
CenterPntOffsetY	サーチウィンドウの中心を CenterPointObject プロパティで位置決めしたあとに、Y オフセットを設定、または返す
Caption	見出しをTextオブジェクトに割りあてる デフォルト: 空欄
CurrentResult	オブジェクトウィンドウのリザルトリストに表示するリザルト、または、1つのサーチウィンドウ内にある対象物などの複数のリザルトを、システムが検出する設定のときにデータを返すリザルトを定義
Description	ユーザー指定の説明を設定 デフォルト: 空欄
Enabled	このオブジェクトを実行するかを指定 デフォルト: True
FailColor	オブジェクトの検出不良時の色を選択
Font	描画する文字の書式を設定します。
FontBold	フォントを太字にします。(SPEL+プログラムからのみ有効)
FontItalic	フォントをイタリックにします。(SPEL+プログラムからのみ有効)
FontName	フォントの名前を設定します。(SPEL+プログラムからのみ有効)
FontSize	フォントのサイズをポイントで設定します。 (SPEL+プログラムからのみ有効)
Graphics	表示するグラフィックを指定 デフォルト: 1 - All
LabelBackColor	オブジェクトラベルの背景の色を設定
TextBackColor	文字の背景の色を設定 デフォルト: Transparent (透明)
Name	Textオブジェクトに独自のオブジェクト名を割りあてる デフォルト: Text01



プロパティ	説明
PassColor	オブジェクトの検出良時の色を選択 デフォルト: LightGreen
ResultObject	描画するビジョンオブジェクトを指定
ResultText1	描画するリザルトを指定
ResultText2	描画するリザルトを指定
ResultText3	描画するリザルトを指定
ShowLabel	文字列にラベルを付与するかを指定
SearchWin	実行時のみ 1コールでサーチウィンドウの左端, 上端, 高さ, 幅のパラメーターを設定、または返す
SearchWinHeight	被検索領域の高さを定義 (単位: ピクセル)
SearchWinLeft	被検索領域の最左端を定義 (単位: ピクセル)
SearchWinTop	被検索領域の最上端を定義 (単位: ピクセル)
SearchWinWidth	被検索領域の幅を定義 (単位: ピクセル)
UserText	ユーザー任意文字列を設定します

### Textオブジェクトリザルト

下の一覧は、Textオブジェクトリザルトを簡単に説明した概略です。各リザルトの詳細は、「Vision Guide 7.0プロパティ&リザルトリファレンス」を参照してください。

リザルト	説明
Passed	検出結果が良か否かを返す
Found	リザルトが取得できたかどうかを返す
PixelX	文字列描画位置のX座標をピクセルで返す
PixelY	文字列描画位置のY座標をピクセルで返す


### Textオブジェクトの使用方法

Vision Guide Textオブジェクトの使い方を理解するための基礎を説明してきました。

Textオブジェクトを使用するのに必要な手順は、次のとおりです。

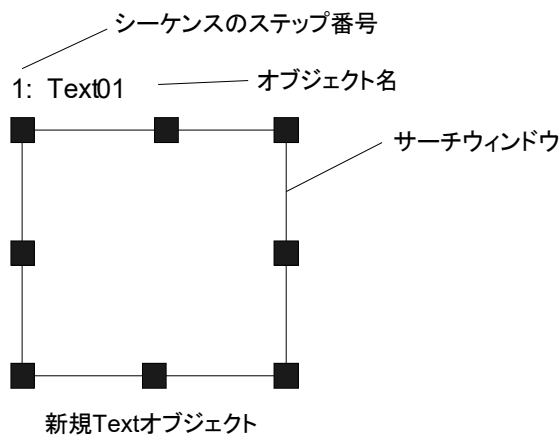
- 新規Textオブジェクトの作成方法
- サーチウィンドウの位置とサイズの設定
- 文字列を設定
- Textオブジェクトの実行と画面の確認

### Step 1: 新規Textオブジェクトの作成

- (1) Vision Guideツールバー - <全ツール> -  <Text>ボタンをクリックします。
- (2) Textオブジェクトボタンの上に<Text>アイコンが表示されます。
- (3) <Text>アイコンをクリックして、Vision Guideウィンドウのイメージディスプレイへドラッグします。  
オブジェクトの名前は、自動的に作成されます。例えば、シーケンスに作成された最初のTextオブジェクトの場合は、“Text01”となります。(名前の変更方法については、後で説明します。)

### Step 2: サーチウィンドウの位置とサイズの設定

画面には、下図のような Text オブジェクトが表示されます。



- (1) Textオブジェクトの名称ラベルをクリックし、マウスを押し続けながら、Textオブジェクトをドラッグして、文字列を描画したい位置へサーチウィンドウの左上コーナーを合わせます。
- (2) サイズハンドルをクリックして、マウスをドラッグすると、Textオブジェクトサーチウィンドウのサイズが変更できます。適当な大きさに調整します。  
(サーチウィンドウの左上を基準に文字列が描画されます)

### Step 3: 文字列を設定

- (1) 描画したいリザルトを持つビジョンオブジェクトをResultObjectで指定します。  
ResultObjectで選択できるビジョンオブジェクトは、ビジョンシーケンス内でTextオブジェクトよりも前に実行される必要があります。
- (2) ResultText1 ~ 3から、描画したいリザルト名を選択します。
- (3) リザルト以外の文字列を描画する場合は、UserTextプロパティを編集します。

(4) Fontプロパティを編集し、フォントを調整します。

Fontプロパティの入力欄をクリックすると、次の設定ウィンドウが表示されます。フォントの書体, スタイル, 大きさが設定できます。



#### Step 4: Textオブジェクトの実行と画面の確認

実行パネルのオブジェクトの<Run>ボタンをクリックします。

Textオブジェクトに設定された文字列が画面に描画されます。描画される文字に問題がある場合は、プロパティの調整を行ってください。

#### Textオブジェクトで描画される文字列

Textオブジェクトで描画される文字列のフォーマットは、下記の通りです。



文字列は、上からUserText, ResultText1, ResultText2, ResultText3の順番で描画されます。UserTextが空欄の場合や、ResultText1~3の値が、“None”の場合は、その部分は描画されません。文字列がサーチウィンドウに入りきらない場合、その部分は描画されません。

## 6.2.22 Decisionオブジェクト

### Decisionオブジェクトの概要

Decisionオブジェクトは、指定したビジョンオブジェクトの成功または失敗に基づいてシーケンス実行のフローを制御するために使用されます。

### Decisionオブジェクトのプロパティ

Decisionオブジェクトで使われるプロパティの一覧を次に示し、簡単に説明します。各プロパティの詳細は、「Vision Guide 7.0 プロパティ&リザルトリファレンス」を参照してください。

プロパティ	説明
ConditionObject	同一シーケンス内のDecisionオブジェクトより前にあるビジョンオブジェクトを指定します。 その結果を用いて、DecisionオブジェクトのTrue分岐またはFalse分岐のどちらかのシーケンスフローを使用するかを決定します。
Description	ユーザー指定の説明を設定 デフォルト: 空欄
Enabled	このオブジェクトを実行するかを指定 デフォルト: True
Name	Decisionオブジェクトに独自のオブジェクト名を割り当て デフォルト: Decision01
TrueCond	ConditionObject で指定したビジョンオブジェクトのPassedリザルトを用いて、True分岐に進む条件を指定します。 TargetPassed: PassedリザルトがTrueの場合、True分岐に進みます。 TargetFailed: PassedリザルトがFalseの場合、False分岐に進みます。 TargetNoExec: ビジョンオブジェクトが実行されなかった場合に、True分岐に進みます。 デフォルト: 0-TargetPassed

### Decisionオブジェクトリザルト

Decisionオブジェクトにリザルトはありません。

### Decisionオブジェクトの使用方法

Vision Guide Decisionオブジェクトの使い方を理解するための基礎を説明してきました。

Decisionオブジェクトを使用するのに必要な手順は、次のとおりです。


- 新規Textオブジェクトの作成方法
- 条件分岐の設定
- 実行オブジェクトの設定と実行
- Coordinates オブジェクトの使い方

## Step 1: 新規Decisionオブジェクトの作成

- (1) DecisionオブジェクトのConditionObjectプロパティに設定するオブジェクトを事前に追加します。この例では、Geometricオブジェクトを追加します。

Decisionオブジェクトは、Decisionオブジェクトの前に実行する必要がある

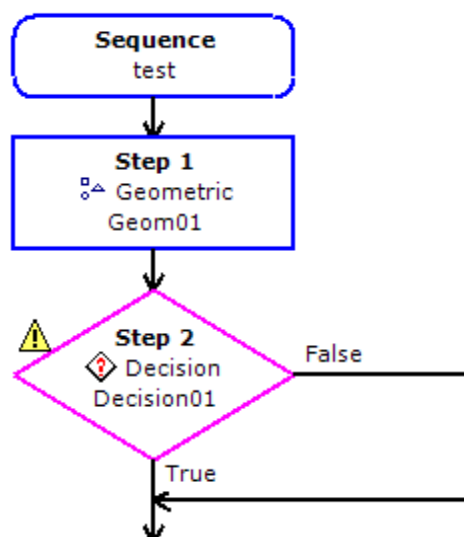
ConditionObjectを設定しないと使用できないため、シーケンスの先頭には追加することができません。

- (2) Vision Guideツールバー - <全ツール> -  <Decision>ボタンをクリックします。

- (3) <Decision>アイコンをクリックして、Vision Guideウィンドウの画像表示部へドラッグします。

オブジェクトの名前は、自動的に作成されます。例えば、シーケンスに作成された最初のDecisionオブジェクトの場合は、“Decision01”となります。(オブジェクト名の変更方法については後述します。)

Decisionオブジェクトはサーチウィンドウを持たないため、フローチャートやシーケンスツリーからステップ位置を確認してください。



## Step 2: 条件分岐の設定

- (1) DecisionオブジェクトのConditionObjectプロパティのドロップダウンリストから結果を確認するオブジェクトを選択します。ConditionObjectは、シーケンス内でDecisionオブジェクトよりも前に実行されるオブジェクトのみ設定できます。この例では、ConditionObjectをGeom01に設定します。

- (2) DecisionオブジェクトのTrueCondプロパティを設定して、(1)で選択したオブジェクトのPassedリザルトの値を用いて、True分岐に進む条件を指定します。

デフォルトの0-TargetPassedを指定されている場合は、(1)のオブジェクトのPassedリザルトがTrueの時に、True分岐に進んでビジョンオブジェクトが実行されます。

PassedリザルトがFalseの時に、True分岐に進んでビジョンオブジェクトを実行したければ、TrueCondプロパティをTargetFailedに指定します。

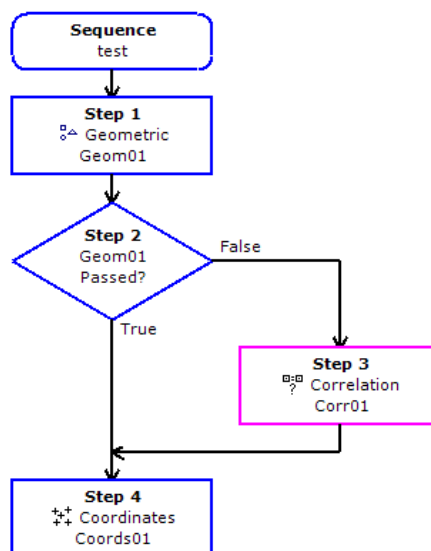
### Step 3: オブジェクト追加と実行

- (1) DecisionオブジェクトのTrue及びFalse 分岐で実行するオブジェクトを追加します。  
Vision Guideツールバーから追加したいオブジェクトを選択した後、それをフローチャートにドラッグして、Decisionオブジェクトの分岐内に配置します。各分岐には、必要な数のオブジェクトを配置できます。  
分岐内には、Decisionオブジェクトを配置することはできません
- (2) Decisionオブジェクトの分岐合流後に、オブジェクトを追加します。  
必要に応じて、Decisionオブジェクトが分岐合流後、シーケンスにオブジェクトを追加できます。
- (3) ビジョンシーケンスの動作を確認する。  
DecisionオブジェクトのConditionObjectで指定されたオブジェクトのリザルトと、TrueCondプロパティの設定値の組み合わせにより、実行する分岐が変化します。  
シーケンス全体を実行することも、シーケンスをステップ実行して動作を確認することもできます。

### Step 4: Coordinates オブジェクトの使い方

場合によっては、DecisionオブジェクトのTrueまたはFalse分岐でのオブジェクト処理に基づいて、 픽셀、カメラ、またはロボットの座標の取得が必要になります。そのような場合、座標結果を提供するオブジェクトの結果をCoordinatesオブジェクトに保存してから、プログラムでCoordinatesオブジェクトの結果から座標にアクセスできます。

- (1) Coordinatesオブジェクトをシーケンスに追加します。
- (2) 座標を保存するオブジェクトのCoordObjectプロパティを設定します。任意の数のオブジェクトが同じCoordinatesオブジェクトに座標を格納できます。CoordObjectを使用する各オブジェクトが実行されると、座標が上書きされます。この例では、Geom01とCorr01の両方でCoordObjectがCoords01に設定されています。



- (3) プログラムで、VGetを使用してCoordinatesオブジェクトから座標を取得します。この例では、Geom01がパスした場合、Geom01座標結果がCoords01にコピーされます。Geom01がパスしない場合、Corr01が実行されます。Corr01がパスした場合、Corr01座標結果がCoords01にコピーされます。

SPELプログラムでは、Coords01からロボットの座標を取得することができます。

VGet test.Cords01.RobotXYU, found, x, y, u

### 6.2.23 Coordinatesオブジェクト

#### Coordinatesオブジェクトの概要

Coordinatesオブジェクトは、他のオブジェクトのリザルトを格納するために使用します。主に、Decisionオブジェクトを含むシーケンスで使用されます。

Decisionオブジェクトを持つシーケンスでは、分岐機能の実行に応じて、目的のリザルトが得られるオブジェクトが変化します。各分岐のオブジェクトに、座標結果を格納するためのCoordinatesオブジェクトを設定することで、Coordinatesオブジェクトから目的の結果を取得することができます。

Coordinatesオブジェクトは、リザルトを格納したいオブジェクトから指定します。下記のプロパティからCoordinatesオブジェクトを選択します。

プロパティ	説明
CoordObject	リザルトを格納するCoordinatesオブジェクトを指定 格納処理はオブジェクトが実行される時に行われ、Decisionの分岐機能などで実行されなかった場合、コピー処理は行われません デフォルト:None

CoordObjectプロパティを指定できるオブジェクトはPixel、Camera、及び、Robot X、Y、Uを持つオブジェクトのみです。CoordObjectプロパティに指定するCoordinatesオブジェクトは、設定するオブジェクトのステップの前後を問わずこのステップでも指定可能です。

Coordinatesオブジェクトは複数オブジェクトのCoordObjectプロパティから指定できます。この場合、各オブジェクトが実行されるたびに格納プロセスが実行されて、格納されたリザルトが上書きされます。

#### Coordinatesオブジェクトのプロパティ

下の一覧は、Coordinatesオブジェクトのプロパティを簡単に説明した概略です。各リザルトの詳細は、「Vision Guide 7.0 プロパティ&リザルトリファレンス」を参照してください。

プロパティ	説明
CurrentResult	オブジェクトウィンドウのリザルトリストに表示するリザルト、または、1つのサーチウィンドウ内にある対象物などの複数のリザルトを、システムが検出する設定のときにデータを返すリザルトを定義
Description	ユーザー指定の説明を設定 デフォルト: 空欄
Enabled	このオブジェクトを実行するかを指定 デフォルト: True
Name	Coordinatesオブジェクトに独自のオブジェクト名を割りあてる デフォルト: Coords01



### Coordinatesオブジェクトのリザルト

下の一覧は、Coordinatesオブジェクトリザルトを簡単に説明した概略です。各リザルトの詳細は、「Vision Guide 7.0 プロパティ&リザルトリファレンス」を参照してください。

リザルト	説明
Angle	検出されたポイントの角度を返す
CameraX	カメラ座標系における、検出オブジェクトのX座標を返す
CameraY	カメラ座標系における、検出オブジェクトのY座標を返す
CameraXYU	実行時のみ カメラ座標系における検出オブジェクト位置のCameraX座標, CameraY座標, CameraU座標を返す
NumberFound	検出されたポイントの数を返す
Found	オブジェクトが検出されたかどうかを返す
Passed	検出結果が良か否かを返す
PixelX	ピクセル座標系における、検出オブジェクトのX座標を返す
PixelY	ピクセル座標系における、検出オブジェクトのY座標を返す
PixelXYU	実行時のみ 検出オブジェクト位置のPixelX座標, PixelY座標, PixelU座標をピクセルで返す
RobotX	ロボット座標系における、検出オブジェクトのX座標を返す
RobotY	ロボット座標系における、検出オブジェクトのY座標を返す
RobotU	ロボット座標系における、検出オブジェクトの回転量を返す
RobotXYU	実行時のみ ロボット座標系における、検出オブジェクト位置のX座標, Y座標, 回転量を返す
ShowAllResults	特定のビジョンオブジェクトのリザルトすべてを、表形式でダイアログに表示 リザルトの比較が簡単にできます。


### Coordinatesオブジェクトの使用方法

Coordinatesオブジェクトの作成方法と使用方法を、下記の項目について説明します。

- 新規Coordinatesオブジェクトの作成方法
- CoordObjectの設定
- シーケンスの実行

次に説明するステップを開始する前に、新規ビジョンシーケンスを作成するか、使用するビジョンシーケンスを選択してください。

### Step 1: 新規Coordinatesオブジェクトの作成

- (1) Vision Guideツールバー - <全ツール> -  <Coordinates>ボタンをクリックしてください。
- (2) < Coordinates >オブジェクトツールバーボタンの上に、Coordinatesアイコンが表示されます。
- (3) Coordinatesアイコンをクリックして、Vision Guideウィンドウの画像イメージ表示部にドラッグしてください。
- (4) オブジェクト名は自動的に作成されます。この例では、このシーケンスで最初に作成されたCoordinatesオブジェクトなので、“Coords 01”というオブジェクト名になります。(オブジェクト名の変更方法については後述します。)

### Step 2: CoordObjectの設定

- (1) Coordinatesオブジェクトにリザルトを格納したいオブジェクトを追加します。  
前述でCoordinatesオブジェクトを追加しましたが、リザルトをコピーしたいオブジェクトとCoordinatesオブジェクトのステップ位置の前後関係に制約は無いため、状況に応じてステップ位置を入れ替えてください。
- (2) オブジェクトを追加後、CoordObjectプロパティからCoordinatesオブジェクトを選択し、指定します。

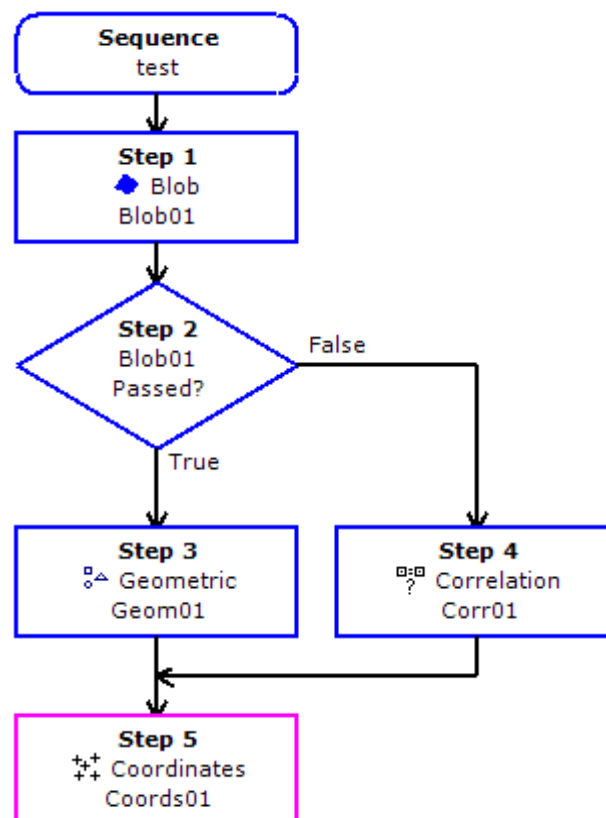
### Step 3: シーケンスの実行

シーケンスを実行します。

シーケンス実行後、Coordinatesオブジェクトの結果を確認してください。CoordinatesオブジェクトをCoordObjectに設定したオブジェクトの結果が格納されています。

以下に示すフローチャートの例では、Geom01とCorr01の両方のCoordsObjectプロパティがCoords01に設定されています。Blob01がパスした場合、Geom01が実行されて、結果がCoords01に格納されます。Blob01がパスしない場合、Corr01が実行されて、結果がCoords01に格納されます。SPELプログラムでは、Coords01からロボットの座標を取得することができます。

VGet test.Coords01.RobotXYU, found, x, y, u



### 6.2.24 1つのオブジェクトから複数のリザルトを返す

Blob, Geometric, Edge, Correlation, DefectFinderオブジェクトは、単一のサーチウィンドウ内で1つ以上の対象物を検索することができます。

ビジョンオブジェクトの複数のリザルトで作業する方法を理解いただくために、次の事項について説明します。

- CurrentResult プロパティ
- NumberToFind プロパティ
- NumberFound リザルト
- ShowAllResults リザルト

#### ビジョンオブジェクトのデフォルトの対象物数と設定可能な最大対象物数

Blob, Geometric, Edge, Correlation, DefectFinderオブジェクトは、デフォルトプロパティの設定では、サーチウィンドウ内に1つの対象物しか検索することができません。これは、NumberToFindプロパティのデフォルトが“1”に設定されているからです。

NumberToFindプロパティを“1”より大きい数に設定すると、ビジョンオブジェクトは、設定された数と同じ数の対象物を検出しようとします。

NumberToFindプロパティを“All”に設定するとオブジェクトの最大検出数(100個)までの検出を試みます。

NumberToFindプロパティを設定し、ビジョンオブジェクトを実行してください。現在の画像イメージ中の容認基準を満たす複数の対象物が検出されて表示されます。(foundオブジェクトに返された位置を示す十字線がある緑色のボックス) また、検出されたブローブの中で、CurrentResultとみなされたブローブが、その他のブローブに比べてより明るい緑色で強調表示されます。

#### 検出された対象物のソート順番

複数検出のオブジェクトの初回実行時には、CurrentResultプロパティは自動的に“1”に設定され、ビジョンオブジェクトの複数リザルトについて1番目のリザルトが、オブジェクトウィンドウのリザルトリストに表示されることになります。

Correlation、Geometricオブジェクトの場合、1番目のリザルトは最高スコアを持つリザルトです。(このCorrelationオブジェクトの検出されたすべての対象物を比べて) 2番目のリザルトは2番目に高いスコアを持つリザルトになります。

Blob、DefectFinderオブジェクトの場合、1番目のリザルトはSizeToFindプロパティとSortプロパティの値に基づいて返されたリザルトになります。(例えば、SizeToFindを“Largest”に設定すると、1番目のリザルトは検出された最大のオブジェクトになります。) リザルトのソートの詳細は、Vision Guide 7.0プロパティ&リザルトリファレンス「SizeToFindプロパティとSortプロパティ」を参照してください。

### ビジョンオブジェクトの複数リザルトを確認する方法

リザルトリストのヘッダーを注意深く見てみると、“1/10”と表示されている場所があるのに気がつきます。(この場合、例として、BlobオブジェクトのNumberToFindプロパティを“10”に設定していると想定してください。)

この表示は、CurrentResultが10個の検索対象物(NumberToFindプロパティで定義されている)のうちの1番目であることを示しています。

リザルトリストのヘッダーに表示されている“1/10”の中の2番目の数値(ここでは“10”という数値)は、NumberToFindプロパティに設定されている値であり、実際に検出された対象物の個数ではない、ということに注意してください。

10個の対象物を検出したい場合に、検出個数が5個とされる場合があります。この場合、検出された対象物の上位10個までがリザルトとして計算され表示されるべきなのに、なぜ5個の対象物だけしか表示されないのかと、疑問に思われるでしょう。ここで説明する方法によって、どうして他の5つの対象物が検出されなかったかを調べることができます。

見たいブローブを表示するようにCurrentResultプロパティの値を変更することによって、検出したい複数ブローブのうちのどのブローブについても、そのリザルトを確認することができます。

このようなリザルトを確認するには、CurrentResultプロパティの設定値フィールドに直接、番号を入力してください。または、CurrentResultプロパティの設定値フィールドにカーソルを動かして、[Shift]キー + [↓]キー、または[Shift]キー + [↑]キーを使ってリザルト内を検索してください。

CurrentResultプロパティの値、またはリザルトリストのヘッダーを参照することによって、どの順番のリザルトがリザルトリストに表示されているかを知ることができます。NumberToFindを10に設定すると、1番目のリザルトは“1/10”と表示されます。2番目のリザルトは“2/10”、3番目のリザルトは“3/10”などとなります。



注意

- CurrentResultプロパティの値を変更すると、プログラムでVGetした結果も変更されます。例えば、SPEL+プログラム中で、ビジョンオブジェクトからの値をVGetで取得するとき、CurrentResultプロパティには3の値が設定されているとすると、VGetの命令は、RobotXYU値の3番目を返すこととなります。このように、CurrentResultの設定変更は、SPEL+プログラムの結果にも影響しますので、注意してください。

ヒント: 複数リザルトを取得する場合には、VGet命令に指定するリザルト名で明示的にリザルト番号を指定してください。

**Example:** VGet seqname.objname.RobotXYU(1), found, X, Y, U

### NumberFoundリザルトの使用方法

NumberFoundプロパティは、実際に検出されたブローブの個数を表示することができます。非常に便利です。このリザルトは、SPEL+言語からも利用できるため、SPEL+プログラムを使って、適切な個数のリザルトが検出されたことを確認し、あるいは検出された個数を計算し、あるいは他のものがどのくらいあるかを計算することができます。

検出されたリザルトの数が5未満であるかを調べるためのコードの一部を示します。

```
VGet seqname.objname.NumberFound, numfound
If numfound < 5 Then
    'put code to handle this case here
```

ビジョンが1つのVRunでできるだけ多くのパーツを検出するのに使われるという状況を考えてください。ロボットはすべての検出したパーツがコンベアに乗せられるまで、一つ一つパーツを取ります。この例は、ロボットに検出した各パーツを取り上げて、それから作動中のコンベア上の同じ位置に1つずつ落とすようにさせるコードの一部分を示しています。

```
VRun seqname
VGet seqname.objname.NumberFound, numfound
For i = 1 to numfound
    VGet seqname.objname.RobotXYU(i), found, X, Y, U
    If found = True Then
        'Set coordinates found from vision
        VPick = XY(X, Y, -100.000, U)
    Else
        Print "Vision Error: part not found"
    EndIf

    Jump Vpick           'Jump to Vision Pickup position
    On Gripper           'Turn on vacuum
    Wait .1
    Jump Vconvey         'Jump to Conveyor Position to drop part
    Off Gripper          'Turn off vacuum
    Wait .1
Next count
```

### 複数リザルトを一度にすべて確認する方法

BlobオブジェクトとCorrelationオブジェクトの複数リザルト機能に関する、非常に便利な機能のひとつに、ShowAllResultリザルトがあります。

例えば、それぞれのリザルトの最高スコアと2番目のスコアを比較して見たい場合、また、3番目の対象物が検出された後で、大きなスコアの落差が見つかる場合などで、ShowAllResultsリザルトを使うと、すべてのリザルトを一度に全部確認することができます。

ShowAllResultsリザルト値のフィールドをクリックすると、1個のボタンが表示されます。そのボタンをクリックするとダイアログが表示されて、現在のビジョンオブジェクトのリザルト全部を表示します。

### ShowAllResultsリザルトの例

Result	Found	Area	PixelX	PixelY	Angle	RobotX	RobotY	RobotU	CameraX	CameraY
1	True	2507.0	542.771	394.182	70.146	(no cal)	(no cal)	(no cal)	(no cal)	(no cal)
2	True	2479.0	513.804	253.456	85.176	(no cal)	(no cal)	(no cal)	(no cal)	(no cal)
3	True	2461.0	131.518	111.677	-59.733	(no cal)	(no cal)	(no cal)	(no cal)	(no cal)
4	True	2457.0	526.077	121.878	15.584	(no cal)	(no cal)	(no cal)	(no cal)	(no cal)
5	True	2434.0	412.145	357.687	-20.011	(no cal)	(no cal)	(no cal)	(no cal)	(no cal)
6	True	2433.0	123.443	264.624	49.105	(no cal)	(no cal)	(no cal)	(no cal)	(no cal)
7	True	2394.0	309.274	106.27	-44.788	(no cal)	(no cal)	(no cal)	(no cal)	(no cal)
8	True	2381.0	228.311	350.134	22.2	(no cal)	(no cal)	(no cal)	(no cal)	(no cal)
9	True	2371.0	284.449	231.224	-58.645	(no cal)	(no cal)	(no cal)	(no cal)	(no cal)
10	True	2321.0	396.725	186.843	-5.727	(no cal)	(no cal)	(no cal)	(no cal)	(no cal)

### 複数リザルトを使用して検索上の問題を解決する方法

同一生産ロットであっても、ときには作業対象のパーツに明らかな相違がある場合や、1つのパーツに類似する2つ以上の対象物がある場合があります。

このような場合、Acceptプロパティに適切な値を設定したと思っても、他のパーツを間違えて検索してしまうことがあります。こうした問題は、原因が判明しづらく、また、Acceptプロパティの適切な値を予想するのも困難です。

リザルトダイアログは、上記の問題やその他のさまざまな問題を解決するために作成されました。

1つのパーツ上の1つの対象物を検出したい場合は、複数リザルトを表示することによって、なぜ検出したい主要な対象物として、二次的な対象物を返すのか確認することができます。このような問題は、一般的に次のような理由によって生じます。

1. サーチウィンドウ中の2つ以上の対象物が非常に類似しているために、Scoreリザルトが非常に近い値を持つ場合
2. Confusionプロパティ、またはAcceptプロパティが十分高く設定されていないため、本来検出したい対象物よりも低いスコアを持つ他の対象物がAcceptプロパティの設定値を満たしてしまう場合

以上の2つの状況は、Vision Guideを初めてご使用になる方にとって、サーチウィンドウ内で1つの対象物を検索する場合に極めて混同しやすいので、ご注意ください。

ときには本来検出したい対象物が検出され、またときには別の対象物が検出されるような場合には、リザルトダイアログを使って問題を特定してください。問題の原因を調べるための手順を、次に示します。

- (1) NumberToFindプロパティを“3”以上に設定してください。
- (2) Vision Guide 7.0開発環境からビジョンオブジェクトを実行してください。
- (3) ShowAllResultsプロパティの設定値フィールドにあるボタンをクリックして、検出結果ダイアログを表示してください。
- (4) 検出された上位3つ、あるいはそれ以上の対象物のスコアを確認してください。
- (5) 対象物が1つ、または2つだけ検出された場合 (Vision Guide 7.0は検出されたとみなされる対象物のスコアを設定するだけです)は、対象物が“1”より多く検出されるようにAcceptプロパティを少なくして、ビジョンオブジェクトを再度実行してください。(Acceptレベルは、ShowAllResultsダイアログを確認したあとで、元に戻すことができます。)
- (6) <ShowAllResultsプロパティ>ボタンをクリックして、検出結果ダイアログを開いてください。
- (7) 検出された上位3つ、あるいはそれ以上の対象物のスコアを確認してください。

上述のように、検出された上位3つ、あるいはそれ以上の対象物のスコアを確認すると、問題の原因が明確になります。ほとんどの場合、このような問題は、次に示すどちらかの場合に発生します。

1. 検出されたそれぞれの対象物が、Acceptプロパティの設定値より大きいスコアを持っている場合  
このような場合には、Confusionプロパティをより高く設定し、適切な対象物だけを常に強制的に検出するようにして、Acceptプロパティのしきい値に一致するようなその他の対象物は返されないようにしてください。また、Acceptプロパティの値を変更することもできます。
2. それぞれの対象物のスコアが非常に近接している場合  
このような場合には、次の処理を行って、本来検出したい対象物が区別できるようにしてください。
  - サーチウィンドウを再調整し、検出対象物としてランダムに返される対象物がサーチウィンドウ内部に入らないようにします。
  - 本来検出したい対象物を再度、モデルに登録します。
  - アプリケーションの照明条件を調整して、本来検出したい対象物が、まちがって検出してしまう対象物のスコアより高いスコアを取得するようにしてください。



### SPEL+ 言語から複数リザルトにアクセスする方法

すでに説明したとおり、**CurrentResult**プロパティによって、どのリザルトがリザルトリストに表示されるかを指定することができます。

また、このプロパティによって、リザルトとして何番目のリザルトを返すかを指定することができます。言い換えると、**Blob**オブジェクトから返される3番目のリザルトから**Area**リザルトを得たい場合は、**CurrentResult**プロパティを3に設定しなければなりません。

オブジェクトウィンドウのプロパティリストを使って**CurrentResult**プロパティを設定する方法については、すでに説明したとおりです。ここでは、**SPEL+**言語から複数リザルトにアクセスする方法について説明します。

**SPEL+**言語から複数リザルトへのアクセスは、得られる**Result**の隣に添字番号をもつ**CurrentResult**プロパティが参照をつけられる配列の型のように、**Result**を扱います。下の1番目の例は、3番目の**Area**リザルトを得て、それを**SPEL+**言語から領域という変数に入れる方法を示します。

```
VGet seqname.objname.Area(3), area
```

下の2番目の例は、同じく3番目の**Area**リザルトを得ますが、今回はそれを**Area()**と呼ばれる配列の第3要素の値として代入する方法を示します。

```
VGet seqname.objname.Area(3), area(3)
```

変数名はまた、上の2番目の例のように固定された要素としてよりも、配列の要素を表すのに使われます。”**var**”という変数は**Area**リザルトの添字として使われることに注目してください。

```
VGet seqname.objname.Area(var), area(var)
```

4番目の例は、1つのビジョンオブジェクトを使って、パーツのようなまとまり (10までとします) を検出すると仮定してください。ロボットでそれらのパーツ (例えばペンだとして) を選ぶとすると、検出された各パーツの座標値を表すために、**X**座標、**Y**座標、**U**座標を変数に入れる必要があります。下のコードは、これらの座標を**RobotXYU**リザルトから取り出して、あとでロボットを動かすのに使われる**X**配列、**Y**配列、**U**配列の中に座標を入れることができます。

```
Function test
  Boolean found(10)
  Integer numfound, i
  Real X(10), Y(10), U(10)

  Jump camshot      'move camera into position snap shot

  VRun seq01        'run the vision sequence to find the pens

  VGet seq01.blob01.NumFound, numfound      'how many found

  For i = 1 to numfound      'get robot coords
    VGet seq01.blob01.RobotXYU(I), found(i), X(i), Y(i), Z(i)
  Next i
  'Add code for robot motion here.....
Fend
```

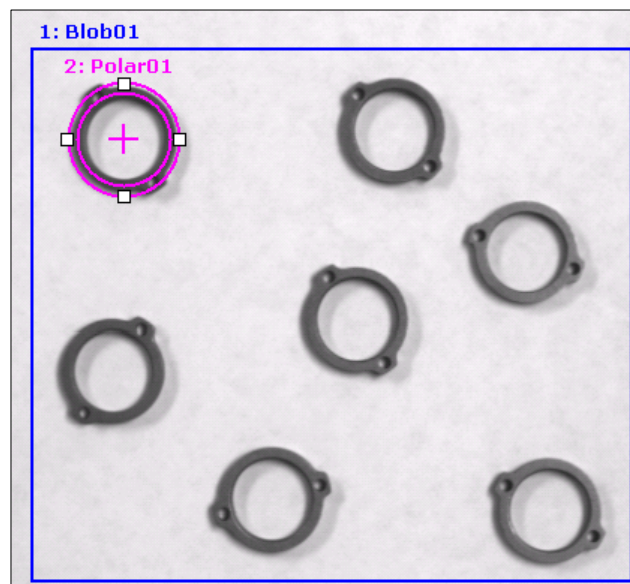
### 6.2.25 自動マルチオブジェクト検索

いくつかのビジョンオブジェクトには、複数のオブジェクトを使った自動検索を実行できます。検索に使用するオブジェクトは、他のオブジェクトからのすべてのリザルトを使用すると指定した場合、自動的に作成されます。この機能により、特徴物の検索に1つ以上のオブジェクトを設定し、実行時に親オブジェクトのすべてのリザルトに対してオブジェクトを自動的に作成、実行できます。

複数オブジェクトの自動検索は、CenterPointObjectおよびFrameと併用できます。

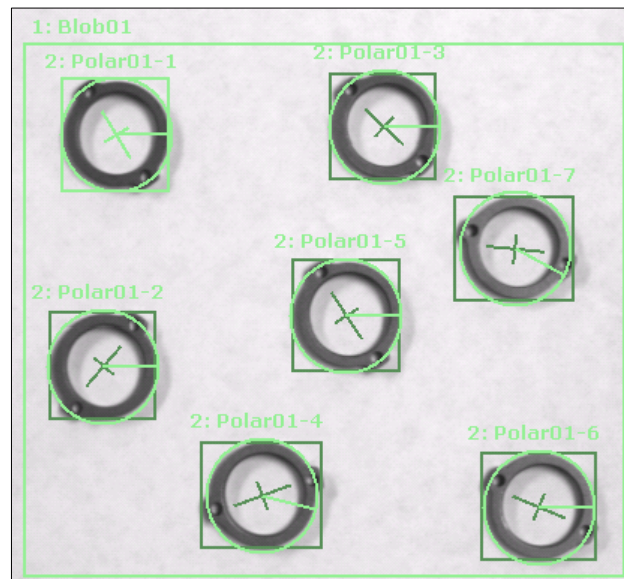
**使用例: CenterPointObject**

1. シーケンスを作成し、Blobオブジェクトを追加します。NumberToFindに“10”を設定します。
2. Polarオブジェクトを作成します。CenterPointObjectにBlobオブジェクトを設定し、CenterPntObjResultに“All”を指定します。



