

博士学位論文  
電子部品の形状認識に関する研究

2016 年

鬼頭秀一郎



# 目次

<b>第 1 章</b>	<b>はじめに</b> .....	<b>1</b>
<b>第 2 章</b>	<b>表面実装作業</b> .....	<b>6</b>
2.1	表面実装における作業.....	6
2.2	撮像システム.....	8
2.3	表面実装部品の種別.....	9
<b>第 3 章</b>	<b>コネクタ部品の特徴抽出法</b> .....	<b>14</b>
3.1	まえがき.....	14
3.2	従来の自動分類システムと問題点.....	15
3.3	信号解析基礎理論.....	17
3.3.1	フーリエ変換.....	18
3.3.2	離散フーリエ変換.....	19
3.3.3	高速フーリエ変換.....	21
3.4	特徴量抽出.....	23
3.4.1	リード群の判別領域の特定.....	24
3.4.2	リード群判別領域画像への FFT の適用.....	26
3.4.3	高次自己相関特徴 (HLAC).....	30
3.4.4	Local Binary Pattern (LBP).....	34
3.4.5	Local Ternary Pattern (LTP).....	34
3.5	サポートベクターマシン.....	35
3.6	評価実験.....	38
3.6.1	グレースケール画像とパワースペクトル画像に対する評価と考察.....	39
3.6.2	リード幅、ピッチに対するロバスト性評価と考察.....	40
3.6.3	特徴量抽出時間.....	43
3.6.4	自動分類器への組込み.....	43
3.6.5	提案手法を用いた自動分類器の評価と考察.....	44
3.7	まとめ.....	47
<b>第 4 章</b>	<b>リード部品形状データ自動生成</b> .....	<b>49</b>
4.1	まえがき.....	49
4.2	AdaBoost を用いたリード先端抽出.....	51
4.2.1	AdaBoost.....	51
4.2.2	カスケード構造.....	55
4.2.3	Haar-Like 特徴量.....	55
4.2.4	HOG 特徴量.....	58
4.2.5	リード検出実験.....	60
4.3	リード群領域の抽出法.....	64
4.3.1	平均振幅差関数.....	65
4.3.2	自己相関関数.....	66
4.3.3	二乗差関数.....	66

4.3.4	正規化二乗差関数.....	67
4.3.5	NSDF を用いた周期性の抽出.....	67
4.3.6	リード群の開始・終了座標の検出.....	71
4.3.7	リードの向きの決定.....	73
4.4	評価実験.....	75
4.4.1	リード群領域の抽出性能の評価と考察.....	75
4.4.2	Real AdaBoost と Haar-Like 特徴を用いたリード検出性能の評価と考察.....	76
4.4.3	提案手法によるリード検出実験と考察.....	78
4.5	まとめ.....	93
<b>第5章</b>	<b>むすび.....</b>	<b>95</b>
	<b>謝辞.....</b>	<b>97</b>
	<b>参考文献.....</b>	<b>98</b>
	<b>研究業績一覧.....</b>	<b>103</b>

# 第1章 はじめに

現在、我々を取り巻く環境には多くの電子機器が溢れている。パソコンや録画機器などはこれまでも広く使われているが、スマートフォンや携帯電話、デジタルカメラ等に代表される携帯型の電子機器の発展は特に著しく、多くの人がスマートフォンなどの端末を持ち歩いている[1]。これら携帯型の電子機器は、携帯性の観点から小型化・高機能化が求められており、それに伴い、電子部品の小型化・高機能化も進んでいる。今後、携帯側端末に実装される部品は更に小型化が進むことが予想されている[2]。高機能化に伴い部品点数が増加することは小型化とは反することになるため、小型化と高機能化を両立するためには、小型の部品に対する高精度・高密度な実装が要求される。2015年現在使用されている最も小型の部品は0402サイズ(0.4×0.2mm)の角チップ部品であり、更に小型の0201サイズ(0.25×0.125mm)のものも登場し[3][4]、これらの部品を部品間ギャップ0.05mmという挟隣接実装を行うため、より高密度かつ高精度な実装技術が必要とされている。この様に高精度・高密度で実装する要求があると同時に、最終製品の形状に合わせた部品形状の複雑化により実装難度が高くなる傾向にある。特にコネクタ部品はその傾向が強く、目的に応じて様々な形状で作成される。