3. Polarオブジェクトをティーチします。

4. シーケンスを実行します。検出したブローブにPolarオブジェクトのインスタンスを作成、実行します。10個のブローブが検出されると、Blobオブジェクトの結果を中心に10個のPolarオブジェクトが作成されます。以下の図では、検出したブローブに対して、7つのPolarオブジェクトが作成されています。



自動マルチオブジェクト検索で、子オブジェクトが複数の結果を検出できる場合、子オブジェクトのそれぞれのインスタンスに対して1つの結果が検出されます。

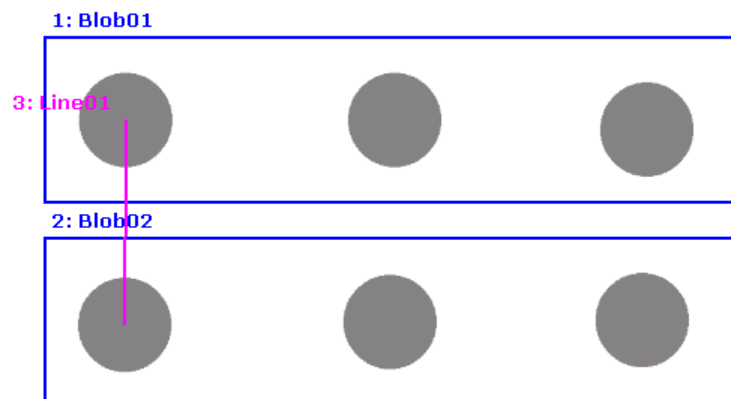
自動マルチオブジェクト検索には、Line, Edge, LineInspectorオブジェクトも使用できます。

StartPointObjectとEndPointObjectを指定し、StartPntObjResultとEndPntObjResultに“All”を設定します。

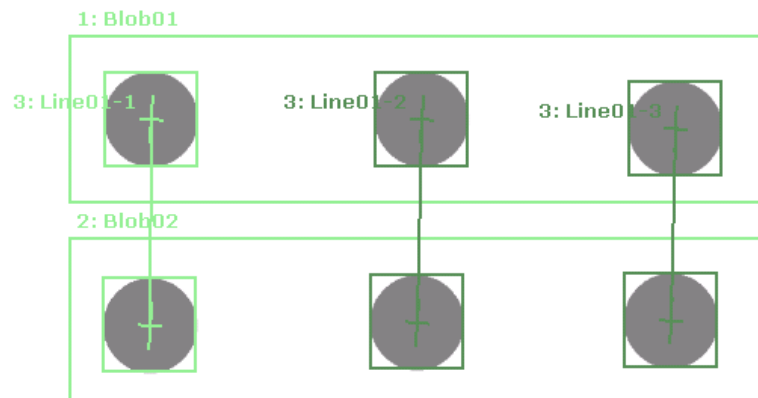
例:

1. 新規シーケンスを作成し、ImageFileに平行する2列のブローブを選択します。(下図参照)
2. Blob01を作成し、NumberToFindに“3”を、Sortに“PixelX”を設定します。1列目を検出できるようにサイズと位置を調整します。
3. Blob01をコピー、ペーストしてBlob02を作成します。2列目のブローブを検出できるようにサイズと位置を調整します。

4. Lineオブジェクトを作成します。StartPointObjectを“Blob01”に、StartPntObjResultを“All”に設定します。EndPointObjectを“Blob02”に、EndPntObjResultを“All”に設定します。

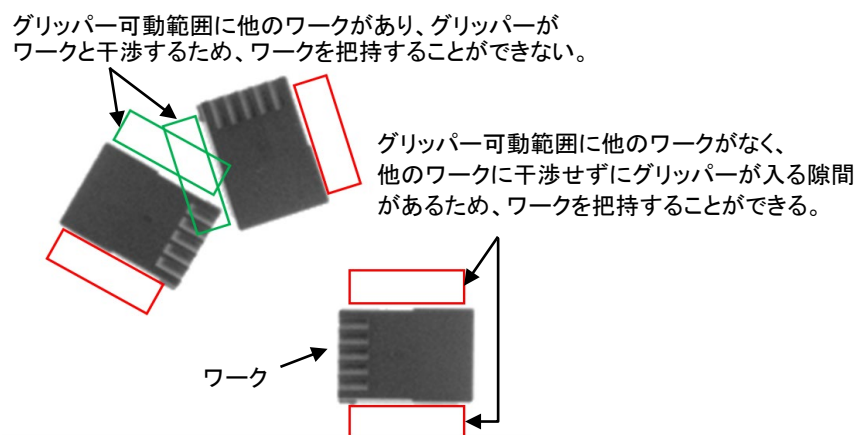


5. シーケンスを実行します。StartPointObjectとEndPointObjectの結果の各ペアに対してLineオブジェクトのインスタンスが作成されます。



## 使用例: CheckClearanceFor

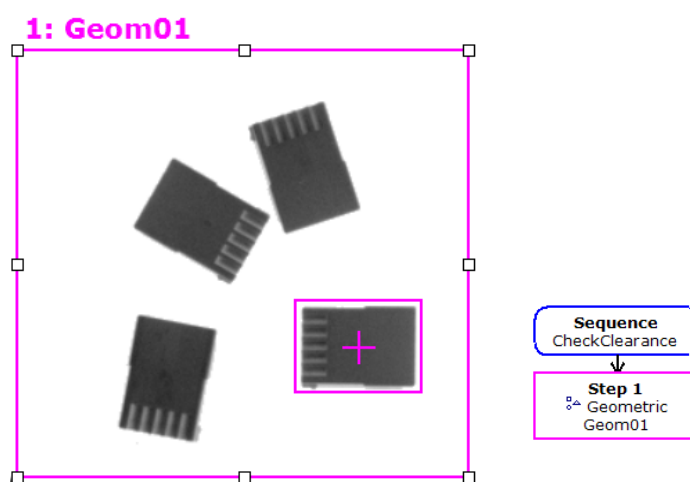
2点支持のグリッパーでワーク両端を把持するアプリケーションで使用するシーケンスを例にして、自動検索の1つである CheckClearanceFor 機能を説明します。シーケンスは、ワークを検出する“Geometric”と、把持干渉せずグリッパーを入れる隙間の有無を判定する“Blob”などの2種類のオブジェクトから構成されます。



例:

1. ワークを検出する親オブジェクトとしてGeometric (Geom01)を作成し、モデルウィンドウの位置とサイズを設定します。(下図参照)  
あわせて、関連するプロパティの設定を行います。この例では、複数ワークを検出するため、NumberToFindを“All”に設定します。

CheckClearanceFor機能で利用できる親オブジェクトは、Vision Guide 7.0 プロパティ&リザルトリファレンスを参照してください。

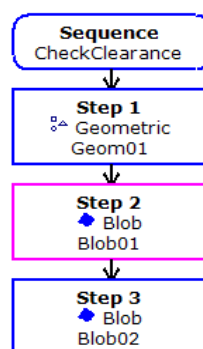
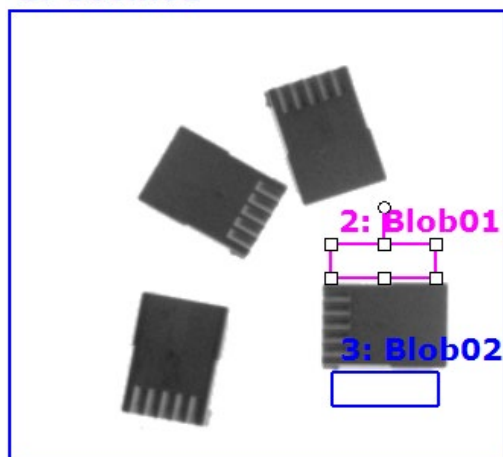


2. 把持干渉を判定する子オブジェクトとしてBlob (Blob01, Blob02)を作成し、Geometricのモデルウィンドウの両端に配置します。(下図参照)  
あわせて、関連するプロパティの設定を行います。



親オブジェクトで検出するワークが回転する場合は、子オブジェクトのSearchWinTypeを“RotatedRectangle”に設定してください。  
“Rectangle”の場合、サーチウィンドウが回転できず、回転角度が追従できなくなります。

### 1: Geom01



子オブジェクトであるBlob01とBlob02のプロパティを設定します。

#### 2.1 CheckClearanceForを“Geom01”に設定します。(下図参照)

この機能により、シーケンス実行時に、親オブジェクトのすべてのリザルトに対して子オブジェクトを自動的に作成し、実行します。

Step 2: Blob01	
Property	Value
AbortSeqOnFail	False
Caption	
CheckClearanceFor	Geom01
ClearanceCondition	Found
CurrentResult	1
Enabled	True
FailColor	Red
FillHoles	False
Graphics	All

CheckClearanceFor機能で使える子オブジェクトは、Vision Guide 7.0 プロパティ&リザルトリファレンスを参照してください。



CheckClearanceForを設定した子オブジェクトは、他のオブジェクトからは設定することができません。

(この例では、Blob01のCheckClearanceForを“Geom01”に設定した場合、Blob02のCheckClearanceForのドロップダウンリストにBlob01は表示されません。)

## 2.2 ClearanceConditionを設定します。

子オブジェクトの検出結果による把持判定を設定します。

ClearanceConditionを“NotFound”に設定して、Foundリザルトが“False”になった場合、自身のClearanceOKを“True”として、把持可能と判定します。

“Found”に設定した場合は、逆の判定となります。

この例では、ワークの両端に物体がなく、グリッパーを入れる隙間があることを確認したいので、“NotFound”に設定にします。(下図参照)

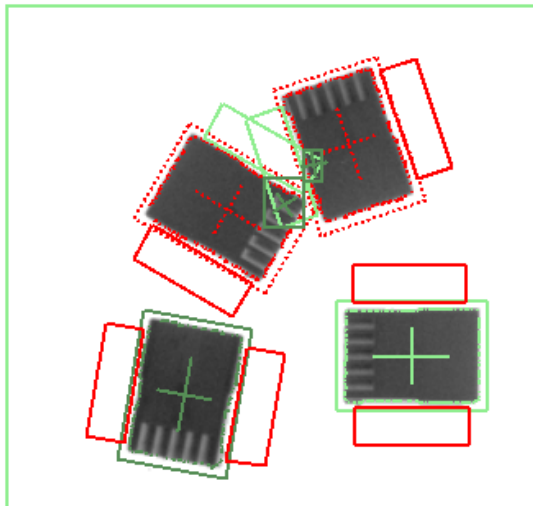
Step 2: Blob01	
Property	Value
AbortSeqOnFail	False
Caption	
CheckClearanceFor	Geom01
ClearanceCondition	NotFound
CurrentResult	1
Enabled	True
FailColor	Red
FillHoles	False
Graphics	All

## 3. シーケンスを実行します。

下図のように緑色の実線と赤色の破線で囲まれた親オブジェクトGeom01の検出結果が表示されます。

緑色の実線で囲まれた検出結果は、把持可能なワークを示します。(子オブジェクトが把持可能と判定されました。)

赤色の破線で囲まれた検出結果は、把持不可と判定されたワークを示します。



把持可能なワークかどうかを値として確認する場合は、親オブジェクトGeom01のClearanceOKを参照してください。



NOTE

CheckClearanceFor機能を使用し把持判定を行う場合は、必ずシーケンス実行させてから、親オブジェクトのClearanceOKを参照してください。親オブジェクトのClearanceOKは、子オブジェクトが実行されて初めて決定されます。

### 6.2.26 すべてのビジョンオブジェクトラベルの表示、非表示の切り替え



#### 全ラベル非表示 (ビジョンツールバーのみ)

<全ラベル非表示>ボタンは、1つのシーケンス中の複数のビジョンオブジェクトで作業するときに、画面が混雑するのを防ぐのに便利な機能です。

Vision Guideツールバーの<全ラベル非表示>ボタンは、オンオフボタンです。ボタンを押すと、選択されているビジョンオブジェクト以外のラベルが消え、表示されているオブジェクトが見やすくなります。

<全ラベル非表示>ボタンが出ているとき (押されていない状態)は、画像イメージ表示部に表示されているビジョンオブジェクトのそれぞれについてラベルが表示されます。

NOTE



<全ラベル非表示>ボタンの機能は、<全グラフィック表示>ボタンの機能と一緒に使われることがあります。この場合、<全グラフィック表示>ボタンを押してあっても、<全ラベル非表示>ボタンが押されていれば、ラベルは表示されません。

NOTE



ビジョンシーケンスで作業しているとき、<全ラベル非表示>ボタンを押していないのに表示されないビジョンオブジェクトがある場合は、そのビジョンオブジェクトのGraphicsプロパティーが“none”に設定されている可能性があります。このプロパティーが“none”になっていると、そのオブジェクトの全部についてグラフィックを表示しない、という設定になります。

作業中のプロジェクトにビジョンシーケンスがない場合は、<全ラベル非表示>ボタンが薄い灰色になっていて、アクセス不可能となっています。

### 6.2.27 すべてのビジョンオブジェクトグラフィックの表示



#### 全グラフィック表示 (ビジョンツールバーのみ)

<全グラフィック表示>ボタンは、作業中のビジョンシーケンスのビジョンオブジェクトについて、すべてのグラフィック (サーチウィンドウ、モデル原点、モデルウィンドウ、直線、ラベル)の表示を、ボタンのクリック操作だけで迅速に行う機能を備えています。

このボタンは、個々のビジョンオブジェクトのGraphicsプロパティーの設定に優先するので、各ビジョンオブジェクトのGraphicsプロパティーをそれぞれ修正しなくても、すべてのビジョンオブジェクトを迅速に表示することができます。

NOTE



<全ラベル非表示>ボタンの機能は、<全グラフィック表示>ボタンの機能と一緒に使われることがあります。この場合、<全ラベル非表示>ボタンが優先です。<全グラフィック表示>ボタンを押してあっても、<全ラベル非表示>ボタンが押してあれば、ラベルは表示されません。

作業中のプロジェクトにビジョンシーケンスがない場合は、<全グラフィック表示>ボタンが薄い灰色になっていて、アクセス不可能となっています。



### 6.2.28 カレントオブジェクトのみ表示



#### カレントオブジェクト表示 (ビジョンツールバーのみ)

シーケンスに複数のオブジェクトがある場合、必要なオブジェクトを選択し作業することが難しい場合があります。<カレントオブジェクト表示>ボタンをクリックすると、現在選択中のオブジェクトのみ表示されます。再度すべてのオブジェクトを表示するには、もう一度ボタンをクリックします。カレントオブジェクトのみ表示する場合、オブジェクトリストでどのオブジェクトを表示するかを選択できます。

# 7. ビジョンキャリブレーション

ビジョンキャリブレーションを使用すると、カメラとロボットシステムとの位置関係をロボットコントローラーに登録し、画像処理で認識した物体のイメージ画像座標位置をロボット座標位置に変換することが可能となります。

ビジョンキャリブレーションは以下の手順で作成します。

- (1) カメラ設置
- (2) キャリブレーション用の画像処理シーケンス作成
- (3) キャリブレーションウィザードによるキャリブレーションの作成
- (4) ポイントティーチング
- (5) キャリブレーション実行

キャリブレーションをより高精度に行うため、レンズ歪み、およびカメラチルト(作業平面とカメラ光軸のずれ)を補正することができます。

このレンズ歪み、およびカメラチルトを補正する場合、キャリブレーションは以下の手順で作成します。

- (1) カメラ設置
- (2) レンズ歪み、およびカメラチルト補正用の画像処理シーケンス作成
- (3) レンズ歪み、およびカメラチルト補正実行
- (4) キャリブレーション用の画像処理シーケンス作成
- (5) キャリブレーションウィザードによるキャリブレーションの作成
- (6) ポイントティーチング
- (7) キャリブレーション実行

### カメラを使用した追加キャリブレーション

- 作業面に設置されたキャリブレーションプレートにローカル平面が検出できます。  
(垂直 6 軸型ロボットの第 6 アーム(J6)に設置された可動カメラのみ)
- 水平多関節型(スカラ)ロボットの第 4 軸(J4)、または垂直 6 軸型ロボットの第 6 アーム(J6)に設置された可動カメラの、カメラ設置位置のツール座標が自動的に検出できます。  
また、水平多関節型(スカラ)ロボットの第 2 アーム(J2)に設置された可動カメラの、カメラ設置位置を増設アーム設定のパラメーターとして自動的に検出できます。
- 上向き固定カメラを使って、ロボットのハンドに取りつけられたツールのツール座標が自動的に検出できます。

[ロボットマネージャー]の[ローカル設定], [ツール設定], [増設アーム設定]のタブを選択し、これらのキャリブレーションのウィザードが開始できます。また、カメラキャリブレーションウィザードから、これらのウィザードを実行することもできます。

いずれの場合も、あらかじめ必要なビジョンシーケンスを作成しておく必要があります。

## カメラを使ったローカル検出機能を使用するための構成

ハードウェア構成	カメラを使用したローカル検出機能
コンパクトビジョン CV1	No
コンパクトビジョン CV2-S, CV2-H (ファームウェア 2.x.x.x)	No
コンパクトビジョン CV2-SA, CV2-HA (ファームウェア 3.0.0.0 以降)	OK
PC ビジョン PV1	OK

上記以外のビジョンキャリブレーション (カメラを使用したツール検出, 増設アーム検出を含む)は、どのハードウェア構成でも使用できます。

また、カメラを使った作業面のローカル検出機能を使用する場合、オプションのキャリブレーションプレートが必要です。

下表の仕様を参考に、作業面の広さに応じて適切なキャリブレーションプレートを選択してください。

## キャリブレーションプレート

	大 (CP01-L)	中 (CP01-M)	小 (CP01-S)	極小 (CP01-XS)
外形寸法 (横×縦×厚さ) [mm]	152×152×2	78×78×2	40×40×2	28×28×2
有効寸法 (横×縦) [mm]	128×128	64×64	32×32	21×21
ドットピッチ (横 / 縦) [mm]	4	2	1	0.5
ドット数	1089 (横 33×縦 33)			1849 (43×43)
材質	白色ガラス			

## 7.1 カメラ設置

各キャリブレーションについて、カメラ設置を選択します。カメラ設置方法によってキャリブレーション方法(ビジョンシーケンスの要件やキャリブレーション手順)が異なります。間違った設定をすると、正しくキャリブレーションが行えません。注意してください。

Vision Guide 7.0では、次のカメラ設置をサポートしています。

カメラ設置	説明
可動カメラ (ロボット第2軸搭載)	カメラは、水平多関節型(スカル)ロボットの第2アーム、または直角座標ロボットの第2軸(J2)に搭載されています。
可動カメラ (ロボット第4軸搭載)	カメラは、水平多関節型(スカル)ロボットの第4軸(J4)、または直角座標ロボットの第4軸(J4)に搭載されています。
可動カメラ (ロボット第5軸搭載)	カメラは、垂直6軸型ロボットの第5軸(J5)に搭載されています。
可動カメラ (ロボット第6軸搭載)	カメラは、垂直6軸型ロボットの第6軸(J6)に搭載されています。
下向き固定カメラ	カメラや対象物は移動せず、ロボットの作業領域に固定されています。このカメラで、ロボット座標系の位置情報を得ます。カメラが指定座標系のXY平面に対して垂直に設置されていない場合、レンズの歪み、およびカメラチルト補正を行う必要があります。 指定座標系とはロボットベース、およびローカル座標を指します。
上向き固定カメラ	カメラは移動せず、ロボットの作業領域(一部)が視野に入るように固定されています。例えばロボットの把持している物体の位置を確認する時に本カメラ設置を使います。 キャリブレーションターゲットは、ハンド上、もしくは把持しているワークを使います。
スタンドアローンカメラ	カメラは、どこにでも設置できます。カメラとロボットとの関連を持たせません。この方法では、ロボット座標系の位置情報を得られませんが、イメージ画像座標系からカメラ座標系への変換が可能となります。つまり、簡単な長さ測定などができるようになります。

ヒント:

レンズの歪み、およびカメラチルト補正は、全てのカメラの設置方法でも設定できます。

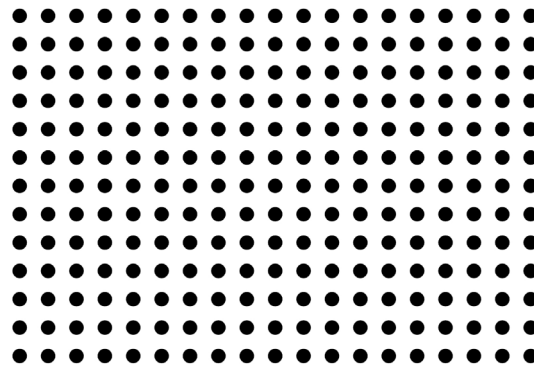
## 7.2 レンズの歪み、およびカメラチルト補正

GeometricオブジェクトやCorrelationオブジェクトの検出率を向上させるため、ロボットをより高精度に動作させるため、または寸法計測をより高精度に行うために、レンズ歪み、およびカメラチルト補正を行うことができます。

カメラ視野全体に、以下のような正方格子状のパターンが映るようにして、レンズの歪みのキャリブレーションを行ってください。格子パターンは正方形である必要があります。オブジェクトのキャリブレーションプレートも使用できます。

ここで使用する正方格子パターンは、歪み補正のために非常に重要です。できるだけ歪みのない格子パターンを作成し、カメラ視野内の作業平面上配置してください。格子パターンを配置する場合、高さ(カメラからのWD)にも注意してください。実際のワークを検出するワーク高さと同一になるようにする必要があります。

格子パターンの角度は、カメラの視野と合わせる必要はありません。



正方格子パターンは、カメラ視野の全体に映るようにすることが重要です。カメラ視野内の一部にのみ格子パターンが映っている状態でキャリブレーションを行うと、格子パターンの映っていないエリアはキャリブレーションで考慮されないため、正しく補正できません。

キャリブレーション実行時、格子パターンは、最低100点のポイントが視野内に入るようにしてください。

レンズ歪み、およびカメラチルト補正が完了すると、以降のカメラ画像は、画像取得時にレンズ歪み、およびカメラチルト補正が行われ、歪みの少ない画像を取り込めるようになります。

ヒント:

キャリブレーション実行時、格子パターンの映り方によって、キャリブレーションがエラーとなる場合があります。この場合、格子パターンを調整して再度キャリブレーションを行うことで、キャリブレーションが正しく行える場合があります。

また、カメラの向きは、カメラ光軸と格子パターンとのなす角が45°~90°となるように設置してください。45°未満の場合、キャリブレーションがエラーになります。

### 7.3 基準点とカメラポイント

基準点とは、イメージ画像座標とカメラ座標系、ロボット座標系との関係をキャリブレーションするために使用される重要な点です。

カメラポイントは、ターゲットを検出する画像を撮像するポイントです。

キャリブレーションは、その座標と基準点を対応させます。基準点とカメラポイントは、ポイントティーチのモードで、ロボットをジョグ動作させてティーチングします。

カメラの設置方法によって、必要な基準点とカメラポイントが異なります。

ロボットを使って指定する基準点には、次の3つのタイプがあります。

TaughtPoints (ティーチポイント)

EndEffector (エンドエフェクター)

UpwardCamera (上向きカメラ)

カメラの設置方法によって、下表のように選択できます。

また、ロボットを使わないスタンドアローンカメラキャリブレーションでは、これらの基準点の座標値 (各基準点間の水平、垂直距離) を、直接数値で入力します。

#### カメラ設置タイプによって選択可能な基準点のタイプ

カメラ設置タイプ	選択可能な基準点タイプ	基準点ティーチ方法	ターゲット検出方法	必要なビジョンシーケンス
可動カメラ	TaughtPoints	1点ティーチ	カメラポイント 9点ティーチ*	ターゲット1個を検出
	UpwardCamera	不要	カメラポイント 9点ティーチ*	ターゲット1個を検出
下向き固定カメラ	TaughtPoints	9点ティーチ	1ショット9点検出	ターゲット9個を検出
	EndEffector	不要	カメラポイント 9点ティーチ*	ターゲット1個を検出
上向き固定カメラ	EndEffector	不要	カメラポイント 9点ティーチ*	ターゲット1個を検出
スタンドアローンカメラ	-	9点座標指定	1ショット9点検出	ターゲット9個を検出

\* カメラポイント自動生成機能が使用できます。(下記参照)

#### 基準点ティーチ方法

基準点タイプが、“TaughtPoints (ティーチポイント)”に設定されている場合、ツールとターゲットの位置が一致するようにロボットをジョグ動作させ、1点または9点をティーチングします。

可動カメラの基準点の設定方法

参照: 7.3.1 可動カメラ基準点

下向き固定カメラと、スタンドアローンカメラの基準点の設定方法

参照: 7.3.2 下向き固定カメラ, スタンドアローンカメラ基準点

### カメラポイント9点ティーチ

ターゲット検出が、“カメラポイント9点ティーチ”の場合は、9回の撮像でそれぞれ1つ、全部で9つのターゲットの位置を検出します。

キャリブレーション実行時には、カメラ視野内の指定した9つの領域にターゲットを移動させて撮像し、ターゲットを検出します。

ポイントティーチのモードにおいては、必要なカメラポイント9点を、画像の中のそれぞれ適切な位置に検出されるように映像を見ながらジョグをしてティーチします。

あらかじめ、対象のカメラで1つのターゲットを検出するビジョンシーケンスを作成しておく必要があります。

シーケンスの作成方法

参照: 7.4.1 ターゲット1個を検出するビジョンシーケンス

### カメラポイント自動生成

ターゲット検出が、“カメラポイント9点ティーチ”の場合は、カメラポイント自動生成の機能が使用できます。

9点のカメラポイントをティーチする代わりに、視野中央のカメラポイント1点をティーチするだけで他の8点を自動的に生成できます。

注意: カメラポイント自動生成の機能を使用すると、キャリブレーション実行時には、自動的にロボットを動かしてカメラポイントを生成します。ロボットと周辺装置の干渉に注意してください。またカメラポイント自動生成中のエラーを回避するため、各関節が伸びる特異点近傍姿勢を避けて使用してください。

### 1ショット9点検出

ターゲット検出が、“1ショット9点検出”のキャリブレーションでは、1回の撮像で9つのターゲットの位置を検出します。

あらかじめ対象のカメラで9つのターゲットを検出するビジョンシーケンスを作成しておく必要があります。

シーケンスの作成方法

参照: 7.4.2 ターゲット9個を検出するビジョンシーケンス

### 7.3.1 可動カメラ基準点

このキャリブレーションでは、1個の基準点を必要とします。

基準点には、ロボットをジョグして教示された点(基準点タイプ=ティーチポイント)か、上向き固定カメラで検出された点(基準点タイプ=上向きカメラ)を使用できますが、上向き固定カメラで検出された点の方が精度も高く、自動的に実施できるので、使用できる場合は積極的に使用するようにしてください。

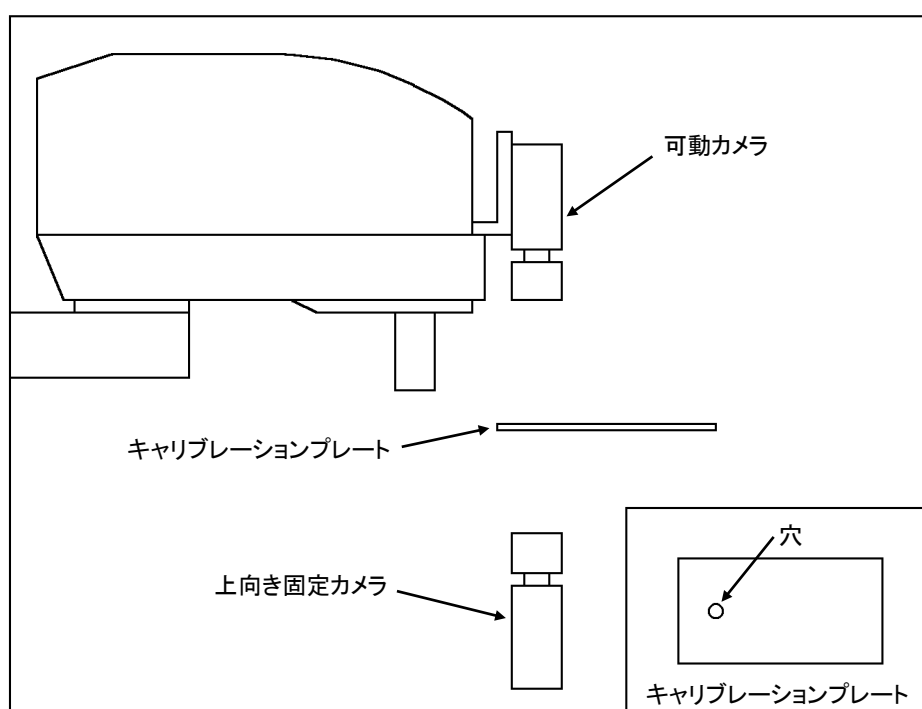
教示された基準点の2つの例を示します。

- ロボット作業領域内にあるパーツまたはキャリブレーションターゲット (1点)
- ロボットハンドに取りつけられたロッドなどをはめこむことができる作業領域内の穴

基準点を検出するのに上向き固定カメラを使用する場合には、「可動カメラと上向き固定カメラの両方からのぞき込むことができる穴のある薄いプレート」、キャリブレーションプレートが必要となります。

キャリブレーションをする間、上向き固定カメラ(キャリブレーション済)は、プレート上の基準点となる穴に位置します。続いて可動カメラについて、9個の位置で、その基準点となる穴を検索することでキャリブレーションが行われます。

可動カメラのキャリブレーションのための基準点検出に、上向き固定カメラを使用すると精度が高くなるのは、上向き固定カメラのキャリブレーションにおいて、ロボットツールは各カメラ位置について180°回転し、ロボットU軸の中心位置を正確に決定することができるからです。このようにして、キャリブレーションされた上向き固定カメラにより正確に検出される点を基準点として使用し、可動カメラがキャリブレーションされるので、ロボット教示による基準点より精度が一般的に高くなります。



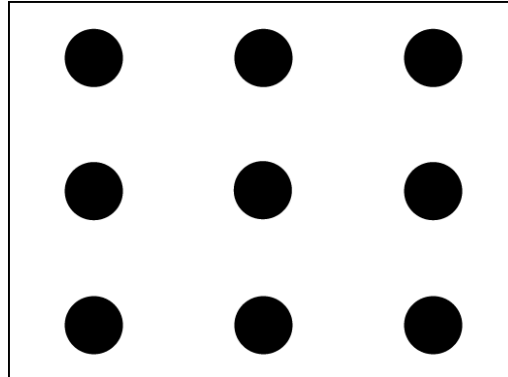
上向き固定カメラを使用した可動カメラのキャリブレーション



### 7.3.2 下向き固定カメラ, スタンドアローンカメラ基準点

基準点タイプが、“TaughtPoints(ティーチポイント)”に設定されている次のキャリブレーションでは、9個のターゲットを備えたキャリブレーションターゲットプレート(シート)が必要です。

下向き固定カメラ  
スタンドアローンカメラ



下向き固定カメラ, スタンドアローンカメラのキャリブレーションターゲット

下向き固定カメラキャリブレーションでは、ターゲットは、ロボットハンドのロッドなどが滑り込むことができるプレート上の穴などします。ターゲット間の距離は、正確である必要はありません。基準点が教示されると、キャリブレーションソフトウェアは、ロボット位置を読み取ります。

スタンドアローンカメラキャリブレーションでは、ターゲットは、パターンシートなどします。ターゲット間の水平距離と垂直距離の情報が必要となります。基準点を教示するとき、キャリブレーションダイアログから、これらの距離の値を入力してください。

### 7.3.3 TwoRefPointsによる教示

可動カメラキャリブレーションと下向き固定カメラキャリブレーションの基準点を教示するとき、一度基準点を教示すると、任意でその位置から180°回転した位置で再教示するように、メッセージが表示されます。この処理により、システムはロボット座標系における基準点のより正確な位置を決定することができます。しかし、ロボットツールが正確に定義され、キャリブレーションの設定で特定されている場合には、180°回転位置での再教示手順は省略することができます。

180°回転位置での再教示手順を省略する場合は、TwoRefPointsキャリブレーションプロパティが、“False”に設定されていることを確認してください。

ツール定義方法の詳細は、下記を参照してください。

#### 11. SPEL<sup>+</sup>でVision Guide 7.0を使う方法のツールの定義

### 7.4 キャリブレーション用ビジョンシーケンスの作成

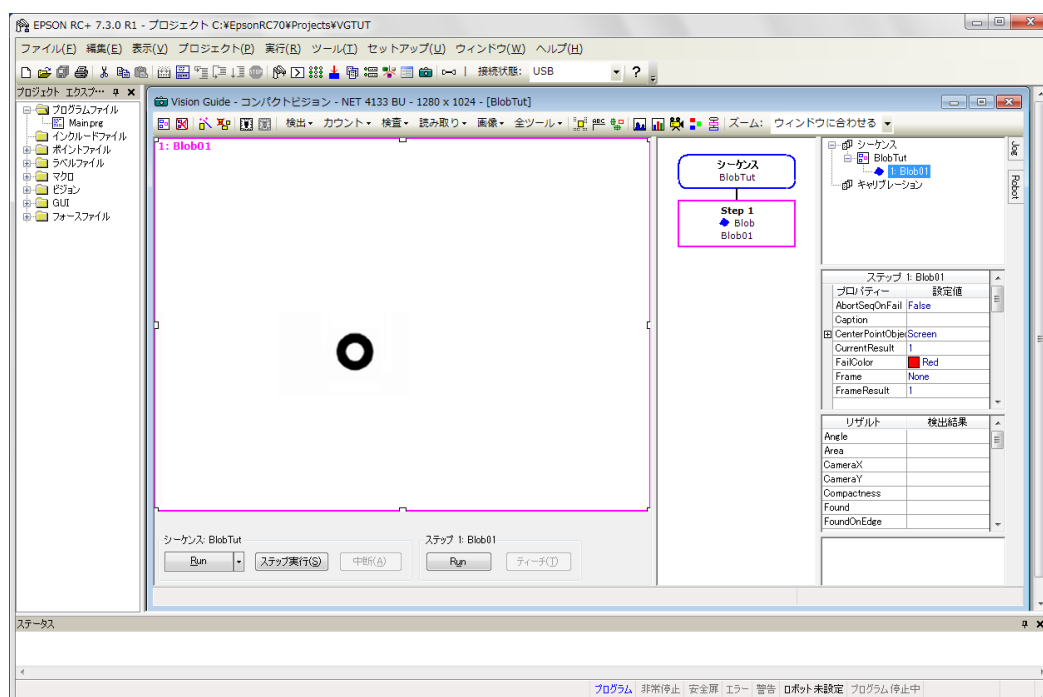
カメラキャリブレーションを実施する前に、キャリブレーションターゲットを検出するためのビジョンシーケンスを作成しなければなりません。

#### 7.4.1 ターゲット1個を検出するビジョンシーケンス

可動カメラ、上向き固定カメラ、または基準点タイプが“EndEffector (エンドエフェクター)”に設定されている下向き固定カメラの場合、シーケンスは視野の全体で1つのキャリブレーションターゲットを検索します。

サーチウィンドウのサイズが、画像イメージ表示部全体と同じ大きさに定義されていることを確認してください。

キャリブレーションをする間、キャリブレーションソフトウェアでは、シーケンス中の最後のオブジェクトのXリザルトとYリザルトを使用します。使用できるオブジェクトについては、「Vision Guide 7.0 プロパティ&リザルトリファレンス PixelXYURリザルト」を参照してください。



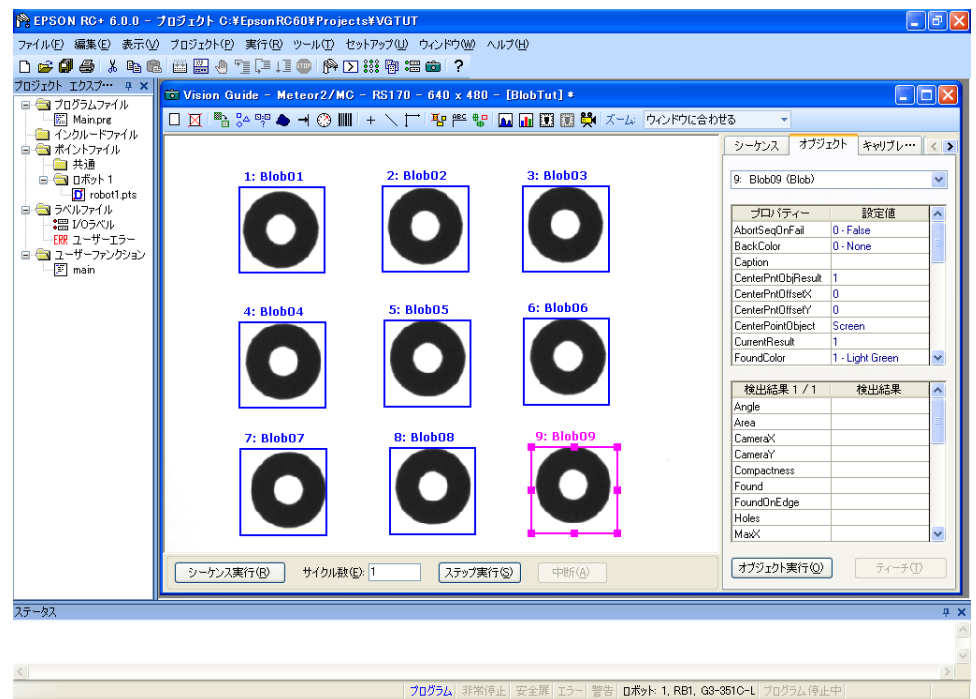
## 7.4.2 ターゲット9個を検出するビジョンシーケンス

基準点タイプが、“TaughtPoints (ティーチポイント)”に設定されている下向き固定カメラ、またはスタンドアローンカメラのキャリブレーションに必要なビジョンシーケンスでは、9つのキャリブレーションターゲットの位置が検出されなければなりません。次の2つの方法があります。

- (1) ターゲットのそれぞれにオブジェクトを作成します。つまり、全部で9個のオブジェクトを作成します。

もしくは、

- (2) 1つのオブジェクト(Blobオブジェクト、やCorrelationオブジェクトなど)を作成し、複数のリザルトを返すように設定します。つまり、NumberToFindを9に設定し、9個のリザルトが返されるようにします。



## スタンドアローンカメラのキャリブレーションを行うために配置されたビジョンオブジェクト

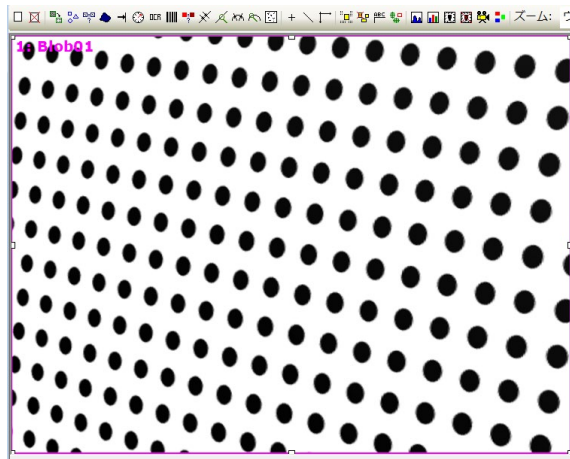
方法1の場合、オブジェクトは、ターゲットが教示される順番で、画像イメージウィンドウ内に配置されていなければなりません。1番目のオブジェクトは左上コーナーに、2番目のオブジェクトは中央上に、3番目のオブジェクトは右上コーナーの位置に設定します。

下向き固定カメラキャリブレーションは、オブジェクトの位置決めに上図の1, 2, 3, 6, 5, 4, 7, 8, 9の順番で行ってください。ロボットの動作にとってこの順番による位置決めが最も効率的で、キャリブレーションをより速く完了できるからです。

### 7.4.3 歪み補正に必要なビジョンシーケンス

歪み補正を実行するには、あらかじめ次の手順でビジョンシーケンスを作成しておく必要があります。

- (1) 正方格子パターンを設置します。
- (2) 正方格子パターンを検出するためのシーケンスを作成します。
- (3) 必要に応じて、Lampプロパティを選択します。
- (4) Blobオブジェクトを作成し、サーチエリアを画面全体に広げます。
- (5) NumberToFind プロパティを、“All”に設定します。
- (6) RejectOnEdge プロパティを、“True”に設定します。
- (7) ドットが検出できるように ThresholdAuto, ThresholdHigh, ThresholdLow, MinArea, MaxArea などの各プロパティを設定します。
- (8) シーケンス作成後、作成したシーケンスを実行します。100 点以上のポイントが検出されるかテストします。



チルト角のついたカメラによる正方格子パターンの撮影画像例

### 7.4.4 ローカル設定, ツール設定, 増設アーム設定に必要なビジョンシーケンス

カメラを使ってキャリブレーションプレートのローカル座標を検出するには、キャリブレーションプレートを認識するためのビジョンシーケンスが必要です。


オプションのキャリブレーションプレートのドットが明瞭に認識できるように、シーケンスのプロパティ (ExposureTime (露光時間)など)を設定します。ビジョンオブジェクトは不要です。

それ以外のカメラを使ったツール設定, 増設アーム設定のキャリブレーションを実行するには、キャリブレーションターゲットの位置を検出するためのビジョンシーケンスが必要です。「7.4.1 ターゲット1個を検出するビジョンシーケンス」と同様に、ターゲット1個の位置を検出するビジョンオブジェクトを持つシーケンスを作成してください。

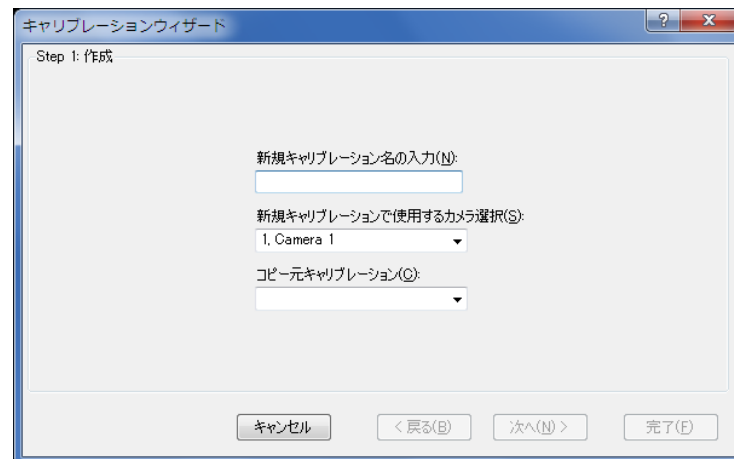
## 7.5 キャリブレーションGUI

ここでは、キャリブレーションで利用するGUIダイアログについて説明します。

### 7.5.1 新規キャリブレーションの作成

Vision Guideツールバー -  <新規キャリブレーション>ボタンをクリックします。

次の画面が表示され、キャリブレーションウィザードが開始されます。



新規キャリブレーション名を入力します。キャリブレーションするカメラを選択した後、[コピー元キャリブレーション]を指定することで、既存のキャリブレーションをコピーできます。

キャリブレーションウィザードでは、キャリブレーション名の入力後、どの画面でも<完了>ボタンをクリックしてウィザードを終了できます。この場合、「7.5.3 キャリブレーションプロパティーとリザルト」を参照して、必要なプロパティーの設定をする必要があります。また、<戻る>ボタンをクリックして、これまでに設定した項目の修正ができます。


<次へ>ボタンをクリックすると、次の入力画面に進みます。

名前がすでに別のキャリブレーションに存在する場合は、エラーメッセージが表示されます。

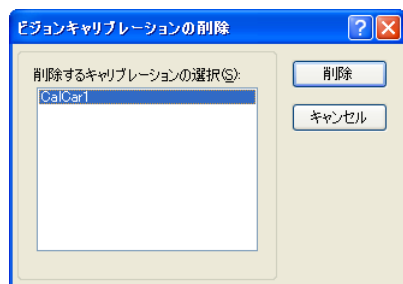
各画面の説明にしたがって、ウィザードを完了してください。設定は、カメラの設置位置や向きによって異なります。各設定の詳細は、「7.6 キャリブレーション手順」で説明しています。

ウィザードを完了させると、ウィザードで設定したキャリブレーションがツリーに追加されます。ツリーの中で、キャリブレーションを選択している間は、キャリブレーションのプロパティーとリザルトが画面に表示されます。ウィザードで設定した項目は、このプロパティー画面でいつでも修正できます。

## 7.5.2 キャリブレーションの削除

Vision Guideツールバー -  <キャリブレーション削除>ボタンをクリックします。

次の画面が表示されます。



削除したいキャリブレーションを選択し、<削除>ボタンをクリックします。

## 7.5.3 キャリブレーションプロパティーとリザルト

キャリブレーションプロパティーには、キャリブレーションウィザードで設定された内容が設定されています。キャリブレーションを作成した後、Vision Guideウィンドウのキャリブレーションウィンドウでキャリブレーションプロパティーを変更できます。

プロパティー	説明
ApproachPoint	キャリブレーション実行中、各カメラポイントに移動する起点となるアプローチポイントを設定
AutoCamPoints	カメラポイントを自動的に生成
AutoReference	可動カメラのキャリブレーション実行時に、キャリブレーション基準点を自動計算するかを設定
AutoRefFinalRotation	基準点を自動計算するときの、ツールの最終的な回転角度を設定
AutoRefInitRotation	基準点を自動計算するときの、アームやツールの最初の回転角度を設定
AutoRefMode	基準点の自動計算モードをRough, Fine, Manualとして指定 Fineを使用すると、アームをより大きく動かして高精度に基準点を計算します。Manualを使用すると、回転角度や許容量を手動で入力できます。
AutoRefMoveMode	基準点を自動計算するときの、ロボットの動作方法を指定 (6軸ロボットのみ)
AutoRefTolerance	基準点を自動計算するときの、ビジョン検出の誤差に対する許容量を設定
Camera	現在選択されているキャリブレーションで使用するカメラを指定

プロパティ	説明
CameraOrientation	<p>カメラの方向を選択</p> <p>次の7つの選択項目があります。</p> <p><b>Mobile J2 (可動カメラ: ロボット第2軸搭載)</b> カメラは水平多関節型(スカラ)ロボットの第2アーム、直角座標型ロボットの第2軸に搭載されます。</p> <p><b>Mobile J4 (可動カメラ: ロボット第4軸搭載)</b> カメラは水平多関節型(スカラ)ロボットの第4軸、直角座標型ロボットの第4軸に搭載されます。</p> <p><b>Mobile J5 (可動カメラ: ロボット第5軸搭載)</b> カメラは垂直6軸型ロボットの第5アームに搭載されます。</p> <p><b>Mobile J6 (可動カメラ: ロボット第6軸搭載)</b> カメラは垂直6軸型ロボットの第6アームに搭載されます。</p> <p><b>Fixed downward (下向き固定カメラ)</b> カメラは移動せず、ロボット座標系の位置情報を提供します。</p> <p><b>Fixed upward (上向き固定カメラ)</b> カメラは移動せず、ロボット座標系の位置情報を提供します。</p> <p><b>Standalone (スタンドアローンカメラ)</b> カメラは移動せず、ロボット座標系の位置情報は提供しません。カメラ座標系の位置情報を提供します。</p>
DistCorrectCal	レンズ歪み、およびカメラチルト補正を実行
DistCorrectEnable	レンズ歪み、およびカメラチルト補正の有効/無効を設定
DistCorrectTargetSeq	<p>レンズ歪み、およびカメラチルト補正で使用するビジョンシーケンスを指定</p> <p>カレントプロジェクトにあるどのシーケンスでも可能です。</p>
DistCorrectType	使用する歪み補正のタイプを設定
Lamp	キャリブレーションが開始されると、自動的にオンになる任意のI/O出力ビットです。
LampDelay	<p>ランプが点灯するまでの遅延を秒単位で設定</p> <p>ランプが点灯するまでの所要時間を遅延として許容します。</p>
LJMMode	ポイントデータの姿勢フラグを制御するためのモードを設定
MaxMoveDist	アーム先端の移動距離の制限を設定
MotionDelay	キャリブレーションのロボット動作の待ち時間をミリ秒単位で設定
PointsTaught	現在選択されているキャリブレーションにポイントを教示
ReferenceType	<p>基準点の選択によって、キャリブレーションで使用される基準点のタイプを指定</p> <p>可動カメラのキャリブレーションでは、基準点の指定方法が2つあります。教示されたポイントを基準点とする方法と、上向き固定カメラで基準点を指定する方法です。</p>

プロパティ	説明
RobotArm	キャリブレーション中に使用するアーム番号を指定 通常は、“0”に設定します。
RobotAccel	キャリブレーション中に適用される加速度を設定
RobotLimZ	可動カメラのキャリブレーションの最初の動作で使用する LimZ値を設定 (スカラロボットのみ)
RobotLocal	現在のキャリブレーションに使用するローカル番号を指定 使用するローカルは、事前に定義しなければなりません。
RobotNumber	現在選択されているキャリブレーションで使用するロボット番号を指定
RobotSpeed	キャリブレーション中に移動するロボットの速度を設定 速度を高速にする場合には、加速度も高い値に設定する必要があります。
RobotTool	キャリブレーション中に使用するツール番号を指定 使用するツール番号は、事前に定義しなければなりません。
RobotXOffset	ロボット座標系における検出パーツのX座標位置に対してオフセット値を設定
RobotYOffset	ロボット座標系における検出パーツのY座標位置に対してオフセット値を設定
RobotXYRotateOffset	XY オフセット値 (RobotXOffset, RobotYOffset)を、Angle リザルトにしたがって回転するか否かを指定
RobotUOffset	ロボット座標系における検出パーツのU座標位置に対してオフセット値を設定
ShowConfirmation	プログラム実行時にオペレーターの承認を得るためにキャリブレーション結果のダイアログを表示するかを設定
TargetSequence	カメラのキャリブレーションに使用されるビジョンシーケンスを指定  カレントプロジェクトにあるどのシーケンスでも可能です。
TwoRefPoints	キャリブレーションで2つの基準点を使用するかを指定
UpwardLamp	上向き固定カメラシーケンスが開始されると、自動的にオンになる任意のI/O出力ビットです。
UpwardSequence	上向き固定カメラのキャリブレーションに使用されるビジョンシーケンスを指定  このリストは、上向き固定カメラを使用して基準点を指定する可動カメラキャリブレーションの実行時にのみアクティブとなります。



リザルト	説明
CalComplete	キャリブレーションの終了状態
CallImageSize	キャリブレーションを実行したときの画像サイズ
DistCorrectCalComplete	レンズ歪み、およびカメラチルト補正が完了しているか否かを返します。
FOVHeight	視野の高さ (単位: mm)
FOVWidth	視野の幅 (単位: mm)
XAvgError	カメラ座標 X 方向の平均偏差
XMaxError	カメラ座標 X 方向の最大偏差
XmmPerPixel	カメラ座標 X 方向の、1 ピクセルあたりの長さ (mm)
XTilt	カメラ座標 X 方向の傾き
YAvgError	カメラ座標 Y 方向の平均偏差
YMaxError	カメラ座標 Y 方向の最大偏差
YmmPerPixel	カメラ座標 Y 方向の、1 ピクセルあたりの長さ (mm)
YTilt	カメラ座標 Y 方向の傾き

## 7.5.4 歪み検出

キャリブレーションのプロパティを操作して、歪み補正のための歪み検出を実行します。歪み検出は、他のビジョンキャリブレーションのポイントティーチを実行する前に行う必要があります。

歪み検出をする手順は以下の通りです。キャリブレーションウィザードの中で歪み補正の設定をした場合は、手順(1)は省略します。



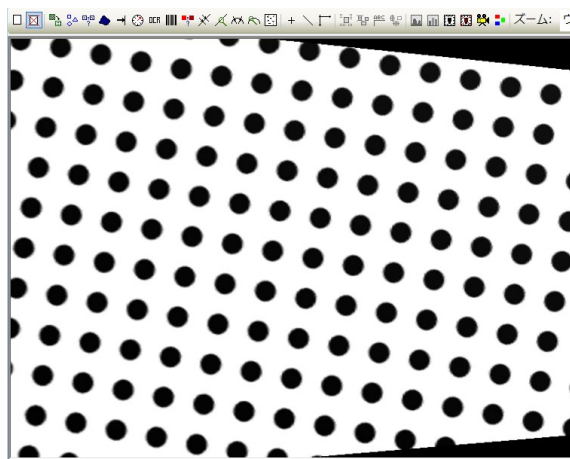
- (1) DistCorrect プロパティの下にある[Enabled]を“True”に設定し、TargetSeq プロパティに、正方格子パターンを検出するためのシーケンスを選択します。

シーケンスの作り方

参照: 7.4.3 歪み補正に必要なビジョンシーケンス

- (2) DistCorrect プロパティの下にある Cal プロパティを選択し、正方格子パターンの検出を実行します。

歪み検出をしたキャリブレーションをビジョンシーケンスのCalibrationに設定してシーケンス実行をすると、補正された画像を確認できます。



レンズ歪み、およびカメラチルトの補正をした後の画像例

### 7.5.5 ポイントティーチング

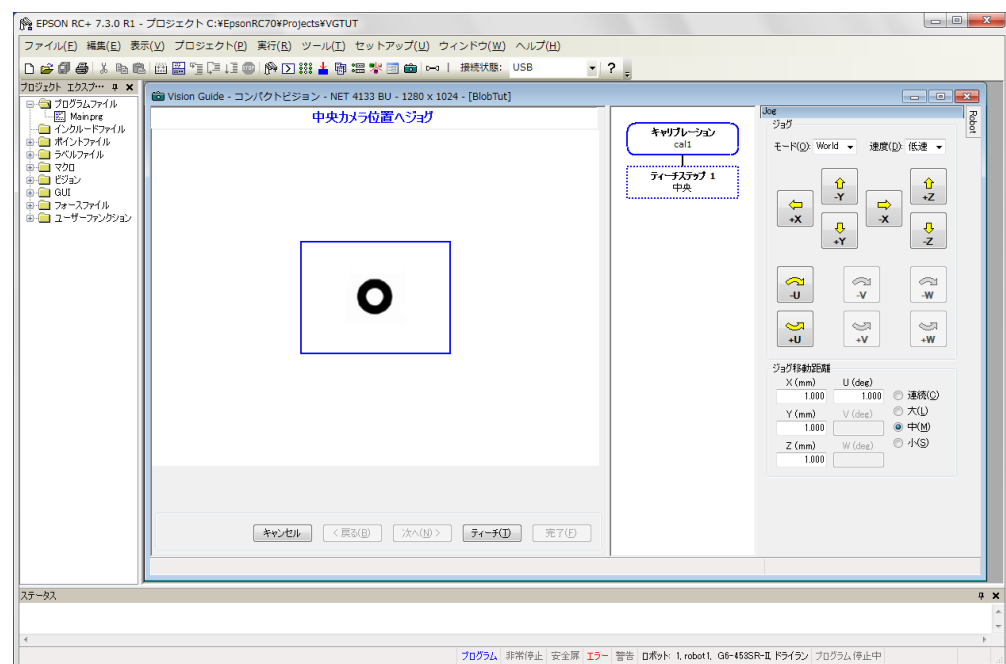
Vision Guideウィンドウのシーケンスツリー、またはキャリブレーションツリーで、キャリブレーションのひとつを選択すると、<ポイントティーチング>ボタンが表示されます。

<ポイントティーチング>ボタンをクリックすると、Vision Guideウィンドウが[キャリブレーションポイントのティーチング]モードに変わり、現在選択されているキャリブレーションにポイントを教示することができます。

[キャリブレーションポイントのティーチング]画面は、現在選択されているキャリブレーションスキームに、キャリブレーションポイントを教示するのに使用します。

ティーチするポイントは、キャリブレーションの設定によって異なります。

キャリブレーションの設定により必要なポイントティーチ画面が表示されます。画面の説明にしたがってジョグ動作を行い、ポイントティーチをしてください。



[キャリブレーションポイントのティーチング]ダイアログ

ウィンドウの左側には、画像が表示されています。右端の[ジョグ]タブを選択すると、ジョグのウィンドウが表示されます。

画面の下方には、メッセージを表示するメッセージボックスがあります。メッセージに従って処理をし、メッセージボックスの右側にある<次へ>ボタンをクリックすると、次のステップ実行を継続することができます。

すべてのステップ実行が完了すると、メッセージボックスが表示され、処理が完了したことを示します。

表示されるメッセージは、ポイントを教示するキャリブレーションのタイプによって異なります。

表示メッセージ	説明
固定基準位置ヘジグ	可動カメラキャリブレーションを行うとき、教示されたポイントを基準点として使う場合に表示  (基準点を上向き固定カメラで自動的に検索する場合は表示されません。)  ハンド(ロッドなど)がキャリブレーションターゲット位置に来るまで、ロボットをジグしてください。
180°回転して 固定基準位置ヘジグ	オプションの処理手順 この機能を有効にするには、ポイントのティーチング前にTwoRefPointsをTrueに設定します。ロボットツールが正確に定義されている場合には、TwoRefPointsをFalseのままにします。 キャリブレーションツールを使わない場合には、TwoRefPointsをTrueに設定します。ロボットを現在の位置から180°回転して、ハンド(ロッドなど)がキャリブレーションターゲット位置に来るようにジグしてください。
左上カメラ位置ヘジグ	キャリブレーションターゲットが、画像イメージ表示部の左上コーナーに表示されるまで、ロボットをジグしてください。
中央上カメラ位置ヘジグ	キャリブレーションターゲットが、画像イメージ表示部の中央上に表示されるまで、ロボットをジグしてください。
右上カメラ位置ヘジグ	キャリブレーションターゲットが、画像イメージ表示部の右上コーナーに表示されるまで、ロボットをジグしてください。
中央右カメラ位置ヘジグ	キャリブレーションターゲットが、画像イメージ表示部の中央右に表示されるまで、ロボットをジグしてください。
中央カメラ位置ヘジグ	キャリブレーションターゲットが、画像イメージ表示部の中央に表示されるまで、ロボットをジグしてください。
中央左カメラ位置ヘジグ	キャリブレーションターゲットが、画像イメージ表示部の中央左に表示されるまで、ロボットをジグしてください。
左下カメラ位置ヘジグ	キャリブレーションターゲットが、画像イメージ表示部の左下コーナーに表示されるまで、ロボットをジグしてください。
中央下カメラ位置ヘジグ	キャリブレーションターゲットが、画像イメージ表示部の中央下に表示されるまで、ロボットをジグしてください。
右下カメラ位置ヘジグ	キャリブレーションターゲットが、画像イメージ表示部の右下コーナーに表示されるまで、ロボットをジグしてください。

#### 7.5.6 キャリブレーションボタン

シーケンス、またはキャリブレーションツリーでキャリブレーションの1つを選択すると、<キャリブレーション実行>ボタンが表示されます。<キャリブレーション実行>ボタンをクリックして、キャリブレーションを開始します。ロボットが動作する場合は、ロボット動作開始前に確認メッセージが表示されます。

### 7.5.7 [キャリブレーションの完了]ダイアログ

キャリブレーションサイクルが完了すると、次のダイアログが表示されます。

このダイアログでは、今回のキャリブレーションと前回のキャリブレーションで得られたキャリブレーションの値の概要を表示します。

今回が初めてのキャリブレーションの場合は、前回のキャリブレーションの値は空白で表示されます。

前回の結果		今回の結果	
1ピクセルあたりのX mm:		1ピクセルあたりのX mm:	0.1545
1ピクセルあたりのY mm:		1ピクセルあたりのY mm:	0.1551
最大X方向偏差:		最大X方向偏差:	0.0686
最大Y方向偏差:		最大Y方向偏差:	0.0660
平均X方向偏差:		平均X方向偏差:	0.0000
平均Y方向偏差:		平均Y方向偏差:	0.0000
X 傾き:		X 傾き:	0.92
Y 傾き:		Y 傾き:	-0.35
視野の大きさ		視野の大きさ	98.91 mm X 74.47 mm

キャリブレーションサイクルが完了した後で表示される、キャリブレーション結果を示すダイアログ

次の表は、[キャリブレーションの完了]ダイアログに表示される値です。

結果を確認した後で、新規キャリブレーション値を使用する場合には、<OK>ボタンをクリックしてください。また、中断する場合には、<キャンセル>ボタンをクリックしてください。

**ヒント：** 偏差の値が1 mm以上の場合や、傾きの値が1以上の場合にはキャリブレーションが正常に完了していないことが考えられます。

キャリブレーション時にキャリブレーションポイントの検出が正常に行われているか、ロボットのキャリブレーションポイントや基準点がずれていないかを確認してください。

起動時にこのダイアログを無効にするには、ShowConfirmationプロパティを“False”に設定します。

**Note：** テレセントリックレンズを使用している場合は、傾きに異常値が表示されることがあります。

値	説明
1ピクセルあたりの X方向長さ、Y方向長さ(mm)	カメラの解像度 (1ピクセルあたりの幅と高さの平均)
最大X方向偏差 最大Y方向偏差	キャリブレーションする途中で生じる最大偏差  これらの最大偏差の値は、1ピクセルあたり1mm以下の値でなければなりません。最大偏差の値が1mm以下でない場合、その原因として、基準点が適切に教示されていないこと、あるいは、不適切な教示や不適切な光条件により、キャリブレーションターゲットの置き方に一貫性がないことが考えられます。
平均X方向偏差 平均Y方向偏差	キャリブレーションする途中で生じる平均偏差
X傾き、Y傾き	カメラの傾きを相対的に表す  傾き方向は、画像イメージバッファ座標系中のカメラが捉えた方向です。つまり、X傾きでは、プラスの値は右方への傾きを示し、マイナスの値は左方への傾きを示します。Y傾きでは、プラスの値は下方への傾きを示し、マイナスの値は上方への傾きを示します。  Vision Guide 7.0には、カメラの傾きを補償する機能がありますが、傾きの値は1.0以下に保つようにしてください。傾きの値を1.0以下に保つことにより、パーツをより正確に、かつ反復的に検出することができます。  傾きの値が高い場合には、しばしば基準点が適切に教示されていないことが原因になっています。たとえば、スタンドアローンカメラのキャリブレーションを行うとき、キャリブレーションターゲット座標を入力しますが、ここで不正確なターゲット座標値を入力すると、傾きの値が高くなってしまいます。
視野の大きさ	カメラ視野の幅と高さ (単位: ミリメートル)

## 7.6 キャリブレーション手順

ここでは、それぞれのカメラ設置について、キャリブレーションの手順を説明します。

### 7.6.1 キャリブレーション手順: 可動カメラ

ロボットに搭載されたカメラからオブジェクトを検索し、その位置をロボット座標系に取り込みます。

#### 準備1: ロボットへのカメラ取り付け

- ロボットにカメラを取りつけます。  
どのような角度でも取りつけることができます。ただし、カメラは使用するローカル座標系に対して垂直 (ローカル座標系のZ軸方向)に合わせて取りつけてください。

#### 準備2: 基準点のタイプの決定

基準点は、次の3つの決定方法があります。

##### Auto reference

手動で教示したポイントを使う方法

上向き固定カメラで検出されたポイントを使う方法

Auto referenceを使用するためには、Auto referenceプロパティを”True”に設定します。高精度を得るために、AutoRefModeを”Fine”に設定してください。

注意: Auto reference機能を使用すると、キャリブレーション実行に、自動的にロボットを動かして基準点座標を取得します。ロボットと周辺装置の干渉に注意してください。特に水平多関節型(スカラ)の第2アーム(J2)に設置されたカメラキャリブレーションにおいてFineモードを使用する場合は、キャリブレーション中にロボットの姿勢が大きく移動(左ハンド姿勢と右ハンド姿勢の切り替わり)するため注意してください。またAuto reference機能処理中のエラーを回避するため、各関節が伸びる特異点近傍姿勢を避けて使用してください。

精度の高い基準点を得るためには、上向き固定カメラを使用してください。

詳しくは、「7.6.3 キャリブレーション手順: 上向き固定カメラ」を参照してください。

Auto reference、もしくは上向き固定カメラで検出した基準ターゲットを使用しない場合は、基準点を教示するのに、次の2つのツールのどちらかを使用してください。

- 第4軸に取りつけられ、第4軸の中心を通して伸びるロッドを使用してください。
- パーツのピックアップに用いるグリッパーなどのツールを、キャリブレーション基準点に動作させることができる場合は、そのツールを使ってください。

これらのロッドやツールに(TLSetコマンドを使って)ツールを定義する場合は、基準点を教示するとき、180°回転位置での再教示をする必要はありません。

#### 準備3: 正方格子パターンを検出するためのビジョンシーケンスの作成 (歪み補正を行う場合のみ)

- (1) キャリブレーションプレートを作成します。

(2) 以下を参照し、ビジョンシーケンスを作成します。

### 7.4.3 歪み補正に必要なビジョンシーケンス


#### 準備4: キャリブレーション基準ターゲットを検出するためのビジョンシーケンスの作成

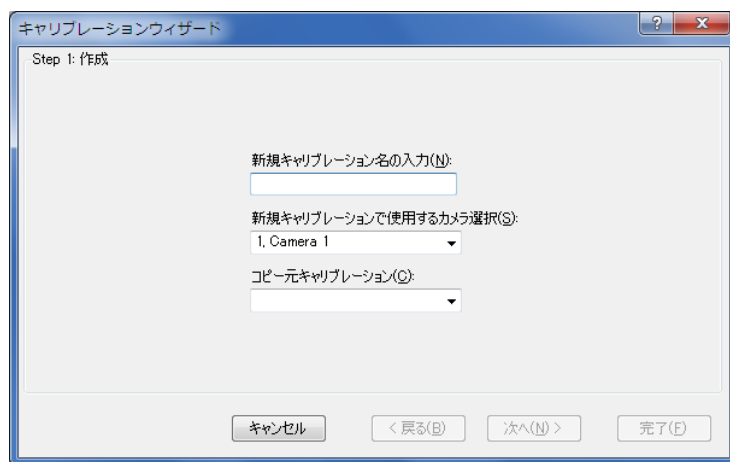
(1) 以下を参照し、ビジョンシーケンスを作成します。

### 7.4.1 カメラポイント9点ティーチに必要なビジョンシーケンス

準備が完了したら、キャリブレーションウィザードを開始して、必要な設定を行います。

#### Step1: キャリブレーションウィザード開始

Vision Guideツールバー -  <キャリブレーション>ボタンをクリックして、キャリブレーションウィザードを開始します。

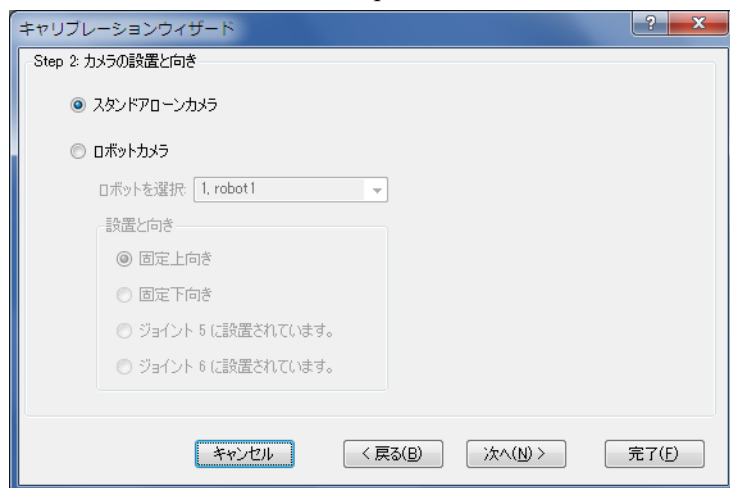


[新規キャリブレーション名の入力]にキャリブレーション名を入力します。

[新規キャリブレーションで使用するカメラ選択]にキャリブレーションの対象とするカメラを選択します。

[コピー元キャリブレーション]にコピー元のキャリブレーションデータを指定すると、設定がコピーされます。

<次へ>ボタンをクリックして、Step2に進みます。





**Step2: キャリブレーションの種類とカメラの向きの設定**

<ロボットカメラ>を選択します。

[ロボットを選択]で、カメラが取り付けられているロボットを選択します。

[設置と向き]で、カメラの取り付け位置を次の中から選択します。

**水平多関節型 (スカラ) ロボットの場合:**

Mobile J2

(可動カメラ:  
ロボット第2軸搭載)

カメラは水平多関節型(スカラ)ロボットの第2アーム、直角座標型ロボットの第2軸(J2)に搭載されています。

Mobile J4

(可動カメラ:  
ロボット第4軸搭載)

カメラは水平多関節型(スカラ)ロボットの第4アーム(J4)、直角座標型ロボットの第4軸(J4)に搭載されています。

**6軸ロボットの場合:**

Mobile J5

(可動カメラ:  
ロボット第5軸搭載)

カメラは垂直6軸型ロボットの第5アーム(J5)に搭載されています。

Mobile J6

(可動カメラ:  
ロボット第6軸搭載)

カメラは垂直6軸型ロボットの第6アーム(J6)に搭載されています。



選択が完了したら、<次へ>ボタンをクリックします。

### Step3: ターゲットシーケンスの指定

ターゲットシーケンスを指定します。

「準備4: キャリブレーション基準ターゲットを検出するためのビジョンシーケンスの作成」で作成したビジョンシーケンスをリストから選択します。

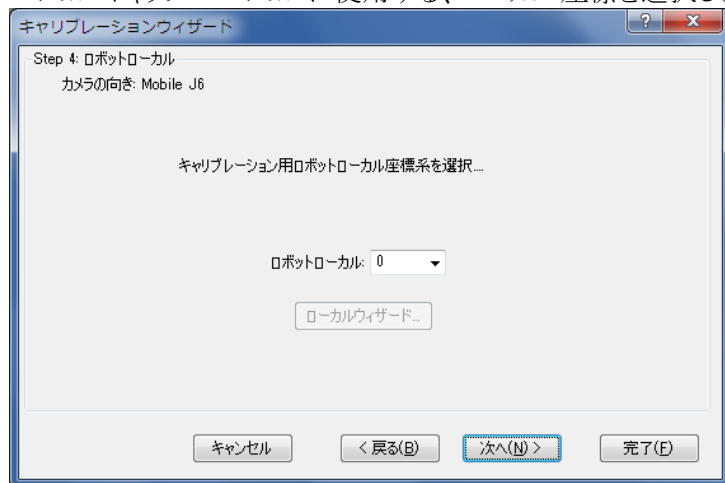
ここでは、Step1で指定されているカメラのビジョンシーケンスのみリストに表示されています。



選択が完了したら、<次へ>ボタンをクリックします。

### Step4: ローカルの設定

ビジョンキャリブレーションに使用する、ローカル座標を選択します。



このダイアログで、ローカルウィザードを開始して、ローカル番号にローカル座標系を定義することもできます。

ウィザードを開始するには、設定する“0”以外のローカル番号を選択して、<ローカルウィザード>ボタンをクリックします。

カメラを使用したローカル設定の方法については、「7.7 カメラを使用したローカル設定」で説明しています。

選択が完了したら、<次へ>ボタンをクリックします。

## Step5: 基準点のタイプの設定

キャリブレーションで使用される基準点のタイプを設定します。



キャリブレーションで使用される基準点は、ロボット座標で指定されたポイントです。

<ティーチポイント>を選択した場合:

**Auto Reference**にチェックがついている場合、基準点を自動的に見つけられます。チェックがついていない場合は、ロボットをジョグ動作させ、手動で基準点をティーチングする必要があります。

ロボット先端に取りつけられたハンドなどの基準点となるツール番号と、アーム番号を指定する必要があります。(増設アーム設定は、スカロボットのみ設定可能です。)

第2アーム(J2), 第4アーム(J4), 第6アーム(J6)のいずれかに設置されているカメラは、[Auto Reference]をチェックすることで、ツールやアームの指定を省略することもできます。省略した場合は、キャリブレーション実行中に自動的にアーム、またはツールの計算を実行し、基準点を自動的に指定します。

このステップで、“0”以外のツール番号を指定して<ツールウィザード>ボタンをクリックすると、ツールウィザードを開始してツール定義をすることもできます。

また、“0”以外のアーム番号を指定して<アームウィザード>ボタンをクリックすると、アームウィザードを開始してカメラのアーム定義をすることができます。

カメラのツールやアーム定義の方法

参照: 7.8 可動カメラ設置位置の検出

**ヒント** : **Auto Reference**機能により基準点のティーチを省略してキャリブレーションを実行したときで、必要なキャリブレーション精度が得られない場合は、**Auto Reference**機能を使用しないでキャリブレーションを実行すると、精度が向上する場合があります。

<上向きカメラ>を選択した場合:

すでにキャリブレーションされた上向き固定カメラを使って基準点を精度よく検出できます。

基準点のタイプとして<上向きカメラ>を選択した場合は、[上向きターゲットシーケンス]で、さらに検出に使うビジョンシーケンスを選択します。

リストには、選択されているカメラが指定されたビジョンシーケンスのみ表示されています。

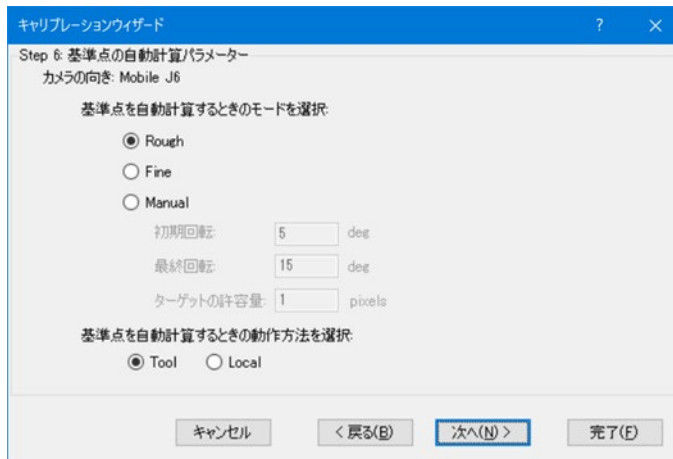
[2点の基準]をチェックした場合:

キャリブレーション中に、Uを180°回転させた2点目の基準点を使用するように設定されます。ツールを使用している場合は、省略できます。

選択や設定が完了したら、<次へ>ボタンをクリックします。

### Step6: 基準点の自動計算パラメーターの設定

基準点を自動で見つけるときのパラメーターを設定します。Step5で[Auto Reference]をチェックすると、このステップが表示されます。



基準点を自動で見つけるときのモードを設定します。カメラの取り付け位置により、ロボットの動作が異なります。

<Rough>を選択した場合:

ロボットは小さく動き、粗い位置決めができます。  
MobileJ2カメラの場合、アームが小さく動きます。  
MobileJ4, J6カメラの場合、ツールが小さく回転します。

<Fine>を選択した場合:

ロボットは大きく動き、精度よく位置決めできます。  
MobileJ2カメラの場合、アームが左右の姿勢変更をともなって動きます。  
MobileJ4, J6カメラの場合、ツールが大きく回転します。

<Manual>を選択した場合

ロボットの動作角度やターゲットの許容量を手動で入力できるようになります。  
<Rough>を選択してもロボットの動きが大きいと感じる場合には、<Manual>を選択して小さな動きに変えることができます。アームは左右の姿勢変更を伴いません。

MobileJ6カメラの場合、基準点を自動で見つけるときの動作方法も設定できます。

<Tool>を選択した場合:

ロボットはツール0座標系のXY平面で動作します。カメラは、光軸がツール0座標系のZ軸方向(第6 関節フランジ面に対して垂直方向)に対して概ね平行となるように取り付ける必要があります。