これらの電子部品は正確には表面実装部品(Surface Mount Device、SMD)と呼ばれ、電子部品実装機(チップマウンタ・マウンタ。以降、マウンタと略す)を用いてプリント基板上へ装着される。マウンタには、先に述べたような高精度な実装機能が要求されると同時に、生産性の観点から実装速度や生産準備に必要な工数の少なさが重要な評価項目となる。近年、多量少品種の生産形態から、ユーザーのカスタマイズ等に応じ、少量多品種の生産も行われるようになった。少量多品種の生産を行う場合、生産の切り替え毎に必要な供給部品の変更や生産に必要な各種データの準備が必要となる。これらに必要な工数は、多量少品種生産でも少量多品種生産でも同じである。つまり目的に応じた特注部品が多い少量多品種の生産形態の現状においては、総生産時間に対する生産準備時間が相対的に大きくなる。そのため、生産準備にかかる時間を極力少なくすることが重要な課題である。

初期のマウンタでは、電子部品は画像処理ではなくメカニカルな機構を用いて位置決めされ、プリント基板上へ搭載されていた。そのため、部品の外形寸法のみが必要な部品形状データとなり、データの入力も時間のかかる作業では無かった。しかし電子部品は小型化や複雑化が進み、実装形態もスルーホールに対する挿入から表面実装の形態に移行していくこととなる。表面実装技術が主流となり、マウンタにも画像処理を用いた部品の位置決め、検査技術が開発された。画像処理による部品装着位置精度はメカニカル機構による位置決めに比べ飛躍的に向上したが、プリント基板へ搭載する部品が小型化・高機能化されるとともに、高精度実装のためより多くのデータ入力が必要となった。画像処理の手法も、初期の段階ではバックライト光源を用いて取得した部品画像に対し、その外形を捉える方法が開発されたが、BGA(Ball Grid Array)部品のよう下面にのみ電極が存在する部品では、はんだボール電極（バンプ）の欠けや欠損がバックライト光源では捉えられないという問題が発生したため、現在ではフロントライト光源を用いた部品画像の撮像とその画像を用いた部品の位置決め・検査を行う手法が主流となっている。

ここで、現在の代表的なマウンタの動作について説明する（図 1-1 参照）。マウンタに搭載した部品供給装置から供給された部品はノズルを用いて吸着され、カメラを用いて部品吸着姿勢を撮像する。撮像された部品画像は画像処理により吸着姿勢を計測（位置決め処理）し、同時に電極の欠損確認などの検査処理を行う。これらの位置決め処理、及び検査処理は、部品の形状や電極の配置パターンなどの情報（以降、形状データとする）を用いて処理を行う画像処理アルゴリズムが部品の種類別に用意されており、適切なアルゴリズムを選択する必要がある。

通常、生産時に使用される形状データと画像処理アルゴリズムは、部品の設計図データを元に基板生産前に作成、決定しておく必要がある。しかし受託生産を行う現場においては、機密保持等の理由により設計図データが入手できない場合や設計図データの代わりに現物が支給される場合がある。その場合、専門の人員が専用の撮像装置などを用いて部品画像を撮像し、画像から形状データの作成、及び画像処理アルゴリズムの決定を行う。それらの作業には専門性が必要となるため省人化の障害となる。画像処理アルゴリズムについては、部品の種類が決まれば一意に決定することが可能であるため、部品の種類を自動認識するシステムが求められている。