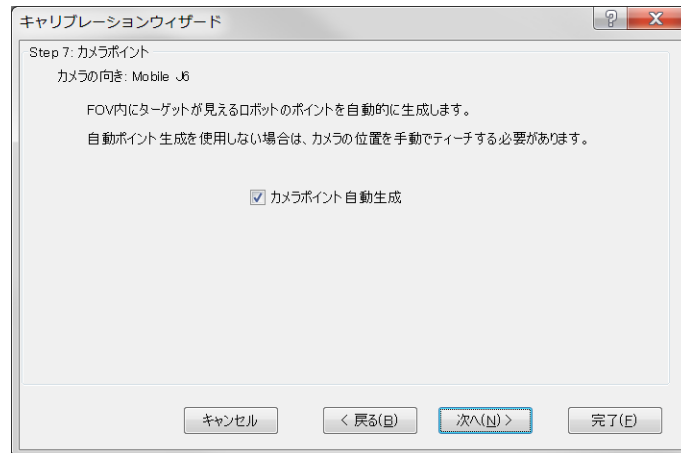
<Local>を選択した場合:

ロボットはSptep4で指定した ローカル座標系のXY平面で動作します。Toolの場合と異なり、カメラは自由な角度で取り付けられます。ただし、ローカル座標系のXY平面がカメラの撮像面と概ね平行となるように、ローカル座標系を指定する必要があります。

設定が完了したら、<次へ>ボタンをクリックします。

### Step7: カメラポイントの設定

キャリブレーション実行時に、カメラポイントを自動で生成するかどうかを指定します。



[カメラポイント自動生成]をチェックした場合:

カメラ視野内でターゲットを検出しながら自動的にロボットを動かして、複数のカメラポイントを自動的に生成します。それぞれのカメラポイントでは、視野内でのターゲットの位置を検出します。自動生成の場合は、カメラ視野の中央付近にターゲットを配置した1点をポイントティーチする必要があります。

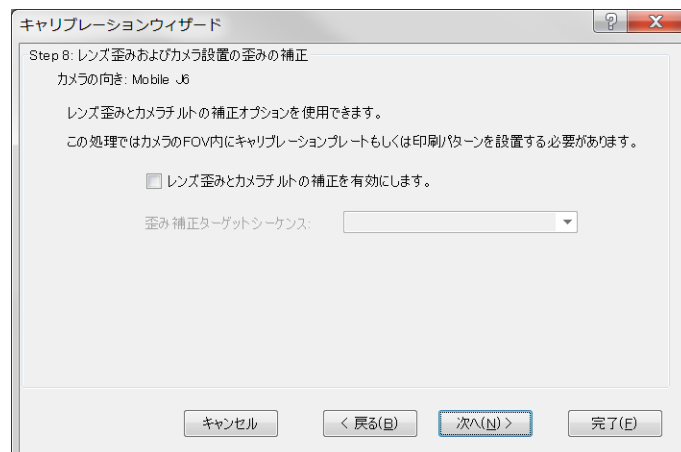
[カメラポイント自動生成]のチェックをはずした場合:

キャリブレーション実行前に手動でロボットをジョグ動作させ、必要な数のカメラポイントをティーチングする必要があります。

設定が完了したら、<次へ>ボタンをクリックします。

### Step8: レンズ歪みとカメラの設置歪みの補正の設定

レンズ歪みと、カメラの設置歪みの補正を行うかを指定します。



チェックボックスをチェックすると、補正が有効になります。補正するためには、あらかじめ歪み補正用のターゲットシーケンスを作成しておき、ここで指定する必要があります。

設定が完了したら、<次へ>ボタンをクリックします。

### Step9: 照明コントロールの設定

キャリブレーション時に使用する照明のコントロールを設定します。照明のコントロールが不要な場合は、設定を変更する必要はありません。



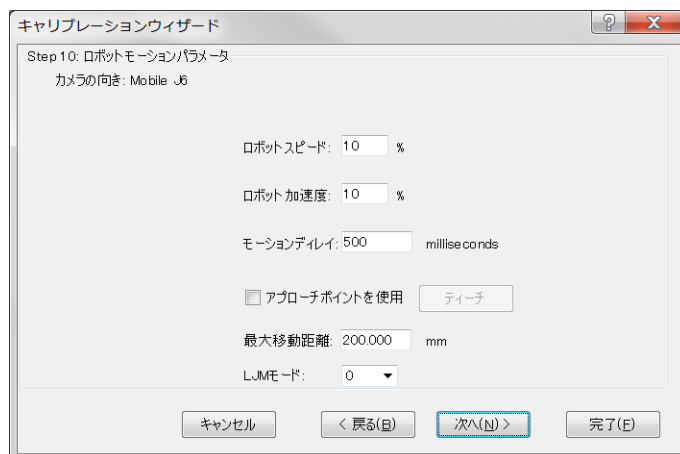
カメラのため照明を使用する場合、照明がオンするまでの待機時間をミリ秒で設定します。さらに、照明をオンする出力ビットを設定します。

Step5で基準点のタイプとして<上向きカメラ>を選択した場合は、上向きカメラの照明をオンする出力ビットも設定できます。

設定が完了したら、<次へ>ボタンをクリックします。

### Step10: ロボットの動作の設定

ロボットの動作に関する設定を行います。



ロボットの速度と加速度、およびロボットが動作した後の静定時間 (撮像までの待ち時間) を設定します。より精密なキャリブレーションを行うためには、遅い速度と加速度に設定し、静定時間を十分に取るようにします。

アプローチポイントも設定できます。

アプローチポイントを設定すると、キャリブレーションポイントへは、毎回、指定されたアプローチポイントから移動します。キャリブレーションポイントへは、常に一定の方向からアプローチするようになり、ロボットの位置が安定します。

アプローチポイントの設定方法:

[アプローチポイントを使用]をチェックし、<ティーチ>ボタンをクリックします。  
表示されるポイントティーチ画面でアプローチポイントをティーチングします。

設定が完了したら、<次へ>ボタンをクリックします。

### Step11: 設定の確認

設定された項目が表示されます。設定を確認します。



<完了>ボタンをクリックしてウィザードを終了します。

### 歪み補正の検出 (歪み補正を有効にした場合のみ)

- (1) シーケンス、またはキャリブレーションツリーで、作成したキャリブレーションを選択します。
- (2) 正方格子パターンを作業面に配置します。
- (3) プロパティリストで、DistCorrect プロパティの下にある Cal プロパティを選択し、正方格子パターンの検出を実行します。

Calibrationプロパティ(キャリブレーションターゲットを位置決めするためのシーケンス)に、作成したキャリブレーションスキームを選択すると、レンズ歪み、およびカメラチルトが補正された画像を確認することができます。この設定を省略しても、キャリブレーション実行時に、自動的に歪みが補正されます。

### ポイントの教示

- (1) <ポイントティーチング>ボタンをクリックします。  
[キャリブレーションポイントのティーチング]ダイアログが表示されます。
- (2) ダイアログ下方のメッセージボックスに表示されるメッセージの指示にしたがいます。  
カメラの設置, 基準点のタイプ, その他の設定によって、必要なカメラポイントと基準点のティーチの内容が異なります。
- (3) カメラポイントを教示します。  
カメラポイントの自動生成を有効にした場合は、1個のカメラ位置を教示します。  
カメラポイントの自動生成を無効にした場合は、9個のカメラ位置を教示します。  
これらのカメラ位置は、9個のロボット位置を表します。1番目の位置は画像イメージ表示部の左上コーナーに、2番目の位置は画像イメージ表示部の中央上に見えるように教示します。以下順次、同様に教示します。よりよい結果を得るために、これらのポイントは、カメラの視野全体に広げてください。  
  
垂直6軸型ロボット使用の場合、通常、カメラポイントのV座標は0になり、W座標はローカルの方向により、0または180になります。キャリブレーションポイントを教示するとき、V座標とW座標を変更する必要はありません。これは、カメラの位置が各カメラポイントのローカルと相対的に一直線上にあるようにするためです。
- (4) 基準点タイプにティーチポイントを指定した場合:  
基準点を教示します。  
パーツのピックアップに用いられるツールがパーツの真上の位置に来るように、ロボットを適切にティーチングします。(ロボットの動作先としてティーチングするのと同じことです。)

### キャリブレーション実行

<キャリブレーション実行>ボタンをクリックして、キャリブレーションサイクルを開始します。

ロボットは、各カメラ位置に動作し、キャリブレーションターゲットビジョンシーケンスを実行します。

9個の位置全部へ移動した後、キャリブレーションパラメーターを確定し、統計データを集めるために、サイクルを繰り返します。

<中止>ボタンをクリックすると、キャリブレーションを停止できます。



### 7.6.2 キャリブレーション手順: 下向き固定カメラ

ロボット作業領域に対して下向きに向けられた固定カメラでオブジェクトを検索し、その位置をロボット座標系に取り込みます。ここでは、作業平面にキャリブレーションプレートを設置する方法を説明します。エンドエフェクターに取りつけられたツール上のターゲットを基準点とする方法は、「7.6.3 キャリブレーション手順: 上向き固定カメラ」に記載しています。

#### 準備 1: カメラの取り付け

- (1) ロボット作業領域に対して下向きになるように、カメラを取りつけます。  
ロボットがカメラと接触しないことを確認してください。

#### 準備2: 歪み補正の準備 (歪み補正を行う場合のみ)

- (1) 歪み検出用のパターンを作成します。  
参照: 7.2 レンズの歪み、およびカメラチルト補正  
オプションのキャリブレーションプレートも使用できます。
- (2) ビジョンシーケンスを作成します。  
参照: 7.4.3 歪み補正に必要なビジョンシーケンス

#### 準備 3: キャリブレーションプレートの作成


- (1) カメラ視野全体に広がる9個の穴、またはターゲットを備えたプレートを作成します。

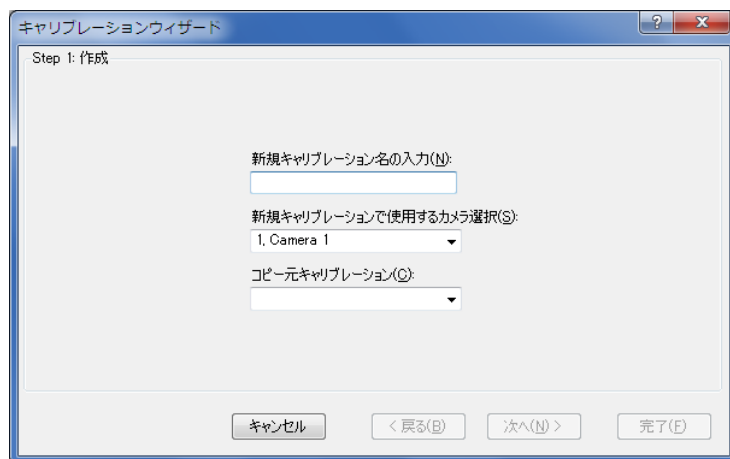
#### 準備4: キャリブレーション基準ターゲットを検出するためのビジョンシーケンスの作成

- (1) ビジョンシーケンスを作成します。  
参照: 7.4.1 ターゲット1個を検出するビジョンシーケンス

準備が完了したら、キャリブレーションウィザードを開始して、必要な設定を行います。

### Step1: キャリブレーションウィザード開始

Vision Guideツールバー -  <キャリブレーション>ボタンをクリックして、キャリブレーションウィザードを開始します。



[新規キャリブレーション名の入力]にキャリブレーション名を入力します。

[新規キャリブレーションで使用するカメラ選択]にキャリブレーションの対象とするカメラを選択します。

[コピー元キャリブレーション]にコピー元のキャリブレーションデータを指定すると、設定がコピーされます。

<次へ>ボタンをクリックして、Step2に進みます。

### Step2: キャリブレーションの種類とカメラの向きの設定

<ロボットカメラ>を選択します。

[ロボットを選択]で、カメラが取り付けられているロボットを選択します。

[設置と向き]で、<固定下向き>を選択します。



選択が完了したら、<次へ>ボタンをクリックします。

**Step3: ターゲットシーケンスの指定**

ターゲットシーケンスを指定します。

「準備4: キャリブレーション基準ターゲットを検出するためのビジョンシーケンスの作成」で作成したビジョンシーケンスをリストから選択します。

ここでは、Step1で指定されているカメラのビジョンシーケンスのみリストに表示されています。



選択が完了したら、<次へ>ボタンをクリックします。

**Step4: ローカルの設定**

ビジョンキャリブレーションに使用する、ローカル座標を選択します。



このダイアログで、ローカルウィザードを開始して、ローカル番号にローカル座標系を定義することもできます。

ウィザードを開始するには、設定する“0”以外のローカル番号を選択して、<ローカルウィザード>ボタンをクリックします。

カメラを使用したローカル設定の方法については、「7.7 カメラを使用したローカル設定」で説明しています。

選択が完了したら、<次へ>ボタンをクリックします。

### Step5: 基準点のタイプの設定

キャリブレーションで使用する基準点のタイプを設定します。



キャリブレーションで使用する基準点は、ロボット座標で指定されたポイントです。

<ティーチポイント>を選択します。

ロボットをジョグ動作させ、手動で基準点をティーチする必要があります。

ロボット先端に取りつけられたハンドなどの基準点となるツール番号と、アーム番号を指定する必要があります。(増設アーム設定は、スカラロボットのみ設定可能です。)

このステップで、“0”以外のツール番号を指定して<ツールウィザード>ボタンをクリックすると、ツールウィザードを開始してツール定義をすることもできます。

また、“0”以外のアーム番号を指定して<アームウィザード>ボタンをクリックすると、アームウィザードを開始してカメラのアーム定義をすることができます。

カメラのツールやアーム定義の方法

参照: 7.8 可動カメラ設置位置の検出

<EndEffector>を指定する場合:

参照: 7.6.3 キャリブレーション手順: 上向き固定カメラ

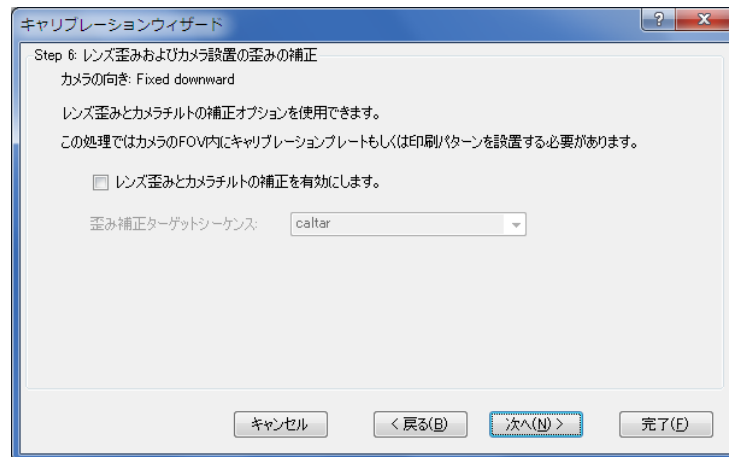
[2点の基準]をチェックした場合:

キャリブレーションで2つの基準点を使用するように設定されます。

選択や設定が完了したら、<次へ>ボタンをクリックします。

## Step6: レンズ歪みとカメラの設置歪みの補正の設定

レンズ歪みとカメラの設置歪みの補正を行うかを指定します。



チェックボックスをチェックすると、補正が有効になります。補正するためには、あらかじめ歪み補正用のターゲットシーケンスを作成しておき、ここで指定する必要があります。

設定が完了したら、<次へ>ボタンをクリックします。

## Step7: 照明コントロールの設定

キャリブレーション時に使用する照明のコントロールを設定します。照明のコントロールが不要な場合は、設定を変更する必要はありません。



照明のコントロールをする場合、照明がオンするまでの待機時間をseconds(秒)で設定し、さらに、照明をオンする出力ビットを設定します。

設定が完了したら、<次へ>ボタンをクリックします。

### Step8: ロボットの動作の設定

ロボットの動作に関する設定を行います。



ロボットの速度と加速度、およびロボットが動作した後の静定時間(モーションディレイ)を設定します。より精密なキャリブレーションを行うためには、遅い速度と加速度に設定し、静定時間を十分に取るようにします。

アプローチポイントも設定できます。

アプローチポイントを設定すると、カメラポイントへは、毎回、指定されたアプローチポイントから移動します。カメラポイントへは常に一定の方向からアプローチするようになり、ロボットの位置が安定します。

アプローチポイントの設定方法:

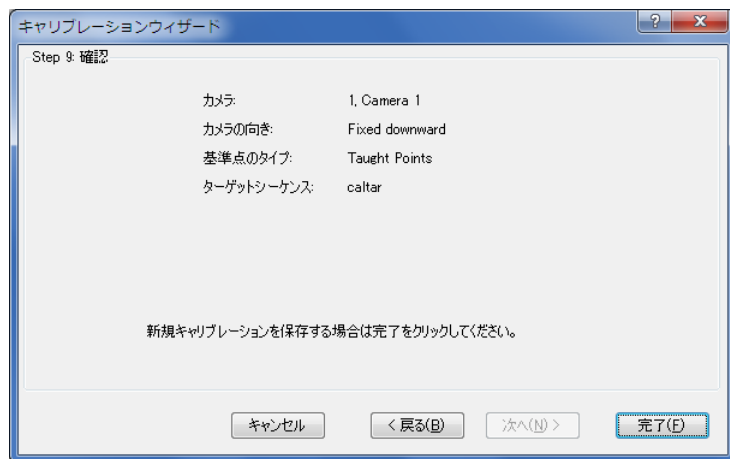
[アプローチポイントを使用]をチェックし、<ティーチ>ボタンをクリックします。

表示されるポイントティーチ画面でアプローチポイントをティーチングしてください。

設定が完了したら、<次へ>ボタンをクリックします。

### Step9: 設定の確認

設定された項目が表示されます。設定を確認します。



<完了>ボタンをクリックしてウィザードを終了します。

**歪み補正の検出 (歪み補正を有効にした場合のみ)**

- (1) シーケンス、またはキャリブレーションツリーで、作成したキャリブレーションを選択します。
- (2) 正方格子パターンを作業面に配置します。
- (3) プロパティリストで、DistCorrect プロパティの下にある Cal プロパティを選択し、正方格子パターンの検出を実行します。

Calibrationプロパティ(キャリブレーションターゲットを位置決めするためのシーケンス)に、作成したキャリブレーションスキームを選択すると、レンズ歪み、およびカメラチルトが補正された画像を確認することができます。この設定を省略しても、キャリブレーション実行時に、自動的に歪みが補正されます。

**ポイントの教示**

- (1) <ポイントティーチング>ボタンをクリックします。  
[キャリブレーションポイントのティーチング]画面が表示されます。
- (2) ダイアログ下方のメッセージボックスに表示されるメッセージの指示にしたがい、各基準点を教示します。  
ポイントを教示し、TwoRefPointsが、“True”に設定されている場合は、第4軸を180°回転してポイントを再教示するように指示します。  
ツールを使用している場合は、この180°回転による再教示の手順は省略することができます。  
この手順を省略するには、<次へ>ボタンをクリックし、次のステップに移ってください。

**キャリブレーション実行**

<キャリブレーション実行>ボタンをクリックして、キャリブレーションサイクルを開始します。

キャリブレーションソフトウェアは、9個のターゲットを位置決めした後、もう1度位置決めして、キャリブレーションパラメーターを確定し、統計データを集めます。

<中止>ボタンをクリックすると、キャリブレーションを停止できます。

### 7.6.3 キャリブレーション手順: 上向き固定カメラ

ロボット作業領域に対して上向きに向けられた固定カメラでオブジェクトを検索し、その位置をロボット座標系に取り込みます。

下向き固定カメラでも、基準点タイプをエンドエフェクター(EndEffector)に設定することにより、同様のキャリブレーションが可能です。

#### 準備 1: カメラの取り付け

- (1) ロボット作業領域に対して上向きになるようにカメラを取りつけます。


#### 準備 2: ハンド上のキャリブレーションターゲットを検出するためのビジョンシーケンスの作成

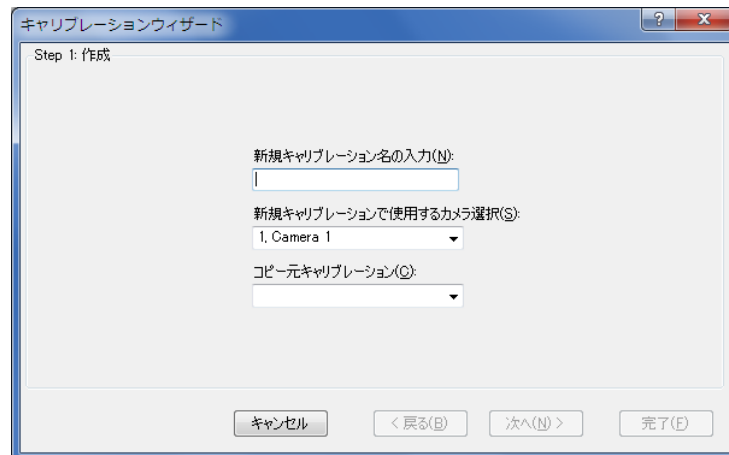
- (1) ハンド上のキャリブレーションターゲットを位置決めするためのシーケンスを作成します。  
キャリブレーションソフトウェアでは、ターゲットの位置を得るのに、シーケンス中の最後のステップ実行を使用します。最後のステップ実行によるXとYの結果がターゲットの中心になります。
- (2) キャリブレーションの間、ハンドはカメラ視野内の9個の異なるポイントに移動して、ターゲットを検索します。また、それぞれの位置で第4軸を180°回転し、ターゲットを再検索します。この再検索によって、各ポイントに第4軸の中心を決定することができます。最高の結果を得るためには、ターゲットは丸型のものを使用してください。

準備が完了したら、キャリブレーションウィザードを開始して、必要な設定を行います。



## Step1: キャリブレーションウィザード開始

Vision Guideツールバー -  <キャリブレーション>ボタンをクリックして、キャリブレーションウィザードを開始します。



[新規キャリブレーション名の入力]にキャリブレーション名を入力します。

[新規キャリブレーションで使用するカメラ選択]にキャリブレーションの対象とするカメラを選択します。

[コピー元キャリブレーション]にコピー元のキャリブレーションデータを指定すると、設定がコピーされます。

<次へ>ボタンをクリックして、Step2に進みます。

## Step2: キャリブレーションの種類とカメラの向きの設定

<ロボットカメラ>を選択します。

[ロボットを選択]で、カメラが取り付けられているロボットを選択します。

[設置と向き]で、<固定上向き>、または<固定下向き>を選択します。



選択が完了したら、<次へ>ボタンをクリックします。

### Step3: ターゲットシーケンスの指定

ターゲットシーケンスを指定します。

「準備2: ハンド上のキャリブレーションターゲットを検出するためのビジョンシーケンスの作成」で作成したビジョンシーケンスをリストから選択します。

ここでは、Step1で指定されているカメラのビジョンシーケンスのみリストに表示されています。



選択が完了したら、<次へ>ボタンをクリックします。

### Step4: ローカルの設定

ビジョンキャリブレーションに使用する、ローカル座標を選択します。



このダイアログで、ローカルウィザードを開始して、ローカル番号にローカル座標系を定義することもできます。

ウィザードを開始するには、設定する“0”以外のローカル番号を選択して、<ローカルウィザード>ボタンをクリックします。

カメラを使用したローカル設定の方法については、「7.7 カメラを使用したローカル設定」で説明しています。

選択が完了したら、<次へ>ボタンをクリックします。

## Step5: 基準点のタイプの設定

キャリブレーションで使用される基準点のタイプを設定します。



上向き固定カメラの基準点タイプは、<EndEffector>です。

ロボットのエンドエフェクターに取りつけられたツール上にあるターゲットを基準点とします。ターゲットは、カメラから見える必要があります。

(下向き固定カメラの場合は、このステップで<基準点タイプ>を選択できますが、ここでは<EndEffector>を選択してください。)

このステップで、“0”以外のツール番号を指定して<ツールウィザード>ボタンをクリックすると、ツールウィザードを開始してツール定義をすることもできます。

また、“0”以外のアーム番号を指定して<アームウィザード>ボタンをクリックすると、アームウィザードを開始してカメラのアーム定義をすることができます。

(増設アーム設定は、スカラロボットのみ設定可能です。)

カメラのツールやアーム定義の方法

参照: 7.8 可動カメラ設置位置の検出

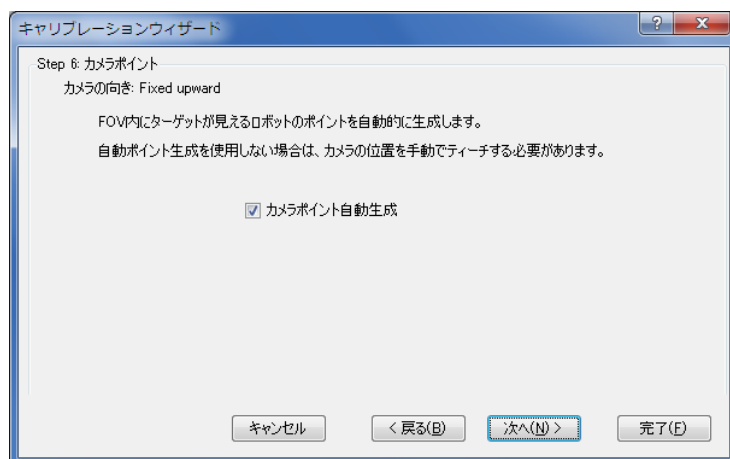
[2点の基準]をチェックした場合:

キャリブレーションで2つの基準点を使用するように設定されます。

選択や設定が完了したら、<次へ>ボタンをクリックします。

### Step6: カメラポイントの設定

キャリブレーション実行時に、カメラポイントを自動で生成するかどうかを指定します。



チェックボックスをチェックした場合:

カメラ視野内でターゲットを検出しながら自動的にロボットを動かして、複数のカメラポイントを自動的に生成します。

それぞれのカメラポイントでは、視野内でのターゲットの位置を検出します。この方法では、カメラ視野の中央付近にターゲットを配置してポイントティーチします。ロボットをジョグ動作させ、他のカメラポイントをティーチングする必要はありません。

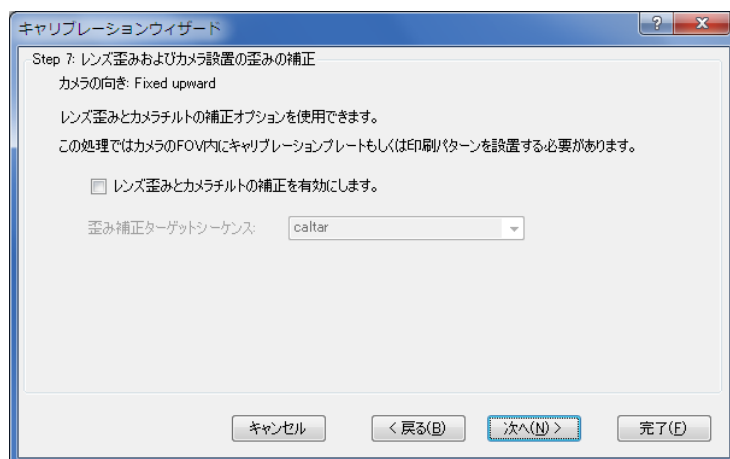
チェックボックスのチェックをはずした場合:

キャリブレーション実行前に手動でロボットをジョグして、必要な数のカメラポイントをティーチする必要があります。

設定が完了したら、<次へ>ボタンをクリックします。

### Step7: レンズ歪みとカメラの設置歪みの補正の設定

レンズ歪みと、カメラの設置歪みの補正を行うかを指定します。



チェックボックスをチェックすると、補正が有効になります。補正するためには、あらかじめ歪み補正用のターゲットシーケンスを作成しておき、ここで指定する必要があります。

設定が完了したら、<次へ>ボタンをクリックします。

## Step8: 照明コントロールの設定

キャリブレーション時に使用する照明のコントロールを設定します。照明のコントロールが不要な場合は、設定を変更する必要はありません。



照明のコントロールをする場合、照明がオンするまでの待機時間をseconds(秒)で設定し、さらに、照明をオンする出力ビットを設定します。

設定が完了したら、<次へ>ボタンをクリックします。

## Step9: ロボットの動作の設定

ロボットの動作に関する設定を行います。



ロボットの速度と加速度、およびロボットが動作した後の静定時間(モーションディレイ)を設定します。より精密なキャリブレーションを行うためには、遅い速度と加速度に設定し、静定時間を十分に取るようにします。

アプローチポイントの設定もできます。

アプローチポイントを設定すると、キャリブレーションポイントへは、毎回、指定されたアプローチポイントから移動します。キャリブレーションポイントへは常に一定の方向からアプローチするようになり、ロボットの位置が安定します。

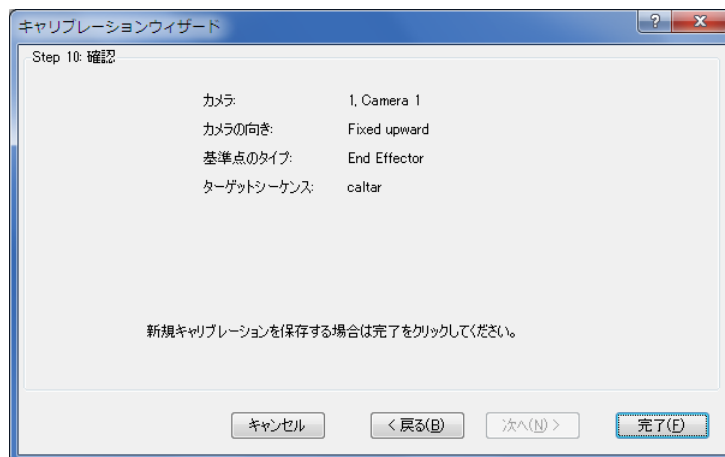
アプローチポイントの設定方法:

[アプローチポイントを使用]をチェックし、<ティーチ>ボタンをクリックします。  
表示されるポイントティーチ画面でアプローチポイントをティーチングします。

設定が完了したら、<次へ>ボタンをクリックします。

### Step10: 設定の確認

設定された項目が表示されます。設定を確認します。



<完了>ボタンをクリックしてウィザードを終了します。

### ポイントの教示

- (1) <ポイントティーチング>ボタンをクリックします。  
[キャリブレーションポイントのティーチング]ダイアログが表示されます。
- (2) ダイアログ下方のメッセージボックスに表示されるメッセージの指示にしたがい、各基準点を教示します。  
表示メッセージは、ポイントを教示し、第4軸を180°回転してポイントを再教示するように指示します。  
ツールを使用している場合は、この180°回転による再教示の手順は省略することができます。この手順を省略するには、<次へ>ボタンを押して、次のステップに移ってください。

### キャリブレーション実行

<キャリブレーション実行>ボタンをクリックして、キャリブレーションサイクルを開始します。

キャリブレーションソフトウェアは、それぞれのカメラ位置にロボットを動作させ、ターゲットを検索し、TwoRefPointsが“True”に設定されている場合は、第4軸を180°回転してそのターゲットを再検索します。キャリブレーションを繰り返し、統計データを集めます。

<中止>ボタンをクリックすると、キャリブレーションを停止できます。

### 7.6.4 キャリブレーション手順: スタンドアローンカメラ

このキャリブレーションでは、物理的な測定をすることができます。

スタンドアローンカメラとしてキャリブレーションされたカメラは、ロボット座標系の計算に使用することができません。mm単位のCameraX座標位置とCameraY座標位置を返すことができます。

#### Step 1: カメラの取り付け

- (1) 作業面に対して45~90°になるようにカメラを取り付けます。

#### Step 2: 正方格子パターンの作成 (歪み補正を行う場合のみ)

- (1) 100点以上のポイントを持つ正方格子状パターンを作成します。  
格子パターンは、カメラ視野の全体に広がるサイズで作成し、できるだけ歪みを抑えてください。格子パターンの精度が画像処理精度に影響します。

#### Step3: キャリブレーションプレートの作成

- (1) カメラ視野全体に広がる9個の穴、またはターゲットを備えたキャリブレーションプレートを作成します。


#### Step 4: 正方格子パターンを検出するためのビジョンシーケンスの作成 (歪み補正を行う場合のみ)

- (1) ビジョンシーケンスを作成します。  
参照: 7.4.3 歪み補正に必要なビジョンシーケンス

#### Step 5: キャリブレーション基準ターゲットを検出するためのビジョンシーケンスの作成

- (1) ビジョンシーケンスを作成します。  
参照: 7.4.2 ターゲット9個を検出するビジョンシーケンス

#### Step 6: キャリブレーションスキームの作成

- (1) Vision Guideツールバー -  <キャリブレーション>ボタンをクリックして、キャリブレーションを選択します。
- (2) キャリブレーションウィザードが開始します。  
キャリブレーション名とカメラを選択します。必要に応じてコピー元キャリブレーションを選択します。
- (3) スタンドアローンカメラを選択します。
- (4) キャリブレーション基準ターゲットを検出するためのビジョンシーケンスを設定します。
- (5) 歪み補正を有効にするかを設定します。  
歪み補正を有効にした場合、正方格子パターンを検出するためのビジョンシーケンスを選択します。
- (6) 残りのステップを必要に応じて設定し、キャリブレーションウィザードを完了させます。

### Step 7: レンズ歪み、およびカメラチルト補正の設定 (歪み補正を行う場合のみ)

- (1) シーケンス、またはキャリブレーションツリーで、作成したキャリブレーションを選択します。
- (2) 正方格子パターンを作業面に配置します。
- (3) プロパティリストで、DistCorrect プロパティの下にある Cal プロパティを選択し、正方格子パターンの検出を実行します。
- (4) Calibration プロパティ(キャリブレーションターゲットを位置決めするためのシーケンス)に、Step 6 で作成したキャリブレーションスキームを選択すると、レンズ歪み、およびカメラチルトが補正された画像を確認することができます。この設定を省略しても、キャリブレーション実行時に、自動的に歪みが補正されます。

### Step 8: キャリブレーション

- (1) 正方格子パターンを取りのぞき、Step3で作成したキャリブレーションプレートを設置します。
- (2) <ポイントティーチング>ボタンをクリックして、キャリブレーションプレートの中の9個のターゲットの座標を設定します。
- (3) <キャリブレーション実行>ボタンをクリックして、キャリブレーションサイクルを開始します。キャリブレーションソフトウェアは、9個のターゲットを位置決めします。さらに、もう1度位置決めして、キャリブレーションパラメーターを確定し、統計データを集めます。



## 7.7 カメラを使用したローカル検出

6軸ロボットを使用している場合、第6アーム(J6)に設置されたカメラを使用してローカル検出をすることが可能です。

ロボットマネージャーで[ローカル設定]のタブを選択し、ローカルウィザードを開始します。また、キャリブレーションウィザードの中でも開始できます。

### 7.7.1 作業面のローカル定義

作業面に置いたキャリブレーションプレートを実動カメラで検出して、作業面に平行なローカル座標を定義します。この機能は、6軸ロボットで第6アーム(J6)に設置された可動カメラを使用している場合に有効です。

**Note:** カメラを使用して作業面のローカル設定をするには、オプションのキャリブレーションプレートを使用します。ロボットをジョグして、カメラの光軸とキャリブレーションプレートとがおおよそ90°となるようにしてください。また、カメラのハードウェア構成により、カメラを使用したローカル設定機能が使用できない場合があります。

参照: 7.ビジョンキャリブレーション

カメラを使ったローカル座標の定義の機能は、ローカル設定ウィザードで実行できます。次のどちらかの方法で、ローカル設定ウィザードを開始します。

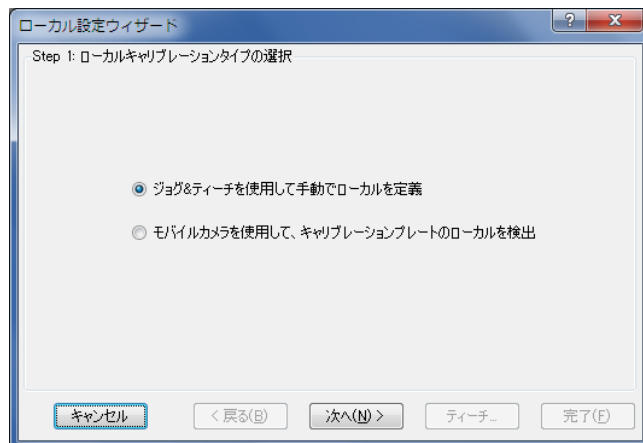
- (1) ロボットマネージャーの[ローカル設定]タブを選択します。  
<ローカル設定ウィザード...>ボタンが表示されます。このボタンをクリックします。
- (2) キャリブレーションウィザードのロボットローカルを選択するステップで、“0”以外のロボットローカル番号を選択すると、<ローカルウィザード...>ボタンが表示されます。このボタンをクリックします。

ローカル設定ウィザード開始後の手順を説明します。(選択項目により、画面内容が変わります。)

### Step1: タイプの選択

ローカル設定ウィザードを開始すると、次のダイアログが表示されます。

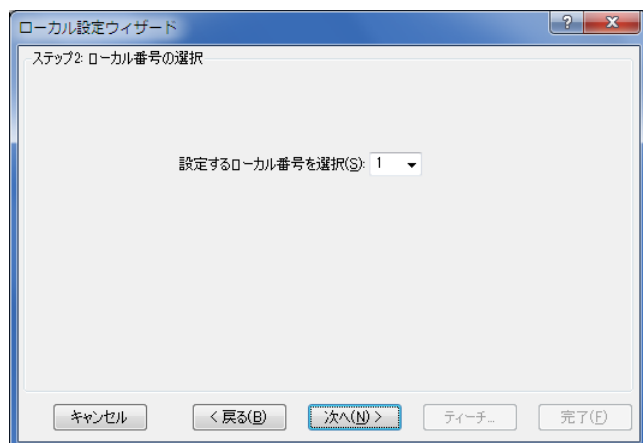
<モバイルカメラを使用して、キャリブレーションプレートのローカルを検出>ボタンを選択します。



<次へ>ボタンをクリックします。

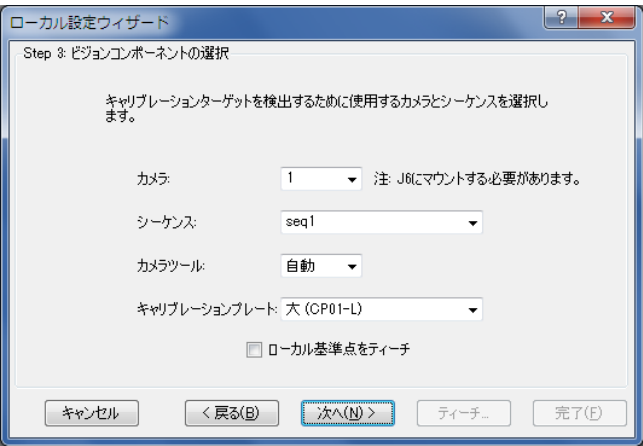
### Step2: ローカル番号の選択

設定するローカル番号を選択します。



<次へ>ボタンをクリックします。

Step3: ビジョンコンポーネントの選択



以下の内容を設定します。

項目	内容
カメラ	キャリブレーションに使用するカメラを選択します。 カメラはJ6(第6アーム)に設置されている必要があります。
シーケンス	キャリブレーションプレートを検出するためのビジョ ンシーケンスを選択します。
カメラツール	キャリブレーション済みの可動カメラのツール番号 を指定します。  “自動”を選択した場合は、カメラツールの検出が 自動的に行われます。
キャリブレーションプレート	使用しているキャリブレーションプレートのタイプを 選択します。
ローカル基準点をティーチ	ローカル基準点をティーチすると、検出したローカ ル平面が、ティーチした点を通るように設定されま す。ティーチしない場合は、ツール0を通るように設 定されます。  この機能を有効にすると、エンドエフェクターがロー カル平面に来るようにロボットをジョグ動作させ、ポ イントをティーチングをする必要があります。

<次へ>ボタンをクリックします。

### Step4: カメラポイントのティーチ

<ティーチ...>ボタンをクリックします。ロボットのジョグ画面が表示されます。



ロボットをジョグ動作し、キャリブレーションプレートの中央をカメラ視野の中央付近に移動させます。

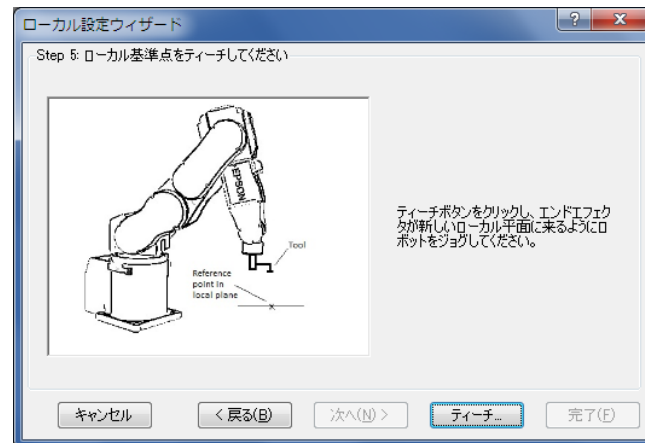


カメラポイントが決まったところで、<ティーチ>ボタンをクリックします。次のステップに進みます。

**Step5: ローカル基準点のティーチ**

ローカル基準点をティーチする場合のみ、このステップが表示されます。

<ティーチ...>ボタンをクリックすると、ロボットのジョグ画面が表示されます。

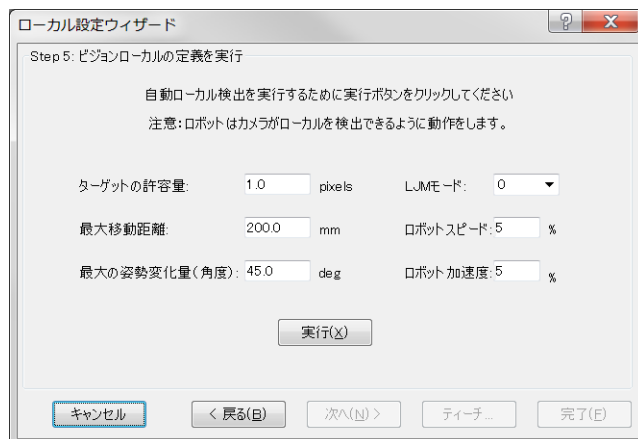


エンドエフェクターがローカル平面に来るようにロボットをジョグ動作させます。

<ティーチ>ボタンをクリックすると、ポイントがティーチングされ、このステップに戻ります。

## Step6: ビジョンローカルの定義を実行

必要に応じて以下の設定を変更してください。



必要に応じて以下の設定を変更します。

項目	内容
ターゲットの許容量	画像上での誤差が、この許容量に収まることを目標に、検出が実行されるようになります。
最大移動距離	アーム先端の移動距離の制限を設定します。 “0”の場合は、制限しません。
最大の姿勢変化量	手先姿勢 (UVW)の最大変位角度 (°)です。 “0”の場合は、制限しません。
LJMモード	SPEL+のLJM関数で使われている値を指定します。 LJMモードは手首部の意図しない回転を防止するためにポイントデータの姿勢フラグを適切に制御します。 “0”の場合は、LJMを使用しません。
ロボットの速度	ロボットの速度を設定します。 精密なローカル設定を行うためには、小さな値を設定します。
ロボットの加速度	ロボットの加速度を設定します。 精密なローカル設定を行うためには、小さな値を設定します。

<実行>ボタンをクリックします。

検出が完了すると、結果が表示されます。結果を確認してください。

<完了>ボタンをクリックすると、指定したローカル番号に結果が設定されます。

**注意:** キャリブレーションプレートの認識やターゲットの検出結果に基づいて、自動的にロボットが動作します。ロボットと周辺装置の干渉に注意してください。またローカル検出中のエラーを回避するため、各関節が伸びる特異点近傍姿勢を避けて使用してください。

## 7.8 可動カメラ設置位置の検出

可動カメラの設置位置を検出できます。

スカラロボットの第2アーム(J2)に設置されたカメラは、増設アーム設定のパラメーターとして設定されます。

スカラロボットの第4軸(J4)、または垂直6軸型ロボットの第6アーム(J6)に設置されたカメラについては、ツールとして設定されます。

### 7.8.1 カメラ設置位置のツール設定

スカラロボットの第4軸(J4)、または垂直6軸型ロボットの第6アーム(J6)に設置されたカメラのツール設定手順を説明します。

あらかじめ、ターゲット検出に必要なビジョンシーケンスを作成します。

参照: 7.4.4 ローカル設定, ツール設定, アーム設定に必要なビジョンシーケンス

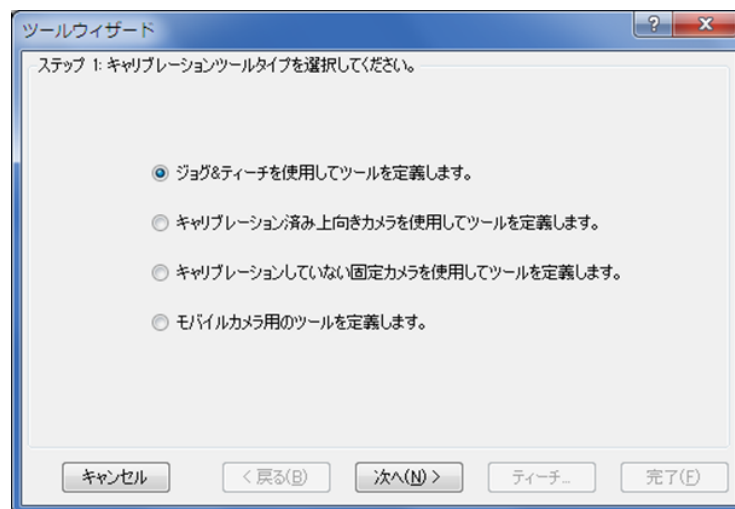
次のどちらかの方法で、ツールウィザードを実行します。

- (1) ロボットマネージャの[ツール設定]タブを選択します。  
<ツール設定ウィザード...>ボタンが表示されます。  
このボタンをクリックします。
- (2) キャリブレーションウィザードの、ツール番号を選択するステップで、“0”以外のツール番号を選択すると、<ツールウィザード...>ボタンが表示されます。  
このボタンをクリックします。

#### Step1: タイプの選択

ツールウィザードを開始すると、次のダイアログが表示されます。

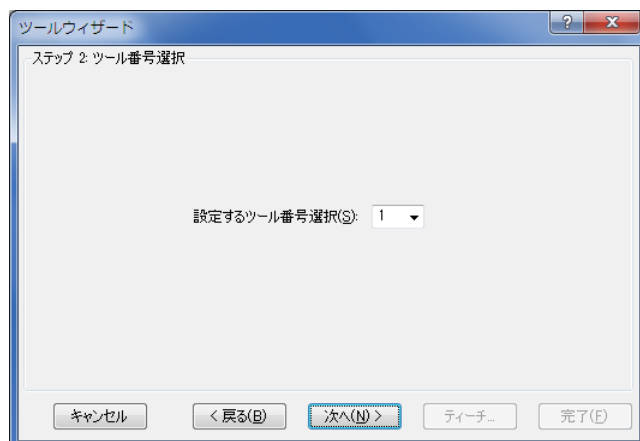
<モバイルカメラ用のツール定義>ボタンを選択します。



<次へ>ボタンをクリックします。

### Step2: ツール番号選択

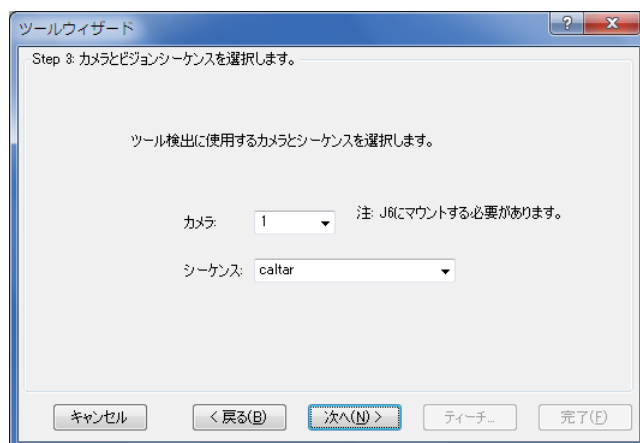
設定するツール番号を選択します。



<次へ>ボタンをクリックします。

### Step3: ビジョン設定

カメラ、およびターゲットを検出するためのシーケンスを選択します。



<次へ>ボタンをクリックします。

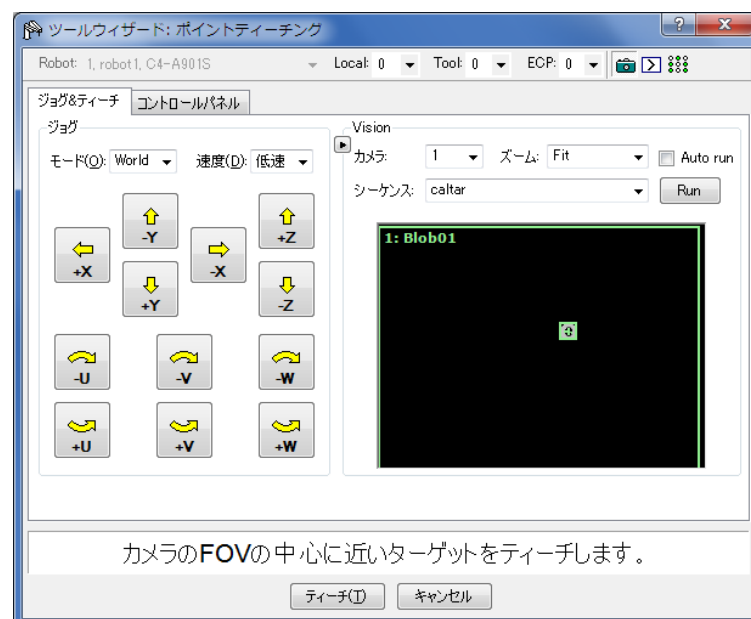


## Step4: カメラポイントのティーチ

<ティーチ...>ボタンをクリックします。ポイントティーチング画面が表示されます。



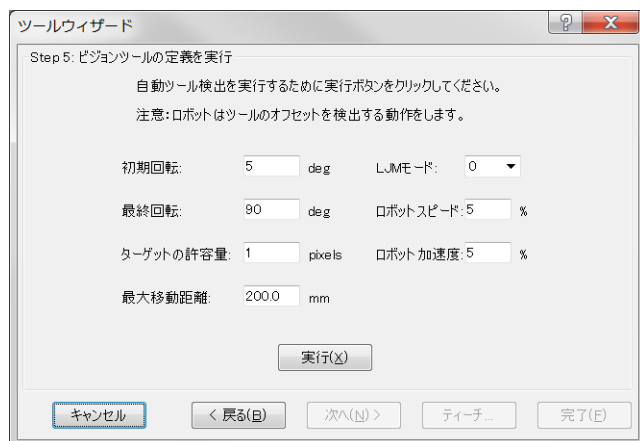
ロボットをジョグ動作し、ターゲットがカメラ視野中央付近に検出されるようにします。



<ティーチ>ボタンをクリックします。

次のステップに進みます。

## Step5: 実行



必要に応じて以下の設定を変更します。

項目	内容
初期回転	粗位置決めのために最初にツールを回転させる角度(°)です。
最終回転	最終的にツールを回転させる角度(°)です。 最終回転角度が大きい(90°以上を推奨)ほど、高精度にツール設定できます。
ターゲットの許容量	画像上での誤差が、この許容量に収まることを目標に、検出が実行されるようになります。
最大移動距離	アーム先端の移動距離の制限を設定します。 “0”の場合は、制限しません。
LJMモード	SPEL+のLJM関数で使われている値を指定します。 LJMモードは手首部の意図しない回転を防止するためにポイントデータの姿勢フラグを適切に制御します。 “0”の場合は、LJMを使用しません。
ロボットスピード	ロボットの速度を設定します。 精密なツール設定を行うためには、小さな値を設定します。
ロボット加速度	ロボットの加速度を設定します。 精密なツール設定を行うためには、小さな値を設定します。

<実行>ボタンをクリックします。

検出が完了すると、結果が表示されます。結果を確認してください。

<完了>ボタンをクリックすると、指定したツール番号に結果が設定されます。

**注意:** ターゲットの検出結果に基づいて、自動的にロボットが動作します。ロボットと周辺装置の干渉に注意してください。またツール設定中のエラーを回避するため、各関節が伸びる特異点近傍姿勢を避けて使用してください。

### 7.8.2 カメラ設置位置のアーム設定

スカラロボットの第2アーム(J2)に設置されたカメラの設置位置を増設アームパラメーターとして設定する手順を説明します。

あらかじめ、ターゲット検出に必要なビジョンシーケンスを作成します。

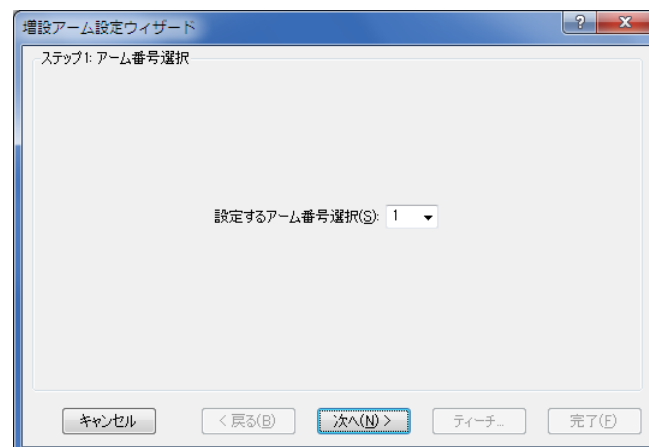
参照: 7.4.4 ローカル設定, ツール設定, アーム設定に必要なビジョンシーケンス

次のどちらかの方法で、アームウィザードを実行します。

- (1) ロボットマネージャーの[増設アーム]タブを選択します。  
<増設アーム設定ウィザード...>ボタンが表示されます。  
このボタンをクリックします。
- (2) キャリブレーションウィザードの、アーム番号を選択するステップにおいて、“0”以外のアーム番号を選択すると、<アームウィザード...> ボタンが表示されます。  
このボタンをクリックします。

#### Step1: アーム番号選択

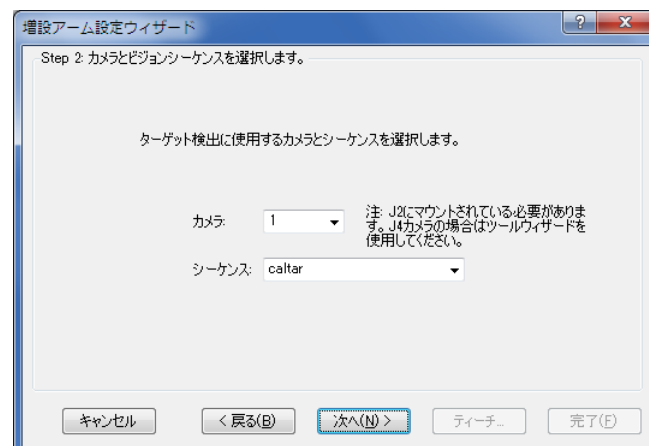
設定する増設アーム番号を選択します。



<次へ>ボタンをクリックします。

#### Step2: ビジョン設定

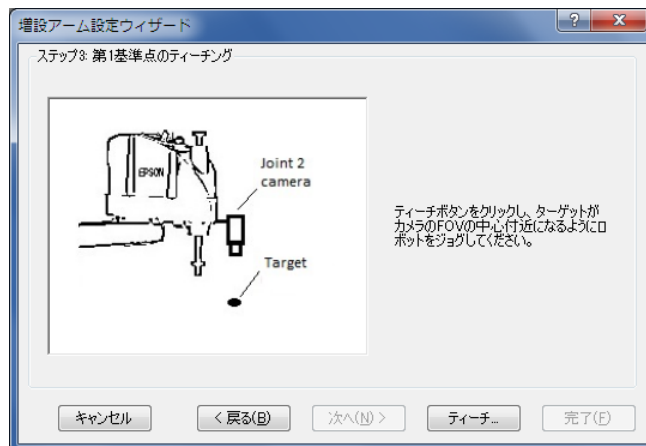
カメラ、およびターゲットを検出するためのシーケンスを選択します。



<次へ>ボタンをクリックします。

### Step3: カメラポイントのティーチ

<ティーチ...>ボタンをクリックします。ポイントティーチング画面が表示されます。



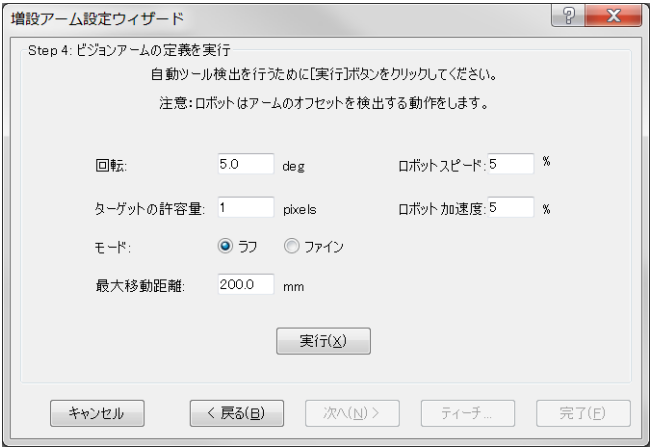
ロボットをジョグ動作し、ターゲットがカメラ視野中央付近に検出されるようにします。



<ティーチ>ボタンをクリックします。

次のステップに進みます。

Step4: 実行



必要に応じて以下の項目を設定します。

項目	内容
回転	粗いアーム設定を行うときの回転角(°)を指定します。
ターゲットの許容量	ビジョン検出結果が対象位置と一致したとみなすピクセル距離を設定します。
モード	ラフ: 粗いアーム設定を実行するモードです。 ロボットは小さく動きます。 精度1mm程度を目指します。 ファイン: 精密にアーム設定を実行するモードです。 ロボットの左右姿勢変更をともない、大きく動きます。 ラフよりも高精度を目指します。
最大移動距離	アーム先端の移動距離の制限を設定します。 “0”の場合は、制限しません。
ロボットスピード	ロボットの速度を設定します。 精密なアーム設定を行うためには、小さな値を設定します。
ロボット加速度	ロボットの加速度を設定します。 精密なアーム設定を行うためには、小さな値を設定します。

<実行>ボタンをクリックします。

検出が完了すると、結果が表示されます。結果を確認してください。

<完了>ボタンをクリックすると、指定し増設アーム番号に結果が設定されます。

**注意:** ターゲットの検出結果に基づいて、自動的にロボットが動作します。ロボットと周辺装置の干渉に注意してください。

### 7.9 カメラを使用したツール設定

固定カメラを使用して、ロボット先端に取りつけられたツールのツール座標を検出できません。

ツール設定ウィザードは、「7.8.1 カメラ設置位置のツール設定」と、ほぼ同じ手順です。

あらかじめ、ターゲット検出に必要なビジョンシーケンスを作成します。

参照: 7.4.4 ローカル設定, ツール設定, アーム設定に必要なビジョンシーケンス

次のどちらかの方法で、ツールウィザードを実行します。

- (1) ロボットマネージャーの[ツール設定]タブを選択します。  
<ツール設定ウィザード...>ボタンが表示されます。  
このボタンをクリックします。
- (2) キャリブレーションウィザードの、ツール番号を選択するステップで、“0”以外のツール番号を選択すると、<ツールウィザード...>ボタンが表示されます。  
このボタンをクリックします。

#### Step1: タイプの選択

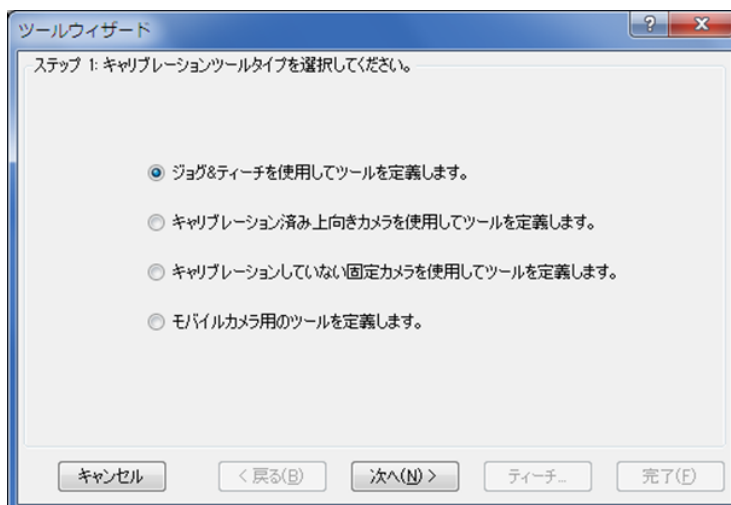
ツールウィザードを開始すると、次のダイアログが表示されます。

次のどちらかのボタンを選択します。

<キャリブレーション済み上向きカメラを使用して定義>

本機能は、シーケンスのキャリブレーションプロパティにキャリブレーション済み上向きカメラが指定されている場合に有効になります。上向きカメラキャリブレーション時に、使用したツールを変更したとき、または異なるツールを設定するときに使用します。

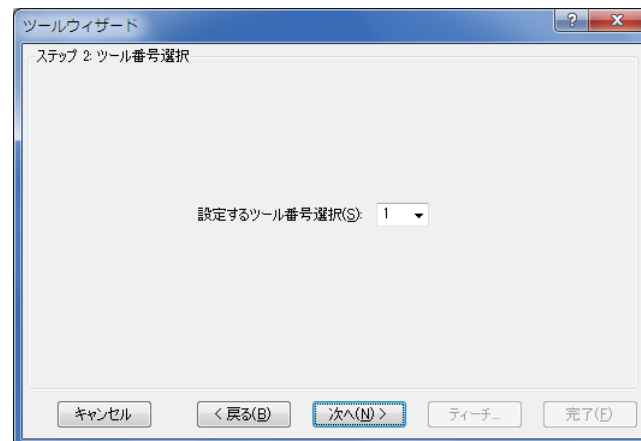
<キャリブレーションしていない固定カメラを使用してツールを定義>



<次へ>ボタンをクリックします。

### Step2: ツール番号選択

設定するツール番号を選択します。



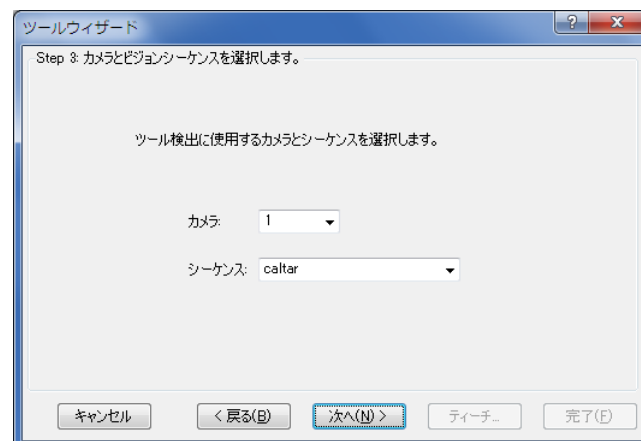
<次へ>ボタンをクリックします。

### Step3: ビジョン設定

カメラ、およびターゲットを検出するためのシーケンスを選択します。

キャリブレーション済み上向きカメラを使用している場合は、オブジェクトも選択します。選択肢には、RobotToolXYUリザルトを持つオブジェクトのみが表示されます。

詳しくは、「Vision Guide 7.0 プロパティ&リザルトリファレンス RobotToolXYUリザルト」を参照してください。



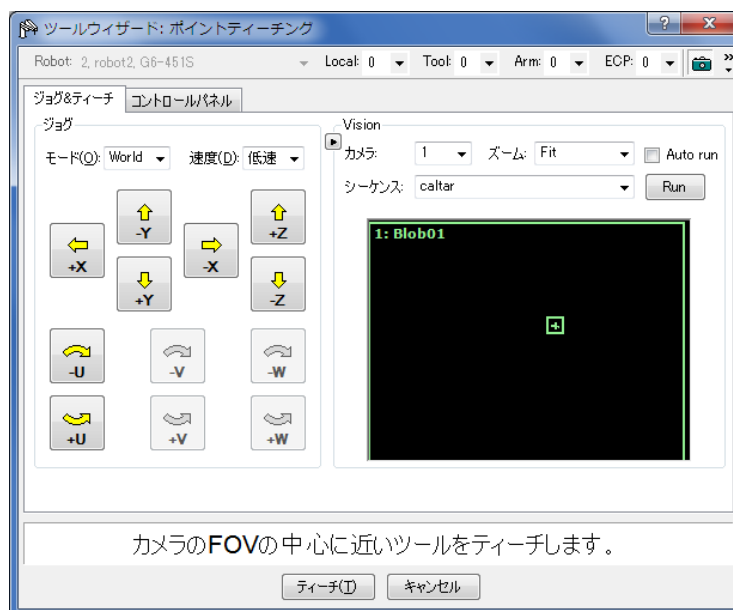
<次へ>ボタンをクリックします。

### Step4: カメラポイントのティーチ

<ティーチ...>ボタンをクリックします。ポイントティーチング画面が表示されます。



ロボットをジョグ動作し、ターゲットがカメラ視野中央付近に検出されるようにします。

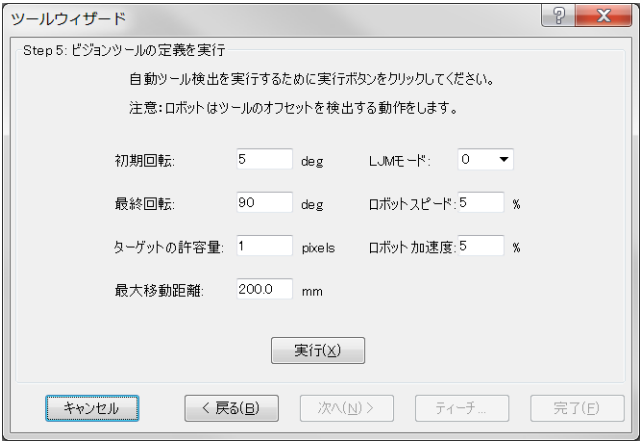


<ティーチ>ボタンをクリックします。

次のステップに進みます。



Step5: 実行



キャリブレーションしていない固定カメラを使用している場合は、必要に応じて以下の項目を設定します。キャリブレーション済み固定カメラを使用している場合は、この設定は、必要ありません。

項目	内容
初期回転	粗位置決めのために最初にツールを回転させる角度(°)です。
最終回転	最終的にツールを回転させる角度(°)です。 最終回転角度が大きい(90°以上を推奨)ほど、高精度にツール設定できます。
ターゲットの許容量	画像上での誤差が、この許容量に収まることを目標に、検出が実行されるようになります。
最大移動距離	アーム先端の移動距離の制限を設定します。 “0”の場合は、制限しません。
LJMモード	SPEL+のLJM関数で使われている値を指定します。 LJMモードは手首部の意図しない回転を防止するためにポイントデータの姿勢フラグを適切に制御します。 “0”の場合は、LJMを使用しません。
ロボットスピード	ロボットの速度を設定します。 精密なツール設定を行うためには、小さな値を設定します。
ロボット加速度	ロボットの加速度を設定します。 精密なツール設定を行うためには、小さな値を設定します。

<実行>ボタンをクリックします。

検出が完了すると、結果が表示されます。結果を確認してください。

<完了>ボタンをクリックすると、指定したツール番号に結果が設定されます

**注意:** ターゲットの検出結果に基づいて、自動的にロボットが動作します。ロボットと周辺装置の干渉に注意してください。またツール設定中のエラーを回避するため、各関節が伸びる特異点近傍姿勢を避けて使用してください。

### 7.10 カメラを使用した3Dツール設定

6軸ロボットを使用している場合、固定カメラを使用して、ロボット先端に取りつけられたツールの3Dツール設定(3次元位置・姿勢検出)ができます。

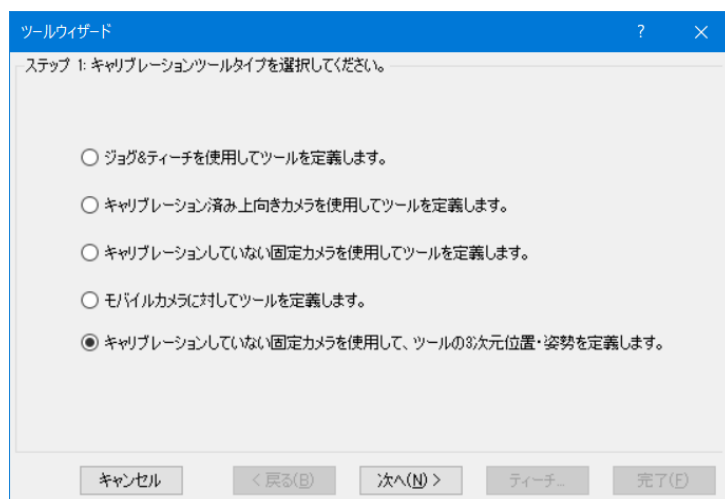
カメラを使用した3Dツール設定は、ツール設定ウィザードから行います。次のどちらかの方法で、ツール設定ウィザードを実行します。

- (1) ロボットマネージャの[ツール設定]タブを選択します。  
<ツール設定ウィザード...>ボタンが表示されます。  
このボタンをクリックします。
- (2) キャリブレーションウィザードの、ツール番号を選択するステップで、“0”以外のツール番号を選択すると、<ツールウィザード...>ボタンが表示されます。  
このボタンをクリックします。

#### Step1: タイプの選択

ツール設定ウィザードを開始すると、次のダイアログが表示されます。

<キャリブレーションしていない固定カメラを使用して、ツールの3次元位置・姿勢を定義します。>ボタンを選択します。



<次へ>ボタンをクリックします。

### Step2: ツール番号選択

設定するツール番号を選択します。



ツールの姿勢を求める場合は、<姿勢(UVW)の計算>をチェックします。  
ツールタイプとして、バー型、水平型のどちらかを選択します。

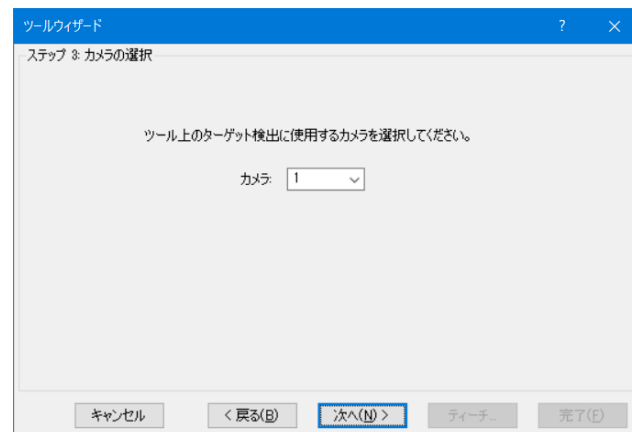
バー型: シリンダーなどの棒状のツールの姿勢を求めます。

平面型: 平面状のツールの姿勢を求めます。ツールに把持させたワークの水平面を利用して、ツール姿勢を求めることもできます。

<次へ>ボタンをクリックします。

### Step3: カメラの選択

ツール上のターゲット検出に使用するカメラを選択します。



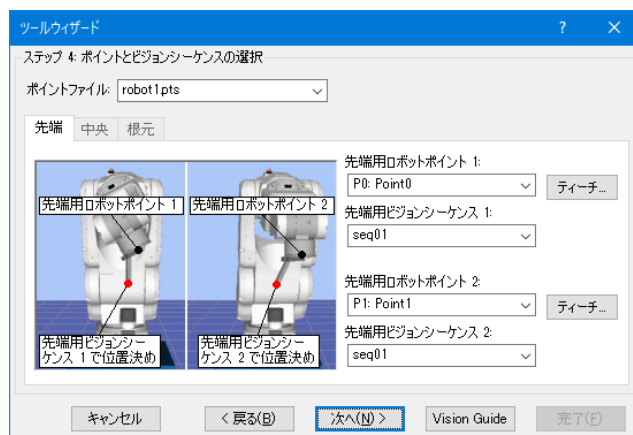
<次へ>ボタンをクリックします。

### Step4: ポイントとビジョンシーケンスの選択

ツール上のターゲット検出に使用するロボットポイントとビジョンシーケンスを選択します。

ロボットポイントでは、カメラの視野内にターゲットが入る位置を指定します。ロボットポイント1, 2は、異なる姿勢に設定してください。なお、ロボットポイントは、<ティーチ...>ボタンをクリックして、ポイントティーチング画面でティーチすることもできます。

ビジョンシーケンスでは、ターゲット検出に使用するビジョンシーケンスを選択します。ロボットポイント1, 2に応じて、それぞれビジョンシーケンスを選択してください。なお、ビジョンシーケンスは、<Vision Guide>ボタンをクリックして、Vision Guide画面で作成することもできます。



Step2で<姿勢(UVW)の計算>にチェックしている場合は、先端, 中央, 根元タブを切り替えて、全てのタブでロボットポイントとビジョンシーケンスを選択します。

<次へ>ボタンをクリックします。

Step5: キャリブレーションパラメーターの設定

ツールウィザード

?

×

ステップ 5: キャリブレーションパラメーターの設定

ターゲットの許容量:  pixels

LJMモード:

最大移動距離:  mm

ロボットスピード:  %

ローカル:

ロボット加速度:  %

キャンセル

< 戻る(B)

次へ(N) >

リセット

完了(F)

必要に応じて以下の設定を変更します。

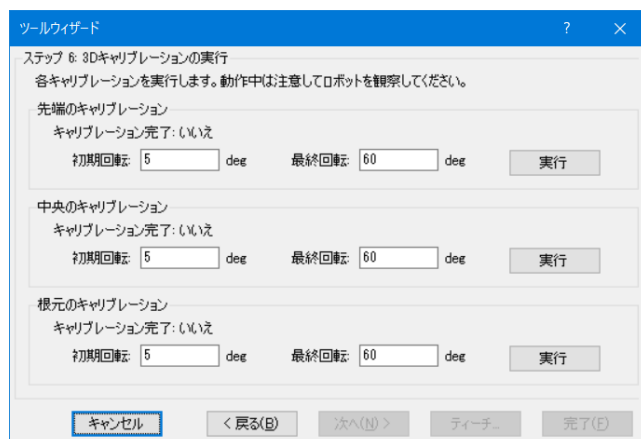
項目	内容
ターゲットの許容量	画像上での誤差が、この許容量に収まることを目標に、検出が実行されるようになります。
最大移動距離	アーム先端の移動距離の制限を設定します。 “0”の場合は、制限しません。
LJMモード	SPEL+のLJM関数で使われている値を指定します。 LJMモードは手首部の意図しない回転を防止するためにポイントデータの姿勢フラグを適切に制御します。 “0”の場合は、LJMを使用しません。
ロボットスピード	ロボットの速度を設定します。 精密なツール設定を行うためには、小さな値を設定します。
ロボット加速度	ロボットの加速度を設定します。 精密なツール設定を行うためには、小さな値を設定します。
ローカル番号	カメラの撮像面と平行なXY平面をもったローカル番号を指定します。

<次へ>ボタンをクリックします。

### Step6: 3Dツール設定の実行

必要に応じて、初期回転、最終回転を設定します。

<実行>ボタンをクリックすると、3Dツール設定が開始します。正常に完了すると、キャリブレーション完了が”はい”になります。



**注意:** ターゲットの検出結果に基づいて、自動的にロボットが動作します。ロボットと周辺装置の干渉に注意してください。またツール設定中のエラーを回避するため、各関節が伸びる特異点近傍姿勢を避けて使用してください。

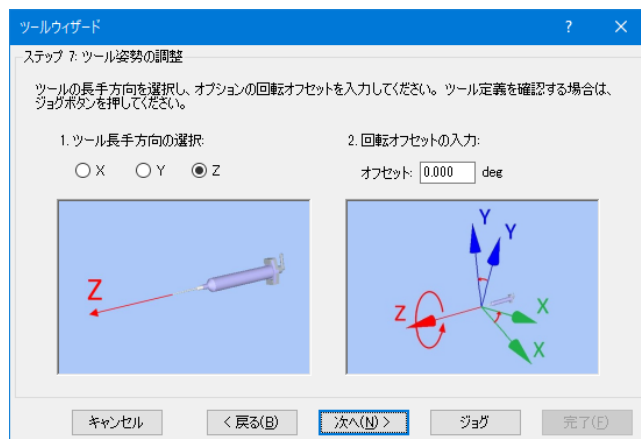
Step2で<姿勢(UVW)の計算>にチェックしている場合は、先端、中央、根元の全てで<実行>ボタンをクリックして、キャリブレーション完了を“はい”にします。

<次へ>ボタンをクリックします。

Step2で<姿勢(UVW)の計算>にチェックしている場合はStep7へ進みます。チェックしていない場合はStep8へ進みます。

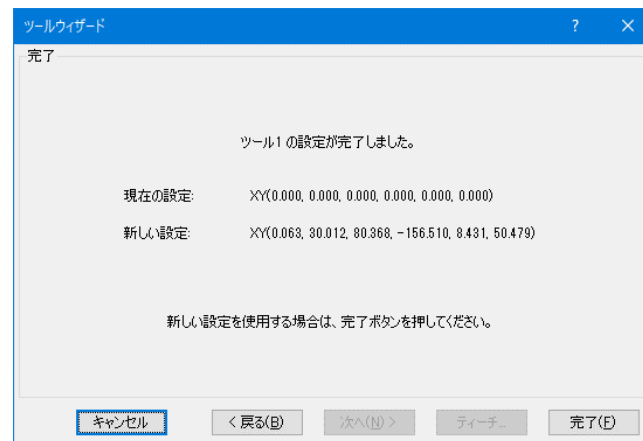
### Step7: ツール姿勢の調整

ツールの長手方向を何軸とするか選択します。必要に応じて、長手方向周りの回転オフセットを設定します。<ジョグ...>ボタンをクリックすると開くポイントティーチング画面で、ツール座標系での動作を確認することもできます。



## Step8: 3Dツール設定の完了

結果が表示されます。結果を確認してください。



<完了>ボタンをクリックすると、指定したツール番号に結果が設定されます。

## 8. ヒストグラムツール



ヒストグラムは、マシンビジョンシステムでの作業に、非常にパワフルな機能を発揮します。ヒストグラムを使用すると、図の形式でデータを見ることができるため、エッジが検出されない理由などを簡単に確認できます。また、ヒストグラム情報をもとに計算した統計値が表示されるため、画像の特性を把握することができます。



サーチウィンドウ内に暗いグレーのブローブと明るいグレーのブローブが検出できるときに、中間の明るさのグレーのブローブを検出するためのヒントを与えてくれます。


### 8.1 ヒストグラムの使用方法

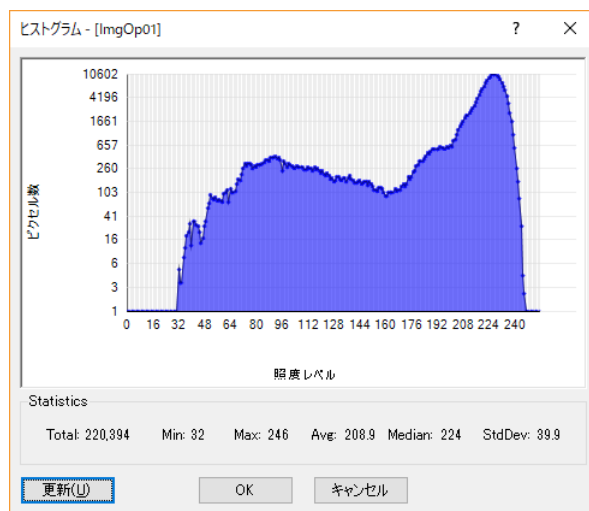
[ヒストグラム]ダイアログを開くには、まず、ヒストグラムが見たいビジョンオブジェクトを選択してください。(これはフローチャート、またはシーケンスツリーから選択できます。)

ヒストグラムは、次に示すビジョンオブジェクトにサポートされています。

Blob, Contour, DefectFinder, Correlation, Geometric, Polar, OCR, CodeReader, ImageOp

次に、ビジョンオブジェクト、またはビジョンシーケンスを実行して、選択したビジョンオブ

ジェクトが有効なリザルトを持つことを確認してください。それから、 <ヒストグラム> ボタンをクリックしてください。ビジョンオブジェクトがCorrelationオブジェクトの場合は、「8.2.1 Correlationオブジェクトで使うヒストグラム」で示したヒストグラムダイアログが表示されます。ビジョンオブジェクトがBlobオブジェクトの場合は、「8.2.2 Blobオブジェクトで使うヒストグラム」に示したダイアログが表示されます。



[ヒストグラム] ダイアログは、横軸に照度レベル、縦軸にピクセル数を表します。これにより、現在の画像イメージ中に、各照度レベルに対してピクセルがいくつ示されているかが簡単にわかります。[ヒストグラム]グラフは、左側にいく(あるいは照度レベルの低い方の数値、0に近づく)ほど、ピクセルの照度レベルが暗く(黒く)なっていることを示します。[ヒストグラム]ウィンドウの右側にいく(または照度レベルの高い方の数値、255に近づく)ほど、ピクセルの照度レベルが明るく(白く)なっていることを示します。



[ヒストグラム] ダイアログに表示されるグラフの結果を用いて、次の統計値が計算されます。

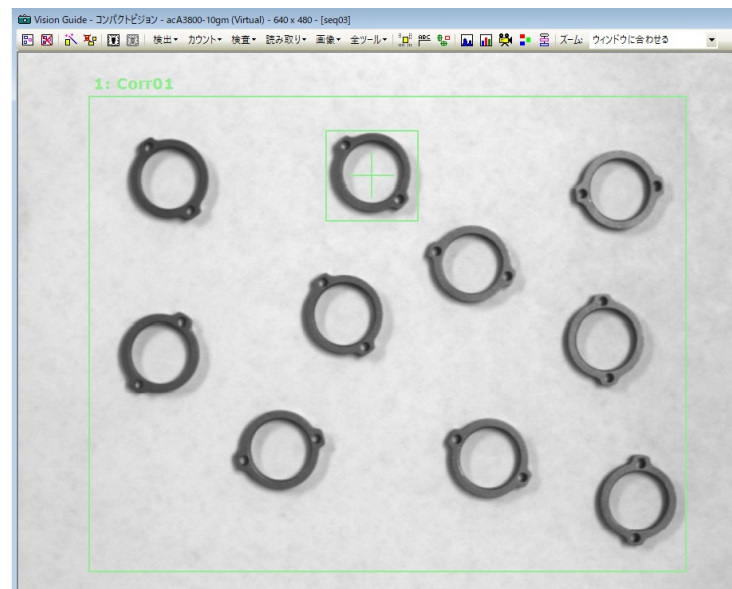
Statistics (統計値)	説明
Total (ピクセル総数)	サーチウィンドウ内全体のピクセル数
Min (最小)	サーチウィンドウ内全体の照度レベルの最小値
Max (最大)	サーチウィンドウ内全体の照度レベルの最大値
Avg (平均)	サーチウィンドウ内全体の照度レベルを計算して得られた平均値
Median (中央値)	サーチウィンドウ内全体の照度レベルの中央値
Std Dev (標準偏差)	サーチウィンドウ内全体の照度レベルを計算して得られた標準偏差

ビジョンオブジェクトが、Correlation, Geometric, Polar, OCR, CodeReader, ImageOp (Operation: Binarize以外の場合)オブジェクトの場合は、「8.2.1 Correlationオブジェクトで使うヒストグラム」で示した[ヒストグラム]ダイアログが表示されます。

ビジョンオブジェクトが、Blob, Contour, DefectFinder, ImageOp (Operation: Binarizeの場合)オブジェクトの場合は、「8.2.2 Blobオブジェクトで使うヒストグラム」に示した[ヒストグラム]ダイアログが表示されます。

## 8.2 ヒストグラムの例

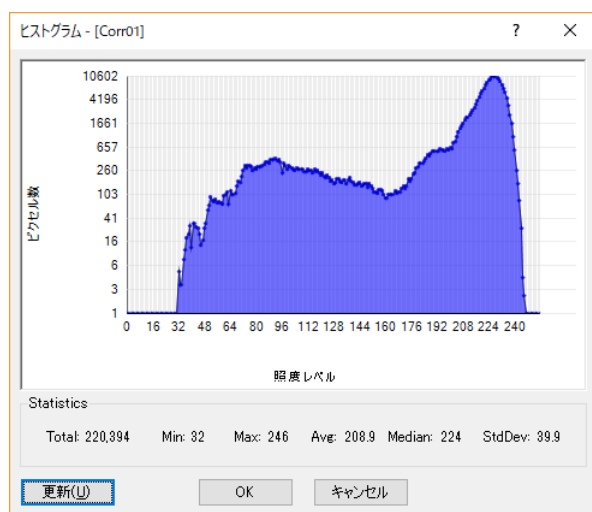
下図は、明るいグレーの背景にある10個の暗いリングです。このパーツをCorrelationオブジェクト、Blobオブジェクトで検出したときを例に挙げてヒストグラムについて説明します。



Corr01オブジェクトを使った10個のリングの画像イメージ例

## 8.2.1 Correlationオブジェクトで使うヒストグラム

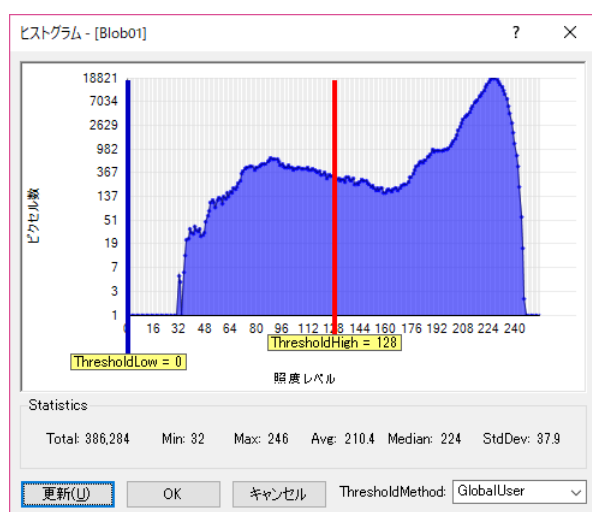
下図は、Corr01オブジェクトの生成された[ヒストグラム]ダイアログを示します。これは多様な照度レベル上に、画像イメージの散らばり方がわかるように、ピクセルがうまく広がっているのを示していることに注目してください。暗いピクセルより明るいピクセルの方が多い(「8.2 ヒストグラムの例」の画像イメージのほとんどがグレーの背景で暗いリングではない)ことが、わかります。



Correlationオブジェクト“Corr01”のヒストグラム例

## 8.2.2 Blobオブジェクトで使うヒストグラム

[ヒストグラム]ダイアログをBlobオブジェクトで使う場合は、Correlationオブジェクトで使う場合とは異なる点がいくつかあります。ThresholdLow, ThresholdHighという値を底部に持つ2本の垂直なバーがあることに注目してください。これらは、検出したブローブの部分として含む照度レベルと、背景の部分として含む照度レベルを指定するのに使われるBlobオブジェクトのプロパティです。

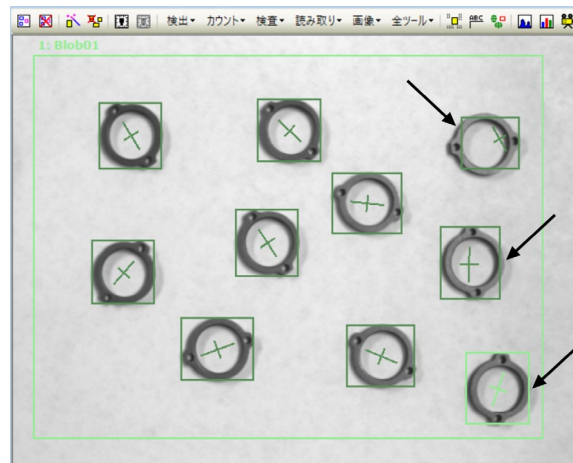


Blobオブジェクト“Blob01”のヒストグラム例 (明るい背景上の暗い画像イメージ)

ThresholdLowスライドバーとThresholdHighスライドバーの間の領域は、ThresholdColorプロパティで設定した、色(黒色または白色)として定義する照度レベルを持つピクセルの集まりです。Polarityプロパティは、白い背景上の黒色のオブジェクトを探すのか、または暗い背景上の白色のオブジェクトを探すのかを定義します。

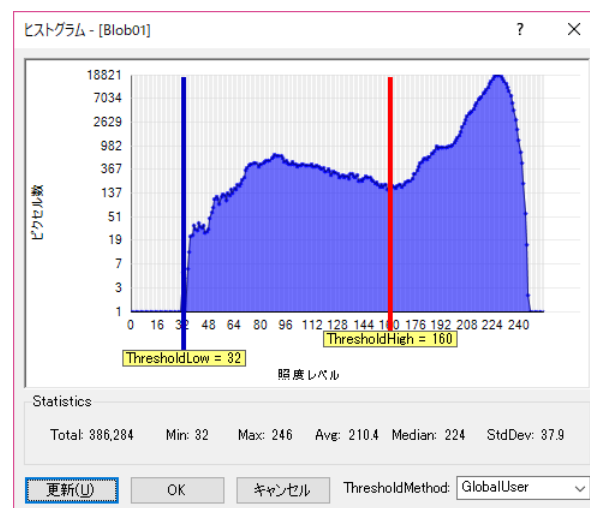
### ThresholdLow プロパティと ThresholdHigh プロパティの調整

もう一度、上図を見てください。ThresholdLowプロパティが0に、ThresholdHighプロパティが128に設定されていることに注目してください。これらはBlobオブジェクトのプロパティのデフォルト値です。リングの画像イメージでBlobオブジェクトを最初に行うとき(NumberToFindの設定を10にして)、下図で示すリザルトを得ます。検出された多くのブローブの極値は、リングの外側パーツには行き渡らず、パーツの一部のみがブローブとして検出されたり、1つのパーツが2つのブローブとして検出される場合があります(問題部分を示す下図の矢印を見てください)。これは、ヒストグラムのリザルトに基づいてThresholdLowプロパティとThresholdHighプロパティを調整していないからです。



適切なThreshold設定が行われていない10個のリングの画像イメージ

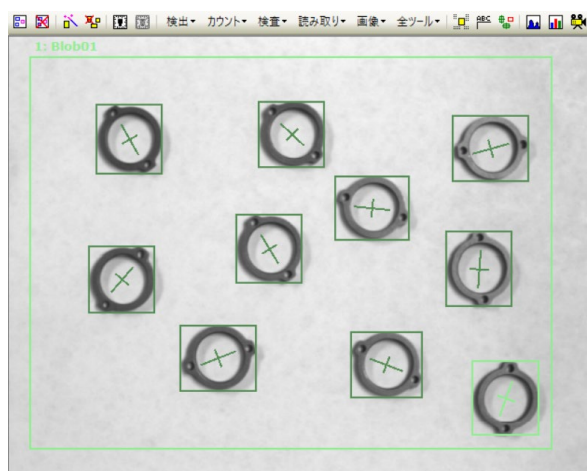
ヒストグラムダイアログを使ってリングの画像イメージのヒストグラムを確認する場合、レベル32あたりから始まるグレーの大きな分布が見えます。もう1つの大きなピークは170あたりから始まっています。リングの画像イメージでグレーの最大分布は明るい背景ですので、170を超えるピクセルの分布は背景のものと簡単にわかります。また、ヒストグラム中の他のピークは約32から170までですから、これはリング(検出したいパーツ)の画像イメージの暗い部分の分布です。ThresholdLow値とThresholdHigh値は、各ブローブの検出ボックスがちょうどリングの外側境界線にくるように、調整することができます。ヒストグラムダイアログのThresholdHighバーとThresholdLowバーをクリックして、下図に示す位置に動かせば、調整できます。



Threshold設定を行ったリングの画像イメージのヒストグラム

## 8. ヒストグラムツール

新しいThresholdLow設定とThresholdHigh設定でBlobオブジェクトを実行したあとで、リングの画像イメージを見ると、返されたリザルトがより望むものに一致しているのがわかります。各リングは、各ブローブの極値をきちんと示しています。

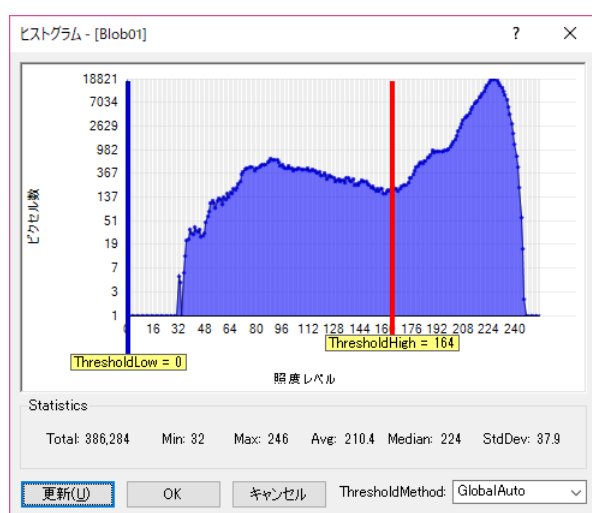


適切にThreshold設定を行った10個のリングの画像イメージ

また、ThresholdAutoチェックボックスがダイアログ右下にあります。このチェックボックスにチェックを入れることで、サーチウィンドウ内の画像から、適切と考えられるThresholdLowプロパティ値とThresholdHighプロパティ値を設定します。上図画像の場合、ThresholdLowプロパティ値は0に、ThresholdHighプロパティ値は164に設定されます。このThresholdAutoチェックボックスの状態は、ThresholdAutoプロパティ値と連動します。ThresholdAutoプロパティ値をTrueに設定した場合、画像処理実行時に閾値の計算が毎回行われます。これにより、照明環境が変化した場合でも、ブローブ検出が正しく機能し続けることが可能となります。

注意:

入力画像に処理するワークが映らない場合など、画面が均質(真っ黒や真っ白)となった場合においても、ThresholdAutoをTrueに設定している場合は、ブローブ検出が可能な閾値まで設定値が下がります。(常に1つ以上のブローブが検出されるようになります)



## 9. 統計ツール



統計計算は、Vision Guide 7.0の組み込み機能であり、非常に便利な機能であることがわかりいただけます。

Vision Guide 7.0は、Vision Guideウィンドウ、Runウィンドウ、オペレーターウィンドウでビジョンオブジェクトを実行するたびに、ビジョンオブジェクトごとの、各リザルトデータの履歴を自動的に保存し、今後の統計の参考とします。

つまり、ビジョンオブジェクトを実行するたびに、オブジェクトの実行方法には関係なく、実行結果のリザルトデータは、そのオブジェクトの結果に関する、統計計算に使用するために保存されます。

多くのビジョンアプリケーションは、さまざまな方法で実行できるので、複数の方法を試行してリザルトを比較することができます。このような場合に統計機能は能力を発揮します。ビジョンオブジェクトの信頼性テストを実行し、それによってアプリケーションにどのビジョンオブジェクトを使うのが最適であるかを定めることができます。また、これらのテストは、SPEL+言語あるいはプログラミング不要のVision Guideポイント クリック インタフェースで実行できます。

<統計>ツールバーボタンをクリックすると、統計ダイアログが表示されます。

## 9.1 ダイアログボックスオプションとその概要

オプション	説明
オブジェクト	統計表示するオブジェクトを、選択するためのドロップダウンリスト
リザルト番号	統計表示するリザルト番号を、選択するためのドロップダウンリスト
検出回数	オブジェクトの実行回数に対する、オブジェクト検出回数を表示 たとえば、5/6は、オブジェクトの実行回数が6回で、検出回数が5回であったことを示します。
データ消去	選択されているオブジェクトの、統計値を消去します。 全データが消去されます。
全データ消去	シーケンス中にあるすべてのビジョンオブジェクトの統計値を消去します。 全データが消去されます。
閉じる	[統計]ダイアログを終了します。 リザルトの全データは、そのまま残ります。
コピー	統計値をクリップボードにコピーします。
エクスポート	統計値をCSVファイルにエクスポートします。

統計 - [Test]

オブジェクト(O): Blob01

リザルト番号: 1 検出回数: 5 / 6

リザルト	単位	平均	標準偏差	範囲	最小	最大
PixelX	pixel	440.121	1.073	2.400	438.201	440.601
PixelY	pixel	383.247	1.004	2.245	381.451	383.696
Angle	deg	-49.244	0.729	1.629	-49.569	-47.940
CameraX	mm					
CameraY	mm					
RobotX	mm					
RobotY	mm					
RobotU	deg					
Time	ms	3.833	0.983	3.000	2.000	5.000
Area	pixel	14536.8	185.1	414.0	14454.0	14868.0

データ消去(B) 全データ消去(A) コピー(C) エクスポート(E)

閉じる

## 9.2 ビジョンオブジェクトとサポートされる統計

次に示すビジョンオブジェクトは、統計機能がサポートされています。

Blob, Correlation, Geometric, Edge, Line, Point, Polar, CodeReader, ColorMatch, BoxFinder, CornerFinder, LineFinder, LineInspector, ArcFinder, ArcInspector, DefectFinder, Coordinates

[統計]ダイアログでは、次の統計値を利用することができます。

統計値	説明
平均	特定のオブジェクトについて検出物体のすべてを計算して得られた平均値
標準偏差	基本的に $((\sum(x_i - \text{平均})^2) / (n - 1))$ の平方根で表される「サンプル標準偏差」 検出された物体のリザルトのみについて、計算を行います。
範囲	(最小-最大)で表されるリザルトの範囲 検出された物体のリザルトのみについて、計算を行います。
最小	特定のオブジェクトについて検出されたすべての物体のリザルトの最小値
最大	特定のオブジェクトについて検出されたすべての物体のリザルトの最大値

### 9.3 サポートされるビジョンオブジェクトリザルト

[統計]ダイアログには、次に示すビジョンオブジェクトリザルトの一覧が表示されます。各ビジョンオブジェクトについて、全てのリザルトが返されるわけではありません。

各リザルトの詳細は、「Vision Guide 7.0 プロパティ&リザルトリファレンス」をご覧ください。

リザルト	説明
PixelX	ピクセル座標系における検出パーツ位置のX座標位置を返す
PixelY	ピクセル座標系における検出パーツ位置のY座標位置を返す
Angle	ピクセル座標系における検出パーツの回転量を返す
Area	ブローブに関して検出される結合ピクセルの数
CameraX	カメラ座標系における検出パーツ位置のX座標位置を返す
CameraY	カメラ座標系における検出パーツ位置のY座標位置を返す
Length	Lineオブジェクトのmm単位の長さ (キャリブレーションが必要です。)
PixelLength	Lineオブジェクトのピクセル単位の長さ (キャリブレーションの必要はありません。)
RobotX	ロボット座標系における検出パーツ位置のX座標位置を返す
RobotY	ロボット座標系における検出パーツ位置のY座標位置を返す
RobotU	ロボット座標系における検出パーツ位置のU座標位置を返す (ロボット座標系におけるパーツの回転方向)
Time	ビジョンオブジェクトの実行に要した時間を返す (パーツを検出するのに要した時間)
Score	特定のパーツがどの程度適切に検出されたか (Correlationオブジェクトがそのモデルに一致している度合い、あるいはEdgeオブジェクトの強度)を表す



## 9.4 SPEL+から使用可能なビジョンオブジェクト統計

次のビジョンオブジェクト統計リザルトの一覧は、SPEL+言語から使用可能なものです。

ビジョンオブジェクト	SPEL+からサポートされている統計リザルト
Blob	AngleMax, AngleMean, AngleMin, AngleStdDev AreaMax, AreaMean, AreaMin, AreaStdDev CameraXMax, CameraXMean, CameraXMin, CameraXStdDev CameraYMax, CameraYMean, CameraYMin, CameraYStdDev PixelXMax, PixelXMean, PixelXMin, PixelXStdDev PixelYMax, PixelYMean, PixelYMin, PixelYStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev RobotUMax, RobotUMean, RobotUMin, RobotUStdDev TimeMax, TimeMean, TimeMin, TimeStdDev
Correlation	AngleMax, AngleMean, AngleMin, AngleStdDev CameraXMax, CameraXMean, CameraXMin, CameraXStdDev CameraYMax, CameraYMean, CameraYMin, CameraYStdDev PixelXMax, PixelXMean, PixelXMin, PixelXStdDev PixelYMax, PixelYMean, PixelYMin, PixelYStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev RobotUMax, RobotUMean, RobotUMin, RobotUStdDev ScoreMax, ScoreMean, ScoreMin, ScoreStdDev TimeMax, TimeMean, TimeMin, TimeStdDev
Geometric	AngleMax, AngleMean, AngleMin, AngleStdDev CameraXMax, CameraXMean, CameraXMin, CameraXStdDev CameraYMax, CameraYMean, CameraYMin, CameraYStdDev PixelXMax, PixelXMean, PixelXMin, PixelXStdDev PixelYMax, PixelYMean, PixelYMin, PixelYStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev RobotUMax, RobotUMean, RobotUMin, RobotUStdDev ScoreMax, ScoreMean, ScoreMin, ScoreStdDev TimeMax, TimeMean, TimeMin, TimeStdDev
Edge	CameraXMax, CameraXMean, CameraXMin, CameraXStdDev CameraYMax, CameraYMean, CameraYMin, CameraYStdDev PixelXMax, PixelXMean, PixelXMin, PixelXStdDev PixelYMax, PixelYMean, PixelYMin, PixelYStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev ScoreMax, ScoreMean, ScoreMin, ScoreStdDev TimeMax, TimeMean, TimeMin, TimeStdDev

ビジョンオブジェクト	SPEL+からサポートされている統計リザルト
Polar	AngleMax, AngleMean, AngleMin, AngleStdDev CameraXMax, CameraXMean, CameraXMin, CameraXStdDev CameraYMax, CameraYMean, CameraYMin, CameraYStdDev PixelXMax, PixelXMean, PixelXMin, PixelXStdDev PixelYMax, PixelYMean, PixelYMin, PixelYStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev RobotUMax, RobotUMean, RobotUMin, RobotUStdDev ScoreMax, ScoreMean, ScoreMin, ScoreStdDev TimeMax, TimeMean, TimeMin, TimeStdDev
Line	AngleMax, AngleMean, AngleMin, AngleStdDev LengthMax, LengthMean, LengthMin, LengthStdDev PixelLengthMax, PixelLengthMean, PixelMeanMin, PixelLengthStdDev
Point	AngleMax, AngleMean, AngleMin, AngleStdDev CameraXMax, CameraXMean, CameraXMin, CameraXStdDev CameraYMax, CameraYMean, CameraYMin, CameraYStdDev PixelXMax, PixelXMean, PixelXMin, PixelXStdDev PixelYMax, PixelYMean, PixelYMin, PixelYStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev
CodeReader	AngleMax, AngleMean, AngleMin, AngleStdDev CameraXMax, CameraXMean, CameraXMin, CameraXStdDev CameraYMax, CameraYMean, CameraYMin, CameraYStdDev PixelXMax, PixelXMean, PixelXMin, PixelXStdDev PixelYMax, PixelYMean, PixelYMin, PixelYStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev RobotUMax, RobotUMean, RobotUMin, RobotUStdDev TimeMax, TimeMean, TimeMin, TimeStdDev
ColorMatch	CameraXMax, CameraXMean, CameraXMin, CameraXStdDev CameraYMax, CameraYMean, CameraYMin, CameraYStdDev PixelXMax, PixelXMean, PixelXMin, PixelXStdDev PixelYMax, PixelYMean, PixelYMin, PixelYStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev TimeMax, TimeMean, TimeMin, TimeStdDev
BoxFinder	AngleMax, AngleMean, AngleMin, AngleStdDev CameraXMax, CameraXMean, CameraXMin, CameraXStdDev CameraYMax, CameraYMean, CameraYMin, CameraYStdDev PixelXMax, PixelXMean, PixelXMin, PixelXStdDev PixelYMax, PixelYMean, PixelYMin, PixelYStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev RobotUMax, RobotUMean, RobotUMin, RobotUStdDev TimeMax, TimeMean, TimeMin, TimeStdDev

ビジョンオブジェクト	SPEL+からサポートされている統計リザルト
CornerFinder	AngleMax, AngleMean, AngleMin, AngleStdDev CameraXMax, CameraXMean, CameraXMin, CameraXStdDev CameraYMax, CameraYMean, CameraYMin, CameraYStdDev PixelXMax, PixelXMean, PixelXMin, PixelXStdDev PixelYMax, PixelYMean, PixelYMin, PixelYStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev RobotUMax, RobotUMean, RobotUMin, RobotUStdDev TimeMax, TimeMean, TimeMin, TimeStdDev
LineFinder	AngleMax, AngleMean, AngleMin, AngleStdDev LengthMax, LengthMean, LengthMin, LengthStdDev PixelLengthMax, PixelLengthMean, PixelMeanMin, PixelLengthStdDev TimeMax, TimeMean, TimeMin, TimeStdDev
LineInspector	CameraXMax, CameraXMean, CameraXMin, CameraXStdDev CameraYMax, CameraYMean, CameraYMin, CameraYStdDev LengthMax, LengthMean, LengthMin, LengthStdDev PixelLengthMax, PixelLengthMean, PixelMeanMin, PixelLengthStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev TimeMax, TimeMean, TimeMin, TimeStdDev
ArcFinder	CameraXMax, CameraXMean, CameraXMin, CameraXStdDev CameraYMax, CameraYMean, CameraYMin, CameraYStdDev PixelXMax, PixelXMean, PixelXMin, PixelXStdDev PixelYMax, PixelYMean, PixelYMin, PixelYStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev RobotUMax, RobotUMean, RobotUMin, RobotUStdDev TimeMax, TimeMean, TimeMin, TimeStdDev
ArcInspector	CameraXMax, CameraXMean, CameraXMin, CameraXStdDev CameraYMax, CameraYMean, CameraYMin, CameraYStdDev LengthMax, LengthMean, LengthMin, LengthStdDev PixelLengthMax, PixelLengthMean, PixelMeanMin, PixelLengthStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev TimeMax, TimeMean, TimeMin, TimeStdDev

ビジョンオブジェクト	SPEL+からサポートされている統計リザルト
DefectFinder	AngleMax, AngleMean, AngleMin, AngleStdDev AreaMax, AreaMean, AreaMin, AreaStdDev CameraXMax, CameraXMean, CameraXMin, CameraXStdDev CameraYMax, CameraYMean, CameraYMin, CameraYStdDev PixelXMax, PixelXMean, PixelXMin, PixelXStdDev PixelYMax, PixelYMean, PixelYMin, PixelYStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev RobotUMax, RobotUMean, RobotUMin, RobotUStdDev TimeMax, TimeMean, TimeMin, TimeStdDev
Coordinates	AngleMax, AngleMean, AngleMin, AngleStdDev CameraXMax, CameraXMean, CameraXMin, CameraXStdDev CameraYMax, CameraYMean, CameraYMin, CameraYStdDev PixelXMax, PixelXMean, PixelXMin, PixelXStdDev PixelYMax, PixelYMean, PixelYMin, PixelYStdDev RobotXMax, RobotXMean, RobotXMin, RobotXStdDev RobotYMax, RobotYMean, RobotYMin, RobotYStdDev RobotUMax, RobotUMean, RobotUMin, RobotUStdDev

## 10. チュートリアル

### 10.1 クイックスタート: Vision Guide 7.0の チュートリアル

#### 10.1.1 チュートリアルの概要

本章の目的は、Vision Guide 7.0の簡単な使用例をいくつか紹介することによって、基本的な用途について説明し、使いやすさを理解していただくことです。ここでは、ほとんどの場合、操作手順だけを説明し、システム動作の詳細については説明していません。これは、このチュートリアルで説明する使用例に、なるべく早く慣れていただくためです。詳細は、他の章で後述しますので参照してください。

このチュートリアルでは、いくつかのオブジェクトの図面を用意しています。図面のコピーを作成し、カメラの下へセットし、それを使って説明します。

このチュートリアルでは、ビジョンを使ってパーツを認識し、ロボットをそのパーツへ動作させるための簡単なアプリケーションの作成方法を説明します。水平多関節型ロボットの第2アームにカメラが搭載されていることを前提として、説明を行います。

本章は、次の項目について説明します。

- 本チュートリアルを使用するにあたって必要となる事項
- カメラレンズの構成
- EPSON RC+ 7.0 新規プロジェクトの作成方法
- 新規ビジョンシーケンスの作成方法
- Blobオブジェクトを使ってパーツを認知する方法
- ビジョンシーケンスと連動するSPEL+言語プログラムの作成方法
- カメラでロボットをキャリブレーションする方法
- ビジョンを使用して、ロボットにパーツへの動作を指示する方法
- 複数の類似パーツを検出して、そのパーツへの動作を指示する方法

次の「本チュートリアルを使用するにあたって必要となる事項」には、本チュートリアルで使われるターゲットを印刷して添付しています。このターゲットは、本チュートリアル全体で使います。

ターゲットが印刷されているページに続いて、「EPSON RC+ 7.0の起動と新規プロジェクトの作成方法」という項目があります。この項目以降が、実際の操作方法の説明になります。

## 10.1.2 本チュートリアルを使用するにあたって必要となる事項

本チュートリアルでは、Vision Guide 7.0でエプソンロボットを操作する方法について説明します。また、EPSON RC+ 7.0とエプソンロボットを、十分に使用していただいていることを前提としています。

- PCにEPSON RC+ 7.0がインストールされていなければなりません。
- カメラを装着し、正しく動作するようにしてください。
- エプソンロボットが、作業面に届くように設置してください。
- スカラロボットの場合、カメラはロボットの第2アームに装着してください。6軸ロボットの場合、カメラは第6関節のフランジに装着してください。オプションのカメラ取付治具を使うと、簡単に装着できます。カメラは、下向きに装着してください。

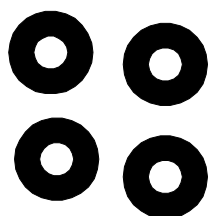


PC、カメラ、コントローラー、およびイーサネットスイッチ(使用する場合)にイーサネットケーブルを装着するときは、PCの主電源、コントローラーの電源、およびカメラの電源がOFFになっていることを確認してください。電源がONの状態に接続すると、カメラが損傷する可能性があります。

- ターゲットパーツの図をコピーして、カメラの下に置いてください。  
(ターゲットパーツの図は、この説明項目の次のページにあります。)
- カメラレンズには、約40mm×30mmの視野を持つレンズを選択してください。ここでのテストは、作業面からのWD(ワークディスタンス)約210mm、16mmレンズを選択し、可動カメラ(CV1とNET 1044BUカメラ)を使用します。多少、フォーカス合わせの作業をする必要がありますが、これについても本チュートリアルで説明します。
- 本チュートリアルでは、実際のパーツのピックアップは行わず、ロボットをターゲットシート上に描かれたパーツのポジションに動作させる作業を行います。そのため、ロボットの第3軸(J3軸)シャフトには、グリッパーかロッドを取りつける必要があります。本チュートリアルでは、第3軸(J3軸)シャフトにロッドが装着されていることを前提として説明します。ロッドは、シャフトの中心に来るように垂直下向きに取り付けると良好に動作します。ロッドは、キャリブレーションや、ターゲットパーツへの動作に使われます。



チュートリアル用の単一ターゲット (ワッシャーの図面)





チュートリアル用の複数ターゲット(複数ワッシャーの図面)



### 10.1.3 EPSON RC+ 7.0の起動と新規プロジェクトの作成

- (1) コントローラーの電源をオンします。
- (2) Windows デスクトップ上の<EPSON RC+ 7.0>アイコンをダブルクリックして、EPSON RC+ 7.0 を起動します。
- (3) EPSON RC+ 7.0 メニュー-[プロジェクト]を選択します。
- (4) [プロジェクト]メニューの[新規プロジェクトメニューエントリー]をクリックします。[新規プロジェクト]ダイアログが表示されます。
- (5) 新規プロジェクト名を入力してください。できれば、名称の最後に“tut”をつけるようにしてください。このように、本チュートリアルを使ってテスト作業をするときには、自分で作成したプロジェクトに独自の名称をつけることができます。ここでは、新規プロジェクトに“vgtut”という名称をつけることにします。名称のフィールドに名称を入力したら、<OK>ボタンをクリックしてください。以上で、“xxxxtut”という名称の新規プロジェクトの作成が完了しました。

### 10.1.4 新規ビジョンシーケンスの作成

- (1) EPSON RC+ 7.0 新規プロジェクトの作成が完了すると、それまで灰色に表示されていたツールバーの多くが色つきで表示されます。 <ビジョン>ボタンを探してクリックしてください。これで、Vision Guide ウィンドウが開きます。
- (2) Vision Guide ウィンドウで操作を始める前に、まずビジョンシーケンスを作成しなければなりません。 <新規シーケンス>ボタンをクリックします。(このツールバーボタンは、Vision Guide ウィンドウのツールバーにあります。EPSON RC+ 7.0 のツールバーにはありませんので注意してください。) ボタンをクリックすると、[新規シーケンス]ダイアログが表示されます。
- (3) 新規ビジョンシーケンスに、“blobtut”という名称を入力して、<OK>ボタンをクリックしてください。“blobtut”という名称は、後ほど EPSON RC+ 7.0 コードで使います。スペルをまちがえないように入力してください。引用符(“ ”)はつけないでください。以上で、“blobtut”という名称の新規ビジョンシーケンスの作成が完了しました。これから行う作業はすべて、このビジョンシーケンスに関するものです。

### 10.1.5 チュートリアルでのカメラレンズの構成

前述の説明で、チュートリアルでのターゲット視野は、約40mm×30mm、また作業面からのWD (ワークディスタンス)は、約210mmとしていました。16mmレンズと1mmのエクステンションチューブでフォーカスの合うWDは、167mm~240mmであることが分かっています。このことから、このチュートリアルでは16mmレンズと1mmのエクステンションチューブを使用します。

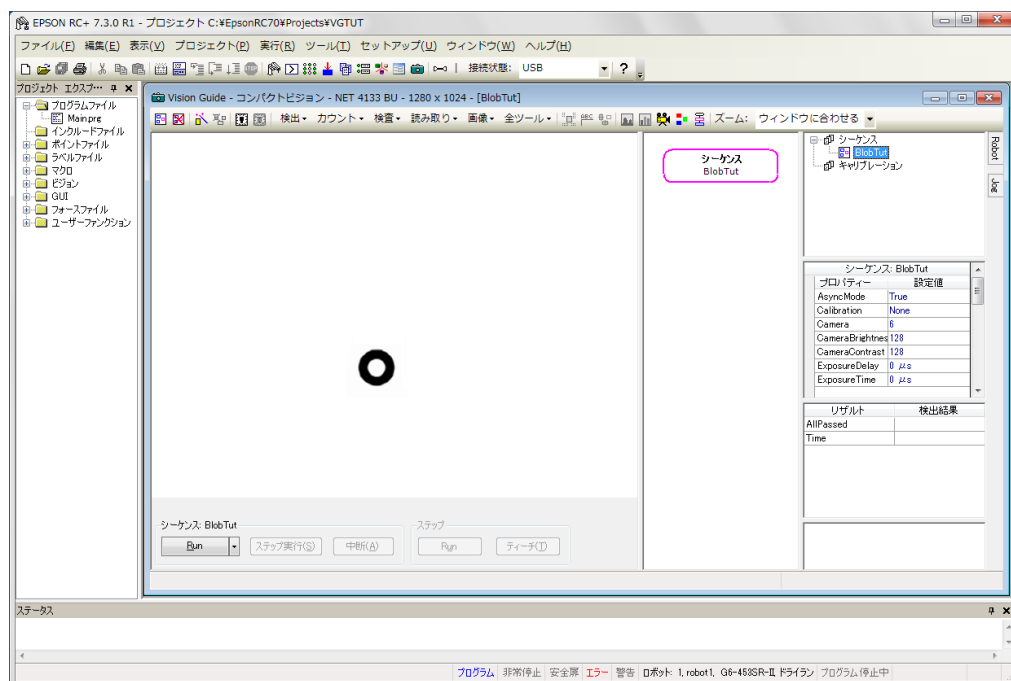
まだカメラにレンズを取りつけて焦点合わせをしていなければ、「インストールのしかた」の章を参照し、焦点合わせまでを行ってください。レンズの選択方法と焦点の合わせ方がよくわからない場合も、次の作業に進む前に、もう一度お読みください。

Vision Guide 7.0からイメージを見て焦点をチェックします。Vision Guideウィンドウはすでに開いているので、ターゲットシート上のターゲットの特性が正しく見えるかどうかを確認します。手順は、次に説明するとおりです。

### 可動カメラをターゲットシートの上方に動かし、レンズのフォーカスを調整する

- (1) 先に作成したターゲットシートのコピーで、中央に1個のワッシャーが印刷されている方を用意してください。このシートを作業面に置きます。ロボットの真正面など、カメラが簡単にこのシート上方に移動できる位置に置いてください。
- (2) ロボットを動作させて、カメラがワッシャーの図の上方に位置するようにしてください。(モーターパワーをオンしなくても、手でアームを動かすことができます。)
- (3) Vision Guideウィンドウの画像イメージ表示部にターゲットのワッシャーが見えるはずです。カメラレンズのフォーカスを調整して、ターゲットにフォーカスを合わせてください。ターゲットが見えない場合、あるいは、ターゲットにフォーカスを合わせることができない場合は、「インストールのしかた」-「焦点距離のチェックと調整」を参照してください。この項目では、カメラレンズの選択方法を詳細に説明しています。

今、カメラはターゲットの真上にあり、Vision Guideウィンドウの画像イメージ表示部に、ターゲットが明瞭に見えるはずです。このとき、Vision Guideウィンドウは、下図のように表示されます。ターゲット(ワッシャー)は、Vision Guideウィンドウの画像イメージ表示部中央に表示されています。




画像イメージ表示部中央にターゲットが表示されているVision Guideウィンドウ

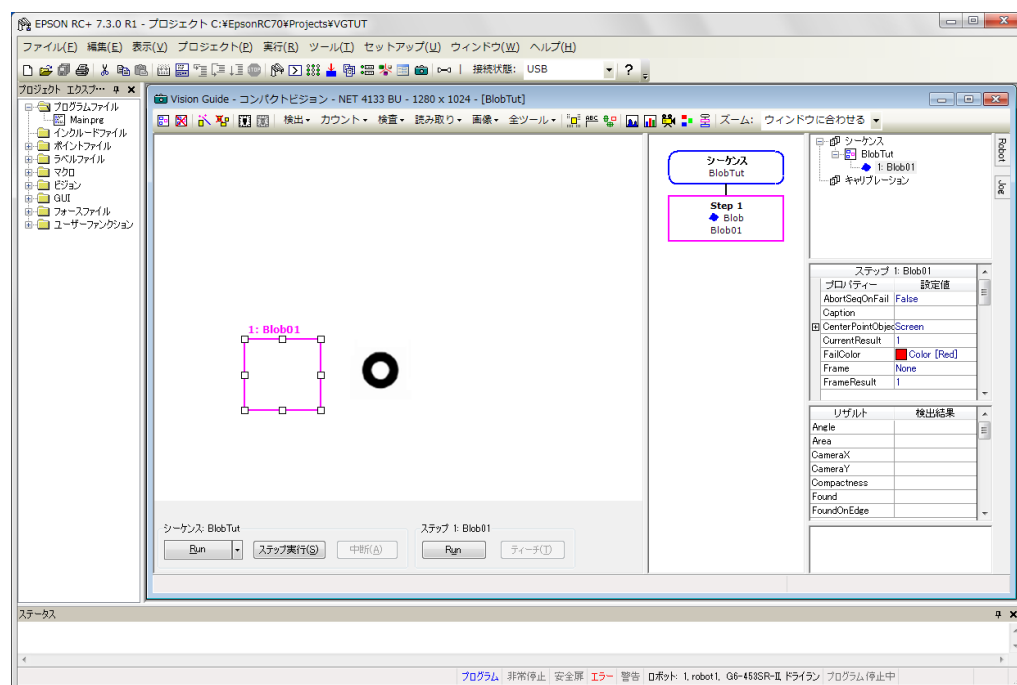
### 10.1.6 Blobオブジェクトを使ってパーツを検出する方法

ターゲットのワッシャーが、Vision Guideウィンドウの画像イメージ表示部中央に表示されているとき、Blobオブジェクトを作成して、ワッシャーを検出することができます。Blobオブジェクトの作成と設定、ワッシャーを検出する手順は、次に説明するとおりです。

#### Step 1: Blob オブジェクトの作成

- (1) Vision Guideツールバー <全ツール> -  <Blob(2値検査)>ボタンをクリックしてください。<Blob (2値検査)>ボタンをクリックしたら、マウスの左側ボタンを押し続けなくてください。
- (2) 画像イメージ表示部中央のワッシャーのイメージに向かって、マウスを動かしてください。Vision Guideツールバーから離れると、マウスポインターはBlobオブジェクトアイコンに変わります。
- (3) 画像イメージ表示部中央へ、マウスを動かし続けてください。そして、マウスを左クリックして離してください。以上の操作で、Blobオブジェクトは画像イメージ表示部に移動します。

画面には、下図のように表示されます。



新規Blobオブジェクト“Blob01”を表示するVision Guideウィンドウ

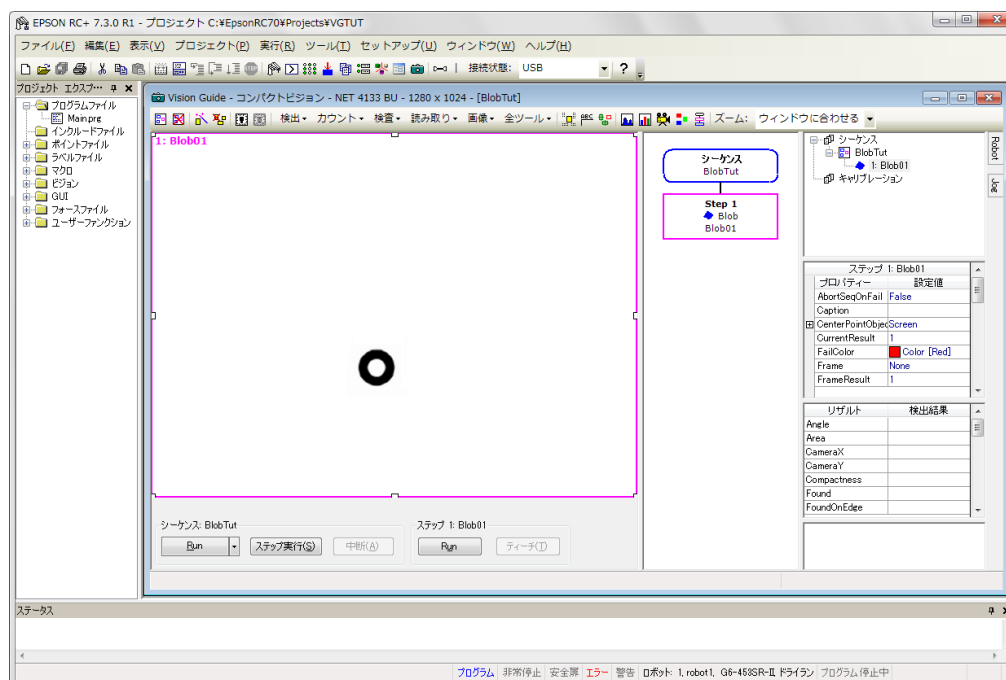
## Step 2: Blobオブジェクトの位置とサイズ

ここで、Blobオブジェクトのサーチウィンドウの位置とサイズを決定する必要があります。

“Blob01” Blobオブジェクトのサーチウィンドウは、下図のワッシャーの左側に見えるボックスです。ワッシャーの全体を視野に入れて近くで検索できるように、サーチウィンドウのサイズを大きくしてみましょう。

- (1) マウスポインターをBlobオブジェクトの名称ラベルに移動して、マウスの左側ボタンを押してください。マウスの左側ボタンを押しながら、Blobオブジェクトを画像イメージ表示部の左上コーナーにドラッグし、サーチウィンドウの左上コーナーが画像イメージ表示部の左上コーナーに重なるようにしてください。
- (2) マウスポインターをBlob01サーチウィンドウの右下サイズハンドルに移動して、マウスの左側ボタンを押してください。マウスの左側ボタンを押しながら、右下サイズハンドルを画像イメージ表示部の右下コーナーにドラッグしてください。Blobオブジェクトのサーチウィンドウが画像イメージ表示部全体の上に重なります。以上の操作によって、カメラの視野内にあるどのようなブローブでも検出することができます。

下の画面は、画像イメージ表示部の全体に重なるように、位置とサイズを変更した“Blob01” Blobオブジェクトのサーチウィンドウを示します。



大きなサーチウィンドウをもつ"Blob01"Blobオブジェクト

### Step 3: プロパティの設定とBlobオブジェクトの実行

ステップ2の操作を終了すると、サーチウィンドウは、サーチウィンドウ内にワッシャーを見るのに十分な大きさになっています。これで、Blobオブジェクトがワッシャーを検出できるかどうかの確認テストができます。

- (1) [Vision Guide]ウィンドウの右上にあるツリーの中の Blob01 オブジェクトを確認してください。Blob01 オブジェクトのプロパティとリザルトを表示します。

ステップ 1: Blob01	
プロパティ	設定値
AbortSeqOnFail	False
Caption	
CenterPointObject	Screen
CurrentResult	1
FailColor	<span style="color: red;">■</span> Color [Red]
Frame	None
FrameResult	1

リザルト	検出結果
Angle	
Area	
CameraX	
CameraY	
Compactness	
Found	
FoundOnEdge	

**Caption**  
オブジェクトラベルに表示するキャプションを割り当てます。

- (2) プロパティリストから Name プロパティを見つけてください。Name プロパティのフィールドでダブルクリックすると、現在のプロパティ名が強調表示されます。ここで、名称を washer として入力してください。これで、Blob オブジェクトのラベル名が “washer” に変更されました。名称ドロップダウンリスト中のジョグタブの一番上と、サーチウィンドウのラベル名を見ると、両方の場所で名称が変更されているのがわかります。

シーケンス

BlobTut

1: Washer

キャリブレーション

ステップ 1: Blob01

プロパティ	設定値
MaxArea	200000 pixels
MinArea	25 pixels
Name	Washer
NumberToFind	1
PassColor	Color [LightGreen]
PassType	SomeFound
Polarity	DarkOnLight

検出結果 1 / 1

検出結果
Angle
Area
CameraX
CameraY
Compactness
Found
FoundOnEdge

Name

オブジェクト、シーケンス、キャリブレーションの名前を設定します。

ここに“washer”と入力してください。

Blob01 オブジェクトのプロパティ

- (3) プロパティリストを見ながら、Polarityプロパティをチェックすることもできます。新規ブローブを作成すると、Polarityプロパティのデフォルトは“Dark on Light”になっていますが、これは明るい背景で暗いブローブを検出するという意味です。このプロパティは“Light on Dark”に変更することができますが、ここでは明るい背景で暗いブローブ(ワッシャー)を検出したいので、Polarityプロパティは変更せずにこのままにしておいてください。

- (4) 以上で、Blobオブジェクト“washer”を実行する準備ができました。

オブジェクトを実行するには、実行画面の左下にある<実行>ボタンをクリックしてください。この操作によって、Blobオブジェクト“washer”が実行され、このチュートリアルの場合では、Blobオブジェクトはワッシャーのように見えるブローブを検出します。

Blobオブジェクトの実行後にサーチウィンドウの色の変化によって、ブローブが検出できたことがわかります。Blobオブジェクトがブローブを検出すると、サーチウィンドウの色は緑色に変化し、検出できなかった場合は、赤色に変化します。

ブローブが検出できたかどうかは、リザルトリストで確認することもできますが、これについては後述します。

- (5) ワッシャーの描かれたターゲットシートをわずかに動かし、実行画面の<実行>ボタンを、もう一度クリックしてみてください。

自分が設定したサーチウィンドウ内にワッシャーがあることを確認してください。画像イメージ表示部内に、ブローブの新規ポジションが検出され、緑色に強調表示されるのが見えます。

ワッシャーがサーチウィンドウの枠外に出るように紙を動かすと、Blobオブジェクトはワッシャーを検出することができません。これは、オブジェクトが検出できなかったときには、サーチウィンドウの色が赤色に変化することからも確認することができます。

Step 4: リザルトのチェックのしかた

“washer”という名称のBlobオブジェクトを実行すると、このオブジェクトに返ったリザルトを確認することができます。リザルトは、プロパティリストの下にあるリザルトリストに表示されます。

- (1) リザルトリストから、Foundという名称のリザルトを探してください。この時点でブローブが検出できているので、Foundリザルトは、“True”になっています。ブローブが検出できなかった場合、Foundリザルトには、“False”が表示されます。
- (2) また、リザルトリストの一番上で、Areaリザルトを確認することができます。これは、ブローブが検出できた領域を示します。
- (3) 垂直スクロールバーを使って、リザルトリストを一番下まで移動してください。Blobリザルトリストの一番下では、Timeリザルトを確認することができます。このリザルトは、ブローブの検出にかかった時間を表します。

ステップ 1: Blob01

プロパティ	設定値
MaxArea	200000 pixels
MinArea	25 pixels
Name	Washer
NumberToFind	1
PassColor	<div><div></div>Color [LightGreen]</div>
PassType	SomeFound
Polarity	DarkOnLight

検出結果 1 / 1

	検出結果
RobotX	(no cal) mm
RobotY	(no cal) mm
RobotU	(no cal) deg
Roughness	1.38
ShowAllResults	Click to show
Time	9 ms
TotalArea	81758.0 pixels

リザルトリストをスクロールダウンするには、このスクロールバーボタンをクリックしてください。(または、スライダーを使ってください。)

リザルトリストを一番下までスクロールして表示しているTimeリザルト




注意

■ ビジョンシーケンスの実行結果は、外らん光や外部機器ノイズにより影響を受けることがあります。  
外らん光や外部機器ノイズにより、影響を受けると、取り込み画像が想定外の結果となり、検出位置結果がサーチエリア(検出エリア)内の不特定位置になる可能性があります。  
サーチエリアは、極力小さく、また誤検出に配慮した画像処理シーケンスの作成を行ってください。



### Step 5: ビジョンシーケンスを保存する方法

この時点で、これまでの作業を保存します。どのようなアプリケーションを使用する場合でも、作業の途中で頻繁に保存をするように心がけてください。EPSON RC+ 7.0のプロジェクトマネージャー機能を使うと、このプロジェクトに関するデータすべてを一度に保存することができます。この手順は、次に示すとおり非常に簡単です。

EPSON RC+ 7.0ツールバー -  <プロジェクトの保存>ボタンをクリックしてください。プロジェクトに何らかの変更を行った場合には、<プロジェクトの保存>ボタンが青色に表示されています。

以上で、新規Blobオブジェクトの作成、サーチウィンドウの位置とサイズの設定ができ、またブローブを検出するために、Blobオブジェクトを実行することができました。

次の手順に進んでみましょう。SPEL+言語からビジョンシーケンスを実行する簡単なプログラムを作成し、プログラムで使うリザルトを取り出す方法について説明します。


#### 10.1.7 ビジョンシーケンスを使用するSPEL+プログラムの作成

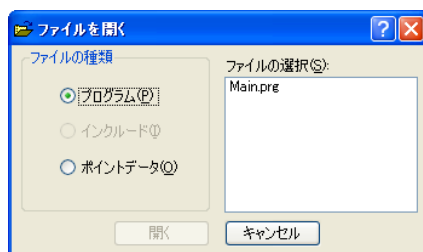
Vision Guide 7.0の最も強力な機能の1つは、ポイント クリック環境から作成されたビジョンシーケンスが、SPEL+言語から使用可能であるということです。つまり、ビジョンシーケンスはセイコーエプソンのビジョンアプリケーションの核心であって、SPEL+での全面的な書き直しを必要とされるようなプロトタイプツールではありません。ビジョンシーケンスとSPEL+言語は統合されて、両者の特性を生かした最高の操作環境を提供します。

この項では、ブローブが検出できたかどうかの確認、ブローブの領域のチェック、これらの情報を含むメッセージのプリントアウトをするためのクイックプログラムを書いてみます。ポイント クリック操作による開発環境の使いやすさ、そして言語のパワーと柔軟性をご理解いただけるはずです。

#### ファイル名MAIN.PRGのプログラムのオープン

EPSON RC+ 7.0の環境に慣れている方は、プログラムファイルの開き方をすでにご存じだと思います。しかし、ここでは、ファイルの開き方がわからない方のために基本的な手順を説明します。

- (1) EPSON RC+ 7.0ツールバー -  <ファイルを開く>ボタンをクリックしてください。すると、下図のようなダイアログボックスが表示されます。



EPSON RC+ 7.0[ファイルを開く]ダイアログ

- (2) まだ他のファイルは作成していないので、上記の[ファイルを開く]ダイアログ中では、main.prgプログラムファイルが反転表示されているはずです。そのままダイアログの左下にある<開く>ボタンをクリックし、main.prgプログラムファイルを開いてください。

### ビジョンシーケンスを使用するSPEL+プログラムの作成

下に示すのは、ビジョンシーケンス “blubut” を実行し、Blobオブジェクト “washer”に関連するプロパティのいくつかをチェックするためのサンプルプログラムです。

例えば、ブローブが検出できたかどうかをチェックすることになります。検出できた場合は、「The washer was found!」というメッセージとともに、ブローブの領域が表示されます。検出できなかった場合は、「The washer was not found!」というメッセージが表示されます。

タイトルバーに **MAIN.PRG** と表示されている編集ウィンドウを見てください。カーソルが、編集ウィンドウの 1 番目の行の 1 番初め(プログラム作成を開始するときの位置)に位置しているはずです。この編集ウィンドウで、次に示すプログラムを入力してください。このとき、大文字と小文字を区別して入力する必要はありません。エディターは、すべてのキーワードを自動的に変換します。


```
Function main
  Real area
  Boolean found

  VRun blobtut
  VGet blobtut.washer.Found, found

  If found = TRUE Then
    VGet blobtut.washer.Area, area
    Print "The washer was found!"
    Print "The washer area is: ", area, "Pixels"
  Else
    Print "The washer was not found!"
  EndIf
Fend
```

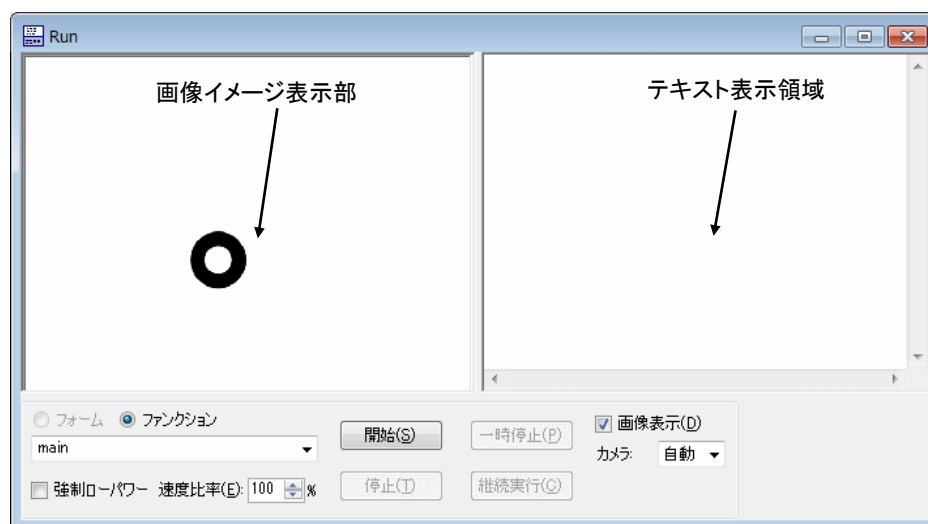
## mainファンクションの実行

EPSON RC+ 7.0のRunウィンドウの使い方は、すでにご存じだと思います。Runウィンドウを使って、上述のセクションで作成したサンプルプログラムを実行します。実行の手順は、次に示すとおりです。

- (1) EPSON RC+ 7.0ツールバー -  < Runウィンドウ>ボタンをクリックしてください。この操作によって、Runウィンドウが表示されます。下図を参照してください。Runウィンドウは、2つの部分に分割されています。Runウィンドウの左半分は画像イメージ表示部で、右半分はテキストの領域になっています。

NOTE  

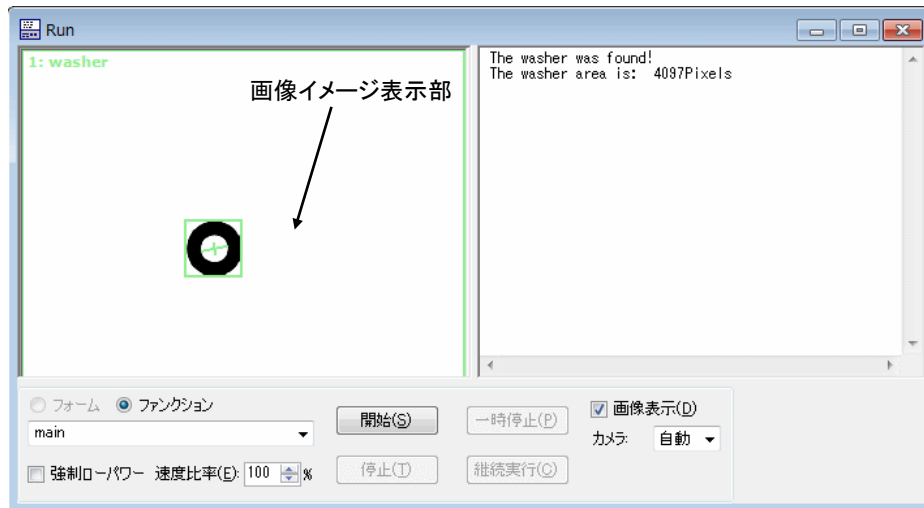

Runウィンドウがテキスト表示領域しか表示せず、画像イメージ表示部が表示されない場合は、Runウィンドウの[ビジョン画像の表示]チェックボックスをクリックしてください。



画像イメージ表示部とテキスト表示領域を示すRunウィンドウ

- (2) Runウィンドウの左下コーナーにある<開始>ボタンをクリックしてください。この操作によって、“main”ファンクションが実行されます。
- (3) “main”ファンクションを実行すると、手順(4)の図に示すような実行例を見ることができます。プローブが検出されると、プローブはRunウィンドウの左側の画像イメージ表示部に緑色で強調表示されます。

- (4) Runウィンドウの左上コーナーにあるRunウィンドウコントロールメニューボックスをダブルクリックしてください。この操作によって、Runウィンドウが閉じます。



"main"実行後のRunウィンドウの表示例


### 10.1.8 可動カメラでロボットをキャリブレーションするには

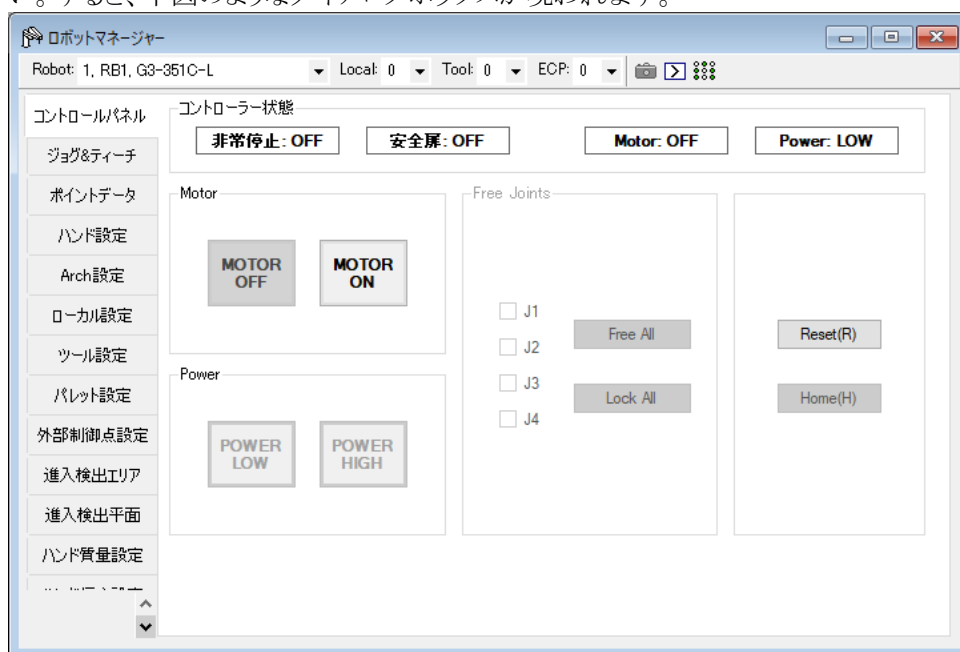
ここでは、水平多関節型ロボットの第2関節に搭載されている可動カメラでロボットをキャリブレーションする方法について説明します。

ビジョンシステムは、キャリブレーションを行うことにより、ロボット座標系とカメラ座標系を自動計算し、座標変換することができるようになります。キャリブレーションが完了すると、ビジョンシステムを使って、ロボットがピックアップする移動先のパーツを検出することができます。

可動カメラでロボットのキャリブレーションを開始する前に、モーターの励磁やツール値の設定などのロボット側の準備が必要です。

#### Step 1: モーターの励磁

- (1) EPSON RC+ 7.0ツールバー -  <ロボットマネージャー>ボタンをクリックしてください。すると、下図のようなダイアログボックスが現われます。



ロボットコントロールパネル

- (2) [ロボットコントロールパネル]の[MOTOR]グループで、<MOTOR ON>ボタンをクリックしてください。
- (3) メッセージボックスには、「モーターパワーをオンします。よろしいですか?」と表示されます。<はい>ボタンをクリックしてください。ロボットの各軸モーターが励磁されます。

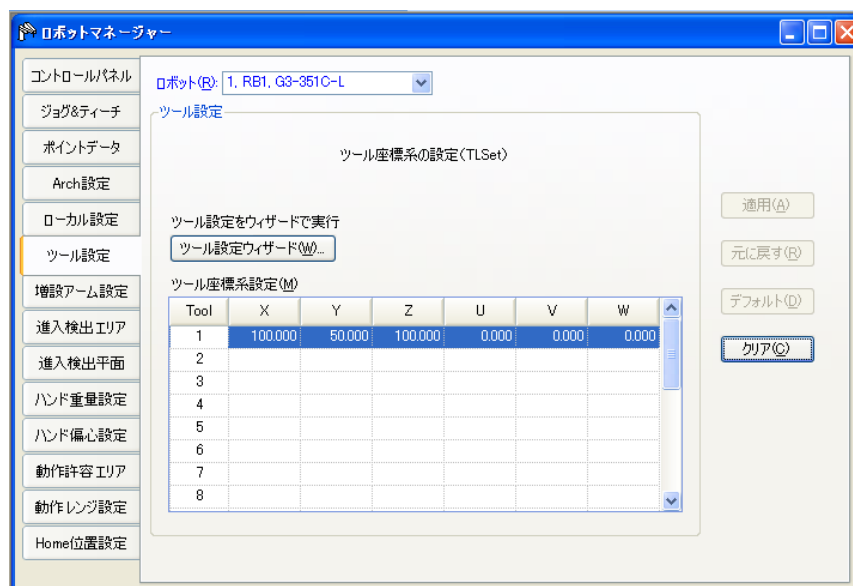
## Step 2: ツール設定タブを使ったツール値の設定

ビジョンシステムを使ったガイダンススペースのアプリケーションにロボットを使用するには、アーム端のツール(グリッパやロッドなど)の正確な位置計測が必要になります。第3軸(J3軸)中央からオフセットされたツールと、第3軸(J3軸)中央に添って装着されているツールがあります。しかし、どれほど注意深くツールを装着しても、第3軸(J3軸)の中央からのオフセットは生じます。したがって、このずれを調整するために、SPEL+言語のツール機能を使う必要があります。

ロボットガイダンスにVision Guideを使用するには、ツール機能を充分に理解しておく必要があります。ツール設定ウィザードを使用して、キャリブレーションロッド(グリッパ)のツールを作成します。

- (1) [ツール設定]タブを選択します。
- (2) <ツール設定ウィザード>ボタンをクリックします。
- (3) ツール設定ウィザードの基準点#1 は、ロッド、またはクリッパが正確にワッシャー(紙に描かれたワッシャー)の中央に位置するようにロボットをジョグし、ティーチします。ロボット、またはグリッパを紙に近づけてジョグする必要がありますが、紙にくっつけてはいけません。(紙とロボットの距離は 5~10 mm)これはロッド、またはグリッパが正確にワッシャーの中央に位置するようにするためです。
- (4) ツール設定ウィザードの基準点#2 は、U 軸を約 180°ジョグした後、ロボット、またはグリッパをワッシャーの中央に位置するようにロボットをジョグしティーチします。
- (5) <完了>ボタンをクリックして新しいツール定義を適用し、ツール設定ウィザードを閉じます。ツール 1 のパラメーターを定義しました。

ツール 1 の新しいツール定義は、下図のようにチュートリアルに表示されます。



ツール設定タブ(ツール定義用)



### Step 3: 新規に定義されたツールのテスト

[ジョグ&ティーチ]タブを選択します。ロボットはジョグしていないので、ロッドはまだ、ターゲットポジション(ワッシャー中央の真上)に位置しているはずです。

- (1) 問題が生じた場合に備えて、ロッドを作業面から上に離しておきます。+Zのジョグボタンを押して、第3軸(J3軸)を10~15 mmほど上げてください。
- (2) ジョグ&ティーチウィンドウの右上コーナーにある[Tool]リストで、下矢印をクリックしてください。
- (3) “1”をクリックして、ツール1を選択してください。
- (4) +U、または-Uのジョグボタンを押してみて、U axis第4軸(J4軸)が回転してもロッドの先端がずれないことを確認してください。(これは、第3軸(J3軸)の中心からずれているツールによく見られます。第3軸(J3軸)からまっすぐ下にあるロッドをお使いの場合は、これを見るのは難しいかもしれません。)
- (5) 第4軸(J4軸)が動いているときには、ターゲットポジションからロッドがいくらか動くのが見えますが、ジョグのステップ実行が完了すると、ロッドの先端はターゲットポジションへ戻ります。
- (6) ツールが適切に動作しないように見える場合は、ステップ2の最初に戻って、もう一度やり直してください。
- (7) [ロボットマネージャー]を閉じてください。ツールの定義とテストはこれで終了です。

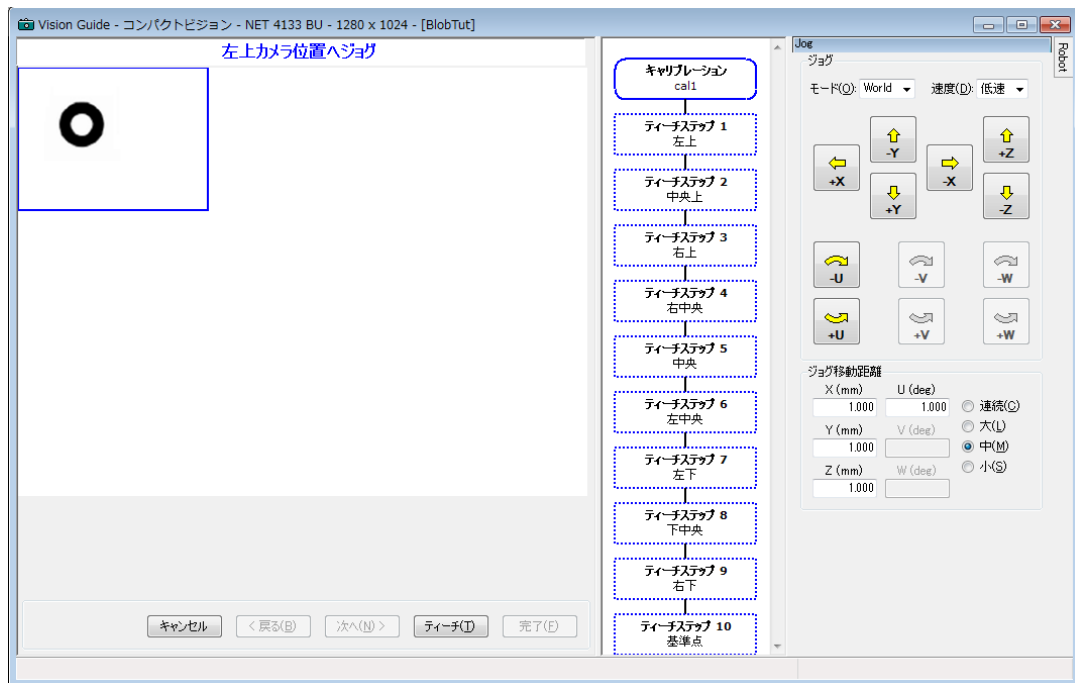
### Step 4: カメラキャリブレーションプロセスの開始

以上で、可動カメラでロボットをキャリブレーションする準備ができました。

- (1) EPSON RC+ 7.0ツールバー -  <ビジョン>ボタンをクリックして、ビジョンウィンドウを画面中央に開いてください。
- (2) Vision Guideツールバー -  <キャリブレーション>ボタンをクリックしてください。  
[新規キャリブレーション]ダイアログが表示されます。
- (3) 新規キャリブレーション名を入力してください。ここでは、“downcal”という名称を入力して、<OK>ボタンをクリックしてください。
- (4) キャリブレーションウィンドウのプロパティリストで、CameraOrientationプロパティを“Mobile J2”に変更してください。
- (5) RobotToolプロパティを“1”に変更してください。これによりキャリブレーションの基準点のティーチングに使うツール1を選択します。
- (6) TargetSequenceプロパティを“blobtut”に設定してください。キャリブレーションでのパーツ検出には、このblobtutシーケンスを使うことになります。

## Step 5: キャリブレーションポイントのティーチング

- (1) 画像表示の下の方にある<ポイントティーチング>ボタンをクリックしてください。
- (2) 下図に示すように、[Vision Guide]ウィンドウはポイントティーチングモードに切り替わります。[Vision Guide]ウィンドウの上に、下図に示すような「左上カメラ位置へジョグ」というメッセージが表示されます。これは、[Vision Guide]ウィンドウの画像イメージ表示部の左上コーナーにワッシャーが表示されるようロボット(とカメラ)をジョグする、ということです。下図に、ロボットをジョグする位置の目安を示します。これがキャリブレーションに必要な9つのカメラ位置の第1カメラ位置です。



## キャリブレーションのためのカメラ位置ティーチング

- (3) 第1カメラ位置を教示します。  
: 左上の位置にワッシャーが見えるようにロボットをジョグします。
- (4) カメラが位置決めできたら、[Vision Guide]ウィンドウの<ティーチ>ボタンをクリックしてください。
- (5) 第2カメラ位置を教示します。  
: 中央上の位置にワッシャーが見えるようにロボットをジョグします。この位置は第1カメラの右側にあたります。
- (6) カメラが位置決めできたら、[Vision Guide]ウィンドウの<ティーチ>ボタンをクリックしてください。
- (7) 第3カメラ位置を教示します。  
: 右上の位置にワッシャーが見えるようにロボットをジョグします。この位置は第2カメラの右側にあたります。
- (8) カメラが位置決めできたら、[Vision Guide]ウィンドウの<ティーチ>ボタンをクリックしてください。
- (9) 第4カメラ位置を教示します。  
: 中央右の位置にワッシャーが見えるようにロボットをジョグします。この位置は、第3カメラ位置の下にあたります。以後、カメラ位置をジグザクの順番で教示します。



(10)カメラが位置決めできたら、[Vision Guide]ウィンドウの<ティーチ>ボタンをクリックしてください。

(11)第5~第9カメラ位置を教示します。

: 続けて、第5~9カメラ位置のそれぞれについて、[Vision Guide]ウィンドウの下で指定された位置にロボットをジョグし、キャリブレーションポイントを教示してください。

[Vision Guide]ウィンドウに表示されるカメラ位置は次のとおりです。

- 5 - 中央
- 6 - 左中央
- 7 - 左下
- 8 - 中央下
- 9 - 右下

(12)次に、「固定基準位置へジョグ」というメッセージが、画面上に表示されます。これは、ロッドがターゲットの中央にくるように、ロボットをジョグすることを意味しています。ここではロッドをターゲットに合わせているので、ビデオディスプレイは無視してください。ジョグボタンを使って、ロッドが正確にワッシャーの中央にくるようにジョグしてください。

(13)ロッドが正確にワッシャーの中央にきたら、[Vision Guide]ウィンドウの下にある<ティーチ>ボタンをクリックしてください。

(14)ロッドの位置は非常に低くなっているので、キャリブレーション中に他のオブジェクトに干渉する可能性があります。+Z方向にジョグして、作業面から十分に高い位置に離しておいてください。このZ高さは、これから教示する9つのキャリブレーションポイントに使用されます。

(15)最後のカメラ位置の教示が終了すると、「すべてのキャリブレーションポイントが教示されました。」と画面上に表示されます。<完了>ボタンをクリックして、次のステップに進んでください。

以上で、キャリブレーションに必要なポイントティーチングは完了しました。

## Step 6: カメラキャリブレーションの最終ステップ

カメラキャリブレーションを完了するのに必要な最終ステップを次に示します。

(1) [キャリブレーション]パネルの<キャリブレーション実行>ボタンをクリックしてください。

(2) メッセージボックスには、「注意:ロボットはキャリブレーション中、SPEED10、ACCEL10で動作します。よろしいですか?」と表示されます。<はい>ボタンをクリックして、キャリブレーションを続けてください。

ロボットは、キャリブレーション中、9つの各々のカメラ位置を2回移動します。ロボットの動作中にキャリブレーションを中断したい場合は、[キャリブレーション実行中]ダイアログの下にある<中断>ボタンをクリックしてください。

(3) キャリブレーションサイクルが完了すると、[キャリブレーション完了]ダイアログが表示され、キャリブレーションの結果のデータを表示します。データを確認して、<OK>ボタンをクリックしてください。

(4) キャリブレーションの結果のデータは、キャリブレーションウィンドウのリザルトリストに表示されています。

### Step 7: "Downcal"キャリブレーションのBlobtutシーケンスへの割り当て

“downcal”キャリブレーションはすでに作成されているので、次に、このキャリブレーションをビジョンシーケンス(blobtut)に割りあてする必要があります。この割りあてによって、blobtutシーケンスでは、ロボット座標系またはカメラ座標系の値で結果を計算することができるようになります。

- (1) [Vision Guide]ウィンドウで、シーケンスツリーからビジョンシーケンス“blobtut”をクリックし、シーケンスウィンドウを前面に表示させてください。
- (2) Calibrationプロパティを“downcal”に設定してください。  
設定方法は、まず、Calibrationプロパティの値フィールドをクリックし、次に下向き矢印をクリックし、最後に“downcal”という名称のキャリブレーションをクリックします。このキャリブレーションは、前述のステップで作成したものです。
- (3) シーケンス、キャリブレーションツリーの右側に表示されている<Jog>ボタンをクリックし、ジョグタブを表示させてください。
- (4) ジョグボタンを使って、カメラをワッシャーの上に位置決めし、ワッシャーが画像イメージ表示部内に見えるようにしてください。
- (5) シーケンスツリーから“washer”オブジェクトをクリックし、オブジェクトウィンドウを前面に表示させてください。
- (6) “washer”オブジェクトを実行します。<オブジェクトの実行>ボタンをクリックしてください。パーツが検出されたら、リザルトリストを見てください。このとき、CameraX, CameraY, RobotX, RobotY, RobotUのリザルトの表示は、“nocal”リザルトになっていません。その代わりに、ロボット座標系またはカメラ座標系での座標位置データが表示されています。

#### 10.1.9 ビジョンガイダンスで使用するポイントの教示

ワッシャーのピックアップ位置のZ高さ、カメラがワッシャーの画像をとる位置、このチュートリアルで作業を開始するための安全な位置を定義するいくつかのポイントを教示する必要があります。

#### Step 1: 「カメラショット」位置の定義

ロボットをパーツに向かって動作させ、ワッシャーが画面上の画像イメージ表示部に見えるようにしてください。ワッシャーが画像イメージ表示部の中央に見え、サーチウィンドウの端に近づきすぎない位置にするのがよいでしょう。washerオブジェクトを実行したばかりなので、カメラはワッシャーを的確にとらえる位置にあるはずです。

- (1) EPSON RC+ 7.0メニューバー - <ツール> - <ロボットマネージャー> - [ジョグ&ティーチ]タブを選択します。
- (2) カメラショット位置をツール1を使って教示するために、Tool:というラベルのドロップダウンリストをチェックして、ツールが1に設定されているかどうか確認してください。ツールが“1”に設定されていない場合は、ドロップダウンリストの矢印をクリックして、ツールを“1”に設定してください。
- (3) ポイント#フィールドのカレントポイントが“P0”であることを確認してください。ポイントが“P0”になっていない時は、ポイントフィールドで“P0”を選択してください。

- (4) ジョグ&ティーチウィンドウの<ティーチング>ボタンをクリックしてください。ラベル名をつけるか、メッセージが表示されます。“camshot”と入力します。この操作によって、“camshot”位置が教示されます。

## Step 2: ワッシャーから離れた安全位置の定義

プログラムのスタート時に移動する安全位置として、ワッシャーから離れたところにポイント位置を教示する必要があります。

- (1) ポイントフィールドで“P1”を選択してください。
- (2) ロボットのZ軸を上方にジョグし、X軸とY軸でロボットを安全位置に位置決めするようにジョグしてください。ここが、プログラムの開始位置となります。ロボットは、ワッシャーに移動する前に、必ずこの位置に移動します。
- (3) [ジョグ&ティーチ]パネルの<ティーチング>ボタンをクリックしてください。ラベルに“safep1”を入力します。この操作によって、“safep1”位置が教示されます。

## Step 3: 「ワッシャー」ピックアップ位置として適切なZ高さの設定

ワッシャーの図面へロボットを動作する場合は不要ですが、本物のワッシャーをピックアップするアプリケーションを行う場合は、ワッシャーのピックアップ位置としてのZ高さを的確に設定する必要があります。ここでは、第3軸(J3軸)にグリッパーを装着してあることを前提に説明します。




- ロボット動作時のZ高さの決定には、十分注意してください。  
Z高さをまちがえると、ロボットのツール(グリッパー)や、周辺機器に重大な損傷を与える可能性があります。

- (1) EPSON RC+ 7.0メニューバー - <ツール> - <ロボットマネージャー> - [ジョグ&ティーチ]タブを選択します。
- (2) まず、ジョグボタンを使って、グリッパーをワッシャーの上に位置決めし、Z高さをワッシャーから5~10 mmの高さに位置決めしてください。グリッパーをワッシャーにぶつけないように慎重に行ってください。
- (3) グリッパーを少しずつ下げ、ワッシャーをつかめる位置にきたら、そのときのZ座標の値を控えておいてください。このZ座標値をプログラム中で使用し、グリッパーをこの高さに移動させるようにします。
- (4) ポイントフィールドで“P2”を選択してください。
- (5) ジョグ&ティーチウィンドウの<ティーチング>ボタンをクリックしてください。ラベル名に“washpos”を入力します。この操作によって、“washpos”の初期位置が教示されます。(しかし、Visionシステムは、新しいXY位置を計算し、それからこのポイントに移動するために使われます。また、現在のZ座標位置に基づいたプログラムに、固定のZ高さを設定します。)

## 10.1.10 ビジョンとロボットを使ってパーツへの移動をする方法

最後の手順として残っているのは、ビジョンシステムとロボットと一緒に動作してワッシャーの位置を検出し、その位置まで移動するように、プログラムを修正することです。

## Step 1: SPEL+プログラムに必要な修正

- (1) EPSON RC+ 7.0ツールバー -  <ファイルを開く>ボタンをクリックします。
- (2) MAIN.PRГプログラム以外に作成済みのプログラムはないので、MAIN.PRГプログラムファイルが反転表示されているはずです。ダイアログの左下にある<開く>ボタンをクリックして、MAIN.PRГプログラムファイルを開いてください。次に示すように、本チュートリアルで以前に実行したプログラムが表示されます。

```
Function main
  Real area
  Boolean found

  VRun blobtut
  VGet blobtut.washer.found, found

  If found = TRUE Then
    VGet blobtut.washer.area, area
    Print "The washer was found!"
    Print "The washer area is:", area, "Pixels"
  Else
    Print "The washer was not found!"
  EndIf
Fend
```

- (3) 次のページに示すように、プログラムを修正します。

## NOTE



SPEL+言語では、コメント表示をアポストロフィ(')で示します。アポストロフィに続いて示される文字は、すべてコメントであって、プログラムの実行には必要とされない部分です。つまり、アポストロフィで始まる文章は、省略することができます。

## Function main

```

'*****
' 以下に重要なステートメントがあります: *
' 「ビジョンガイダンスで使用するポイントの教示」の手順3で記録 *
' したZ座標値を使用してください。 *
' 下記の“xx”に、記録したZ座標値(負数)を代入してください。 *
'*****
#define ZHeight -xx

Real area, x, y, u
Boolean found
Integer answer
String msg$, answer$

Power Low          '低速でロボットを動かし、加速する
Tool 1             '位置決めにツール1を使用する
Jump safept        '安全開始位置までロボットを移動する

Do                 'オペレーターが停止するまでループし続ける
  Jump camshot      'ロボットを動かし、撮像する
  VRun blobtut       'ビジョンシーケンス(blobtut)を起動する
  VGet blobtut.washer.RobotXYU, found, x, y, u

  If found = True Then
    VGet blobtut.washer.area, area
    Print "The washer was found!"
    Print "The washer area is: ", area, "Pixels"
    washpos = XY(x, y, ZHeight, u) 'Set pos to move to
    Jump washpos
    msg$ = "The washer was found!"
  Else
    msg$ = "The washer was not found!"
  EndIf
  msg$ = msg$ + CRLF + "Run another cycle (Y/N)?"
  Print msg$
  Input answer$
  If Ucase$(answer$) <> "Y" Then
    Exit Do
  EndIf
Loop
Fend

```

### Step 2: ワッシャーの検出とワッシャー位置への動作を行うプログラムの実行

- (1) EPSON RC+ 7.0ツールバー - <Runウィンドウ>ボタンをクリックしてください。プログラムがコンパイルされて、Runウィンドウが表示されます。
- (2) Runウィンドウの<Start>ボタンをクリックしてください。
- (3) プログラムは、ワッシャーを検出し、ロボットをその位置へ動作させます。ワッシャーが検出できたら、ワッシャーをわずかに動かしてください。もう一回サイクルを実行しますか、というダイアログが表示されますので、<はい>ボタンをクリックしてください。ワッシャーが検出できない場合は、もう一度実行しますか、というダイアログが表示されます。  
どちらのダイアログでも、<いいえ>ボタンをクリックするとプログラムの実行を終了することができます。

## 11. SPEL+でVision Guide 7.0を使う方法

### 11.1 概要

Vision Guide 7.0では、プログラミングを最小限にとどめながら、SPEL+言語から、ビジョンシーケンス、ビジョンオブジェクトとキャリブレーションを操作できるようにしています。

ほとんどのアプリケーションでは、ポイント クリック インタフェースで作成されたビジョンシーケンスによって、大抵の作業が実行できます。SPEL+からシーケンスを実行し、その実行結果を利用してロボットガイダンスやパーツ検査等を、簡単に行うことができます。

このようなタイプの設計によって、アプリケーションエンジニアは、実際にアプリケーションを設計する前に、プログラミングに必要なシーケンスを作成することができます。したがって、作業の見積りをする前に、視野や精度等の確認ができ、設計時には、実際のプロジェクトで作成されたシーケンスを使用することができることになります。

Vision Guide 7.0は、ロボットコントローラーで実行されるSPEL+言語のコマンドによってサポートされています。ポイント クリック インタフェースでビジョンシーケンスを作成したら、これらのシーケンスをSPEL+プログラムで実行して、リザルトを得ることができます。また、ビジョンシーケンスとオブジェクトのプロパティを表示したり、設定したりすることもできます。

本章では、SPEL+プログラムでのVision Guide 7.0コマンドの使用方法について説明します。

### 11.2 Vision Guide 7.0 SPEL+ コマンド

SPEL+プログラムで利用できるビジョンコマンドの概要について、次に説明します。

#### 基本ビジョンコマンド

VRun	カレントプロジェクトのビジョンシーケンスを実行
VGet	シーケンスまたはオブジェクトのリザルトやプロパティを取得し、1つあるいはそれ以上の変数に格納
VSet	シーケンスまたはオブジェクトに、プロパティの値を設定

#### **ビジョンコマンドでプログラミングをする前に**

ビジョンコマンドでプログラミングを開始する前に、EPSON RC+ 7.0プロジェクトにビジョンシーケンスを作成する必要があります。

SPEL+プログラムで使用するビジョンコマンドは、プログラム内で作成されたこれらのシーケンスとオブジェクトに適用されます。

### 11.3 SPEL+からVRunコマンドでビジョンシーケンスを実行する方法

VRunコマンドを使用して、SPEL+プログラムからカレントプロジェクトのビジョンシーケンスを実行することができます。

```
Function VisionTest
```

```
    VRun seq1
End
```

上のシンプルなプログラムは、“seq1”というシーケンスを実行して、プログラムをどこから始めるかにより、RUN ウィンドウかオペレーターウィンドウに、グラフィックリザルトを表示します。

VRunの動作は、RuntimeAcquireシーケンスプロパティにより異なります。このプロパティは、シーケンスを処理する前に画像を撮影するかしないか、または、トリガー入力を使用することを定義します。次の表は、RuntimeAcquire設定と、VRunの関係を示しています。

RuntimeAcquire	VRun Actions	使用
0 - None	画像を撮影しません。 前回の画像を使います。 シーケンスをステップごとに実行します。	同じ画像で複数のシーケンスを実行 通常、RuntimeAcquire設定が “Stationary”なら、最初のシーケンス が画像を撮影します。 その場合、残りのシーケンスは画像 を撮影しません。
1 - Stationary	新規画像を取得します。 シーケンスをステップごとに実行します。	画像を撮影して、シーケンスを実行 (デフォルト) これがVRunを使う最も一般的な方法です。
2 - Strobed	トリガー入力を待ちます。 新規画像を取得します。 シーケンスをステップごとに実行します。	トリガー入力で使用 VRun は直ちに帰り、システムはトリ ガー入力を待ち、画像を撮影して シーケンスを実行します。

シーケンスプロパティのAsyncModeがTrueの場合、VRunはカメラの露光後に帰り、バックグラウンドで画像取得を継続・完了し、シーケンスを処理します。VGetなど、同じシーケンスの次のビジョンコマンドは実行が完了するまで自動的に待機します。

RuntimeAcquireを“Strobe”に設定した場合、VRunはトリガーを待機状態にし、戻ります。システムはバックグラウンドでトリガー入力を待ち、画像を撮影してシーケンスを処理します。

シーケンスからリザルトを取得する前に撮影が完了するまで待機させたい場合は、AcquireStateシーケンスリザルトを使用します。



```

Function VisionTest
    Integer state
    Boolean passed
    VRun strobedSequence
    ' 画像が撮像されるまで待機
Do
    VGet strobedSequence.AcquireState, state
Loop Until state = 3
    ' リザルトを取得
VGet strobedSequence.AllPassed, passed
Fend

```

RuntimeAcquireがStrobeの場合にVRunを実行し、「AcquireState =3」を待たずに次のビジョンコマンド(VGetなど)を実行すると、2番目のコマンドはトリガーの受信とシーケンスの実行を待ちます。トリガーが届かない場合、SPEL+ タスクは中止されます。

複数のSPEL+ タスクの実行では、同じカメラを使用しない場合、画像の取得とシーケンスの処理は並行して行われます。

複数のSPEL+タスクで1つのカメラを使い、RuntimeAcquireが「Strobe」に設定されている場合、SyncLockやSyncUnlockを用いて1度に1つのシーケンスが処理されるようにする必要があります。次の例では、シーケンス seq1とseq2はどちらもcamera 1を使用しています。

```

Function visTask1
    Integer state
Do
    SyncLock 1 ' camera 1 へのアクセスをロック
    VRun seq1
Do
    VGet seq1.AcquireState, state
Loop Until state = 3
    VGet <some results here>
    SyncUnlock 1 ' camera 1 へのアクセスをアンロック
Fend

Function visTask2
    Integer state
Do
    SyncLock 1 ' camera 1 へのアクセスをロック
    VRun seq2
Do
    VGet seq2.AcquireState, state
Loop Until state = 3
    VGet <some results here>
    SyncUnlock 1 ' camera 1 へのアクセスをアンロック
Fend

```

SyncLockおよびSyncUnlockの詳細は、「SPEL+ランゲージリファレンス」を参照してください。

注意:

ac2500-14gm/gcのGigEカメラを“Strobe”モード(外部トリガーモード)で使用する場合、外部にストロボ照明が必要になります。

ストロボ照明を使用しない場合、カメラはローリングシャッターモードで動作するため、移動物体を正しく認識できません。

## 11.4 VGet, VSetコマンドで SPEL+のリザルトやプロパティにアクセスする方法

VGetコマンドとVSetコマンドは、SPEL+プログラム中のシーケンスまたはオブジェクトのプロパティを確認したり、設定したりするのに使用することができます。

VGetコマンドは、リザルトを取り込むのに使用されます。例えば、サーチウィンドウのサイズや位置、許容パラメーター、カメラゲイン、指定最大領域を変更することができます。

Vision Guide 7.0のポイント クリック インタフェースからアクセスできるプロパティやリザルトのすべては、SPEL+プログラムからもアクセスすることができます。また、中には、複数リザルトを設定したり返したりするために、VGetコマンドとVSetコマンドを使わなければアクセスできない特殊なプロパティ(SearchWin, RobotXYU, ModelWin)があります。

VGetコマンドとVSetコマンドは、共通のシンタックスを使用しています。

これらのコマンドでは、まず、シーケンス名を指定しなければなりません。さらに、ビジョンオブジェクトのプロパティやリザルトにアクセスするため、ビジョンシーケンス名の後にビジョンオブジェクト名を続けなければなりません。

シーケンス名、オブジェクト名、プロパティ名、リザルト名を区別するには、ピリオドを使用します。複数リザルトを使う場合は、リザルト名の後に括弧に入れたリザルト番号を追加して、指定するリザルトを選択してください。

シーケンスのプロパティやリザルトには、次のシンタックスを使用しています。

**VGet** *seqName.propName, var* 'put property value in variable

**VSet** *seqName.propName, value* 'set property to value

オブジェクトのプロパティやリザルトには、次のシンタックスを使用しています。

**VGet** *seqName.objName.resultName, var*

**VGet** *seqName.objName.propertyName, var*

**VSet** *seqName.objName.propertyName, value*

オブジェクトの複数リザルトには、次のシンタックスを使用しています。

**VGet** *seqName.objName.resultName(resultnum), var*

また、シーケンス名とオブジェクト名は、文字列変数にすることもできます。

詳細は、「11.5 シーケンス名とオブジェクト名に変数を使用する方法」を参照してください。

### 11.4.1 VGetの使用方法

VGetコマンドは、プロパティやリザルトを取り込み、SPEL+変数に置換します。プログラムには、VGetからの値を受け取るのに適切なデータ型の変数を供給しなければなりません。

次に示すのは、SPEL+プログラムでのVGetの使用方法的例です。

```
Function Inspect
' Run the vision sequence
VRun InspectPart
Integer i, numberFound
Real area
VGet inspPart.Part1.NumberFound, numberFound
For i = 1 to numberFound
' Loop through each item that was found
' Get the area of the blob result
VGet inspPart.Part1.Area(i), area
Print "Area of result ", i, " is ", area
Next i
Fend
```

### 11.4.2 VSetの使用方法

VSetは、ランタイムのプロパティの値を設定します。このコマンドを使用すると、SPEL+プログラムからダイナミックにプロパティの設定を調整できます。

ほとんどの場合では、Vision Guideウィンドウからプロパティを設定でき、プロパティの設定を変更せずに、SPEL+プログラムからビジョンシーケンスを実行することができますが、プログラム実行中の変更が要求される場合に、VSetコマンドを使用します。

次に示すのは、SPEL+プログラムでのVSetの使用方法的例です。1番目のVSetは、シーケンスプロパティを設定するのに使用されていることに注意してください。2番目のVSetでは、シーケンスの実行前に、サーチウィンドウの位置とサイズを再定義するのに使用されるSearchWinというオブジェクトプロパティを設定しています。

```
Function findPart

' Set camera gain for sequence "findPart"
VSet findPart.CameraContrast, 32

' Set search window for object "part"
VSet findPart.part.SearchWin, 100, 100, 50, 50

' Run the sequenced
VRun findPart
Fend
```

## 11.5 シーケンス名とオブジェクト名に変数を使用する方法

VRun, VGet, VSetのコマンドのキャリブレーション名引数, シーケンス名引数, オブジェクト名引数には、文字列変数を使用することができます。

次の例では、どのビジョンシーケンスとビジョンオブジェクトで動作するかを指定するために、文字列変数 seq\$ と obj\$ を使用しています。

```
Function visTest
  #define PICKZ -100.0
  String  seq$, obj$
  Boolean found
  Real    x, y, u

  seq$ = "test"
  obj$ = "Blob01"
  VSet seq$.Camera, 1
  VSet seq$.Calibration, "CAMCAL1"
  VRun seq$
  VGet seq$.obj$.RobotXYU, found, x, y, u
  If found Then
    pick = XY(x, y, PICKZ, u)
    Jump pick
    On vacuum
    Wait .1
    Jump place
    Off vacuum
    Wait .1
  EndIf
  Jump park
Fend
```

また、VRun, VGet, VSetのコマンドのシーケンス名引数とオブジェクト名引数には、配列を使用することもできます。次の例を参照してください。

```
Function test
  String obj$(10)
  Integer count

  obj$(0) = "corr01"
  obj$(1) = "corr02"
  obj$(2) = "blob01"
  obj$(3) = "blob02"

  For count = 0 to 3
    VRun seqname
    VGet seqname.obj$(count).Found, found
  Next count
Fend
```

## 11.6 SPEL+でシーケンスリザルトを使用する方法

VRunコマンドでシーケンスを実行すると、SPEL+プログラムで使うことができるリザルトが複数できます。VGetコマンドは、リザルトとプロパティを読み取るのに使用されます。ロボットガイダンスに最もよく使用されるコマンドリザルトは、RobotXYUリザルトです。

```
Function getPart
  #define PICKZ -100.0
  Boolean found
  Real x, y, u

  VRun findPart
  VGet findPart.corr01.RobotXYU, found, x, y, u
  If found Then
    pick = XY(x, y, PICKZ, 0)
    Jump pick
  EndIf
Fend
```

上記の例で、ビジョンシステムによって検出された位置にロボットを動作させる前に、検出されたかどうかのチェックをしていることに注意してください。ビジョンオブジェクトのリザルトを使用してロボットを動作させる前に、ビジョンオブジェクトが検出されていることを検証してください。

## 11.7 SPEL+言語から複数リザルトにアクセスする方法

CorrelationオブジェクトとBlobオブジェクトのようなVision Guideオブジェクトは、NumberToFindプロパティを使って1つのオブジェクトで複数の対象物を検出することができます。CurrentResultプロパティは、オブジェクトウィンドウでリザルトを確認する場合に、リザルトリストに表示するリザルトを設定するのに使います。また、リザルトとしてどのリザルトレコードを返すかを指定することができます。言い換えると、Blobオブジェクトから返される3番目のリザルトからAreaリザルトを得たい場合は、CurrentResultプロパティを3に設定しなければなりません。[オブジェクト]ウィンドウのプロパティリストを使ってCurrentResultプロパティを設定する方法については、すでに説明したとおりです。ここでは、SPEL+言語から複数リザルトにアクセスする方法について説明します。

SPEL+言語から複数リザルトへのアクセスは、得られるResultの隣に添字番号をもつCurrentResultプロパティが参照をつけられる配列の型のよう、Resultを扱います。下の1番目の例は、3番目のAreaリザルトを得て、それをSPEL+言語から領域という変数に入れる方法を示します。

```
VGet seqname.objname.Area(3), area
```

下の2番目の例は、同じく3番目のAreaリザルトを得ますが、今回はそれをArea()と呼ばれる配列の第3要素の値として代入する方法を示します。

```
VGet seqname.objname.Area(3), area(3)
```

変数名はまた、上の2番目の例のように固定された要素としてよりも、配列の要素を表すのに使われます。“var”という変数はAreaリザルトの添字として使われることに注目してください。

```
VGet seqname.objname.Area(var), area(var)
```

4番目の例は、1つのビジョンオブジェクトを使って、パーツのようなまとまり(10までとします)を検出すると仮定してください。ロボットでそれらのパーツ(例えばペンだとして)を選ぶとすると、検出された各パーツの座標値を表すために、X座標, Y座標, U座標を配列変数に入れる必要があります。下のコードは、これらの座標をRobotXYUリザルトから取り出して、あとでロボットを動かすのに使われるX配列, Y配列, U配列の中に座標を入れることができます。

```
Function test
  Boolean found(10)
  Integer numfound, i
  Real x(10), y(10), u(10)

  Jump camshot      'move camera into position snap shot

  VRun seq01         'run the vision sequence to find the pens

  VGet seq01.blob01.NumberFound, numfound 'how many found

  For i = 1 to numfound 'get robot coords
    VGet seq01.blob01.RobotXYU(i), found(i), x(i), y(i), z(i)
  Next i

  'Add code for robot motion here.....
Fend
```

## 11.8 マルチタスクでビジョンコマンドを利用する方法

SPEL+プログラムで、複数のタスクからビジョンシーケンスの実行とリザルトの取り込みができます。SPEL+は、内部的に一度に1つの実行するビジョンコマンドを自動で処理します。複数のタスクがビジョンシステムを必要とする場合、コマンドは先着順で実行されます。1つのタスクがビジョンシステムを利用しているあいだ、他のタスクは、最初のタスクが現在のビジョンコマンドを終了するまでブロックします。(待ちます。)

異なるカメラを使用している異なるタスクから、同じシーケンスを実行することもあります。他のタスクを待つ間、1つのタスクでシーケンスを実行しリザルトを得たいとき、ビジョンシステムをロックするSyncLockとSyncUnlock命令を使用します。そして他のタスクにリリースします。

```
Function VisionTask(camera As Integer)
  Boolean found
  Do
    SyncLock 1 ' Lock vision just for this task
    VSet FindParts.Camera, camera
    VRun FindParts
    VGet FindParts.AllFound, found
    SyncUnlock 1 ' Unlock vision for other tasks
  Loop
Fend
```

## 11.9 ロボットでビジョンを使用する方法

ロボットガイダンスにビジョンシーケンスからのリザルトを使用する前に、そのシーケンスで使用するカメラをキャリブレーションする必要があります。詳細は、「7. ビジョンキャリブレーション」を参照してください。

キャリブレーションを必要とするシーケンスオブジェクトからリザルトを取得しようとしたとき、キャリブレーションがされていない場合は、実行時にエラーが発生します。

### 11.9.1 位置リザルト

キャリブレーションされたカメラを使用したビジョンシーケンス内のオブジェクトによって得た位置リザルトは、指定したローカル座標系にあります。他のロボット/ビジョンシステムとは違い、ピクセル座標からロボット座標への変換ファンクションコールを行うための更なるステップはありません。これは、Vision Guide 7.0により完全に処理されます。座標系は簡単に検索できます。

ビジョンシステムで検出された位置に、ロボットのハンドを正確に位置決めするには、ハンドにツールを定義する必要があります。次の項目の「ツールの定義」を参照してください。

ロボットのガイダンスには、次のリザルトを使用することができます。

**RobotXYU** 検出された状態でのX, Y, Uの座標を返す

**RobotToolXYU** 上向きカメラ時のみ

検出された状態でのロボットツールを定義するためのX, Y, Uの座標値を返す

**RobotX** X座標を返す

**RobotY** Y座標を返す

**RobotU** U座標を返す

**RobotPlacePos** 上向きカメラ時のみ

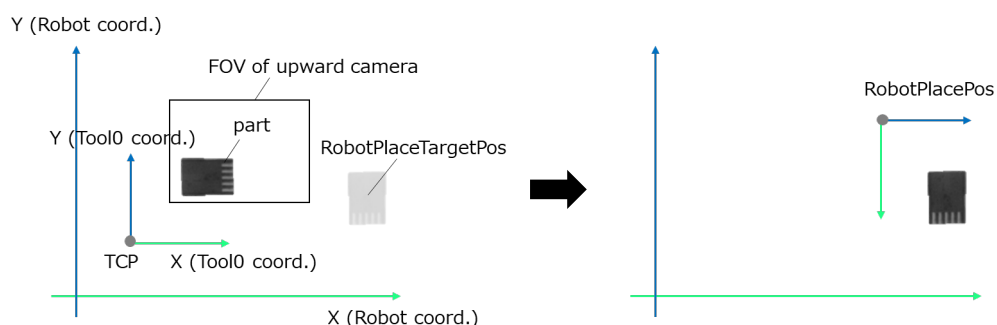
CalRobotPlacePosで登録したワーク配置位置 (RobotPlaceTargetPos) に、検出されたワークを配置するためのロボット位置をPoint型で返す

注意:

各オブジェクトにてCalRobotPlacePosを設定する必要があります。

U座標0°位置は、ロボットベースからまっすぐにのびた位置にあるY軸に沿っています。

RobotPlacePosについて、以下の図で説明を補足します。



この図では、作業面であるロボット座標系に上向きカメラが配置されており、CalRobotPlacePosによってワーク配置位置 (RobotPlaceTargetPos) が登録されています。



実際の作業では、まず、ピックされたワークの位置を、ビジョンシーケンスで検出 (VRun) します。その後、ワークを配置するときのロボット位置 (RobotPlacePos)を取得 (VGet) します。

図の右側は、RobotPlacePosに移動した後の状態です。RobotPlacePosに移動すると、ワークの位置姿勢をワーク配置位置と一致させることができます。

RobotPlacePosで得られるポイントは、姿勢フラグがデフォルトの値であるため、移動時に、必要に応じて補正が必要です。次の例では、ピックされたワークの位置を上向きカメラで検出し、関節移動量が少なくなるようにワーク配置位置の上まで移動する方法を示しています。

```
Function placePart
  'Move robot into position snapshot
  Go camshot

  VRun findPart
  VGet findPart.Blob01.RobotPlacePos, P100

  '-----
  ' When the robot is SCARA
  Double diffJ4 'Angle difference between RobotPlacePos and latch
                position(J4)
  diffJ4 = PAgl(P100, 4) - PAgl(LatchPos(WithoutToolArm), 4)
  If diffJ4 > 180 Then
    Go P100 -U(360) 'Joint4 will be the shortest movement
  ElseIf diffJ4 < -180 Then
    Go P100 +U(360) 'Joint4 will be the shortest movement
  Else
    Go P100
  EndIf

  '-----
  ' When the robot is 6-axis
  Go P100 LJM 4 '4: Joint6 will be the shortest movement
Fend
```

RobotPlacePosで得られるポイントが動作範囲外となり、エラー4007が発生することがあります。その場合は、ワーク配置位置を変更するか、RobotToolXYUを使用してください

### 11.9.2 ツールの定義

ロボットハンドのツールを定義するのは重要なことです。

ツールを定義することによって、ロボットはハンドがある場所の情報を得、位置情報のすべてが、TOOL0位置に関するものではなく、ハンドに関するものになります。

SPEL+でツールを定義するには、TLSETコマンドを使用してください。

次に、3つのツールオフセットの計算方法を示します。

## ロボットマネージャーツールウィザードでツールを定義する

ロボットマネージャーツールウィザードからツールを定義することができます。

ツールウィザードを使用するには、次の手順にしたがってください。

- (1) ロボットマネージャを開きます。
- (2) [ツール設定]タブを選択します。
- (3) <ツール設定ウィザード>ボタンをクリックします。
- (4) ウィザードに示される手順にしたがい、ツールを作成します。

## ツールオフセットの計算に、上向き固定カメラを使用する方法

ツールオフセットの計算に、上向き固定カメラを使用する方法を次に示します。ツールオフセットの特定には、RobotToolXYUリザルトを使用します。

ファンクションでは、まずツールの先端を位置決めするためのシーケンスを実行します。次に、VGet RobotToolXYUでツールオフセットを取得し、TLSetでツールを定義します。

```
Function DefineTool
  Boolean found
  Real xTool, yTool, uTool

  VRun findTip
  VGet findTip.tip.RobotToolXYU, found, xTool, yTool, uTool
  If found Then
    TLSet 1, XY(xTool, yTool, 0, 0)
  EndIf
Fend
```

## ツールオフセットの手動計算方法

下記のようなツールオフセットを計算するとき、軸を(SFREEを使用して)フリージョイント状態にすることはできません。つまり、ロボットの各軸を手で動かすことはできません。ジョグタブ、またはジョグ&ティーチウィンドウからジョグボタンでロボットをジョグしてください。

ツールオフセットの計算は、次の手順にしたがって行ってください。



- 垂直6軸型ロボットを使用している場合、下記のLocal命令を最初に実行し、以下の手順を実行してください。

```
> Local 1, Here
```

- 手順中の現在位置取得は、Local 1での位置を取得する必要があります。  
ジョグ&ティーチ画面を使用する場合、ジョグモードはLocalに、現在位置の表示モードはWorldに、Local番号は1に設定してください。  
コマンドウィンドウを使用する場合、下記のようにコマンドを実行することでLocal 1での位置を確認できます。

```
> Print Here@1
```



- (1) U軸を0°に位置決めします。
- (2) ツールを“0” (TOOL0)に設定します。
- (3) ロボットをジョグし、ハンドが基準点に重なるように注意深く合わせます。U軸の設定は変更しないでください。
- (4) 現在位置のX座標とY座標をX1とY1として控えておきます。
- (5) U軸を180°の位置にジョグします。
- (6) ロボットをジョグし、ハンドが基準点に重なるように注意深く合わせます。U軸の設定は変更しないでください。
- (7) 現在位置のX座標とY座標をX2とY2として控えておきます。
- (8) 下記の公式を使って、ツールオフセットを計算します。

$$xTool = (X2 - X1) / 2$$

$$yTool = (Y2 - Y1) / 2$$

- (9) ロボットマネージャー、あるいはコマンドウィンドウのツールページから、ツールを定義します。

```
TLSET 1, XY(xTool, yTool, 0, 0)
```

- (10) ツール設定をテストします。

現在ツールを前述の手順で定義したツールに設定してください。例えば、TOOL1と設定します。そして、ハンドを基準点に重なるようにジョグしてください。

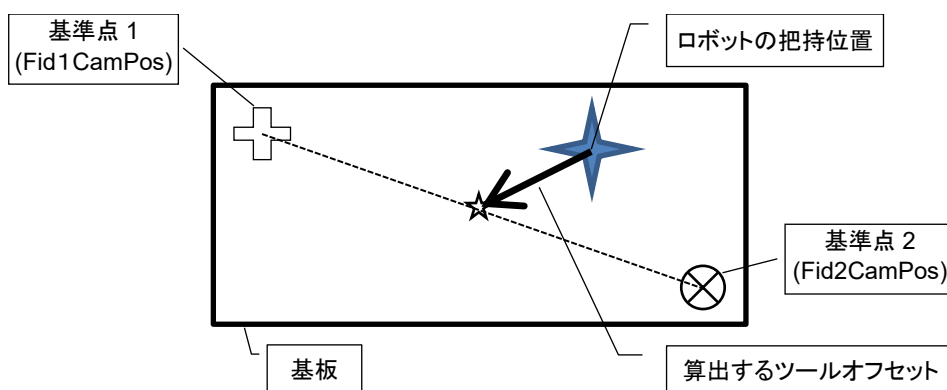
ここで、U軸をジョグします。このとき、ハンドは基準点に重なったままになっていなければなりません。

### 11.9.3 ロボットが把持している基板でのツール計算

この例で、Vision Guide 7.0は、ロボットが把持している基板のツール計算を行います。基板内の2つの基準点の座標からツールオフセットを算出します。これには上向き固定カメラが1台必要です。カメラをキャリブレーションした後、基板を置く位置を教示する必要があります。

置く位置の教示方法:

- (1) 基板をロボットでピックアップします。
- (2) ツール1を計算するために、ファンクション例CalBoardToolを呼び出します。
- (3) Tool 1に切り替えます。
- (4) 基板を置く位置にジョグします。
- (5) 置く位置を教示します。



```
Function CalcBoardTool As Boolean
  Boolean found
  Real x, y, theta
  Real toolX1, toolY1, toolU
  Real toolX2, toolY2

  CalcBoardTool = False
  Jump Fid1CamPos ' Locate fiducial 1 over camera
  VRun SearchFid1
  VGet SearchFid1.Corr01.RobotToolXYU, found, toolX1, toolY1, toolU
  If Not found Then
    Exit Function
  EndIf

  Jump Fid2CamPos ' Locate fiducial 2 over camera
  VRun SearchFid2
  VGet SearchFid2.Corr01.RobotToolXYU, found, toolX2, toolY2, toolU
  If Not found Then
    Exit Function
  EndIf
  x = (toolX1 + toolX2) / 2
  y = (toolY1 + toolY2) / 2
  theta = Atan2(toolX1 - toolX2, toolY1 - toolY2)
  toolU = RadToDeg(theta)
  TlSet 1, XY(x, y, 0, toolU)
  CalcBoardTool = True
Fend
```

#### 11.9.4 パレット検索用カメラの位置決め

J2 マウントカメラを使用している場合、カメラ撮像位置を制御するのは困難です。しかし、Arm を使用することで、カメラ撮像位置を簡単にパレット上で移動することができます。Arm の詳細は、「EPSON RC+ 7.0 SPEL+ ランゲージリファレンス」を参照してください。

**ヒント:**

Arm コマンドは SCARA 型ロボットでのみ使用可能です。