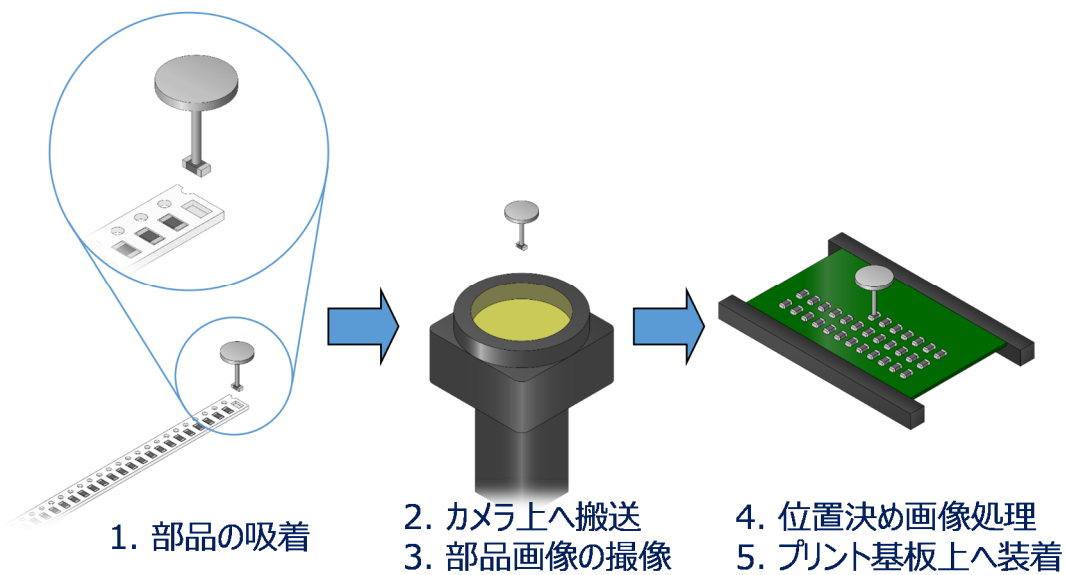


図 1-1 マウンタの動作

本論文では、部品画像からその部品の形状データ、及びマウンタで処理するための画像処理アルゴリズムを自動的に生成・選択することで生産準備にかかる作業時間の短縮を目的としている。IC 部品やコネクタ部品など、多くの電子部品にはリードと呼ばれる短冊状の電極が配置されている。特にコネクタ部品は最終製品の形状に合わせて設計されることも多く、新製品が設計される度に新たな形状データの入力が必要となる。そのため画像からリード固有の特徴を抽出し、部品の形状データを自動生成するシステムを構築することは、基板製造工程の自動化・省人化には有効な手段となる。

画像から何らかの特徴量を抽出する画像処理や検出手法は様々なものが提案されている。例えば人検出技術[5]-[9]や顔検出技術[10]はその代表例であり、効果的な報告がされている。これらの認識に関する研究は顔の構造、例えば目と頬の位置関係とその輝度差や、頭や肩から得られる勾配強度や勾配位置情報を元に顔か否か、あるいは人か否かを判別する。この手法では、人が持つ共通の特徴を抽出し機械学習器を用いて判別している。

一方で、与えられた画像に対しその中に含まれるものが何かを一般的な名称で認識するという、一般物体認識技術がある[11]。Bag of features に代表される一般物体認識技術[12]では、画像から得られる局所的な特徴量をベクトル量子化し、局所特徴量のヒストグラムをその画像の特徴として認識する手法である。この手法は顔検出や人検出とは異なり、物体の位置関係による特徴はヒ

ストグラム化する時点でキャンセルされる。そのため、飛行機やトラという一般的な画像の認識において、それらの向きやフレーム中の位置に対してロバストになるという利点がある。

近年では、深層学習（ディープラーニング）と呼ばれる多層構造のニューラルネットワークを用いた機械学習技術が注目を集めている。特に Deep Convolutional Neural Network (CNN) を用いた文字認識[13]や画像のカテゴリ認識[14]などに飛躍的な成果を上げている。この手法の利点はタスクに適した特徴量を入力画像から直接抽出できる点に有るが、大量の学習サンプルが必要となり、学習に時間がかかるという欠点がある。MNIST の手書き文字認識のタスク[15]は、0-9 までの手書きの数字画像を訓練データとして 60000 サンプル、テストデータとして 10000 サンプルが与えられ、認識能力を競うタスクである。ImageNet Large-scale Visual Recognition Challenge (ILSVRC) は、2010 年から開始された大規模画像認識コンペティションである。ここでは 1000 万件を超える画像が 400 以上のカテゴリに分類されており、それらを学習してどのカテゴリに分類される画像かを答えるタスクである。いずれにおいても CNN を使用したチームが好成績を残しているが、膨大な学習データを用いて学習することで効果的な結果を得ている。

ここで改めて電子部品の画像認識について着目する。近年のマウンタでは、部品の位置決め・検査処理にフロントライト光源を用いるようになり、画像から得られる情報量が増加した。画像からの情報を省人化や自動化に用いることができる状況は既に構築されている。電子部品がどのカテゴリに分類するかという問題に対して、深層学習を用いる方法では学習サンプルを何万、何十万と用意することとなり、現実的には困難と考えられる。そのため電子部品のもつ固有の特徴を抽出することが必要となる。またリードの形状をデータ化するにあたり、画像中からリードを 1 本ずつ検出する手法は、人検出や顔検出という特定物体認識を行うタスクとなる。電子部品は自然画像とは異なり、人工的な構造物であるため幾何学的なパターンが多く含まれており、その特徴を捉えることが認識には効果的と考えられる。そこで本論文では、電子部品のリードに共通な特徴を画像から抽出する手法と、特定物体認識技術を応用したリード形状データの自動生成手法を使用し、部品形状データの自動生成システムの構築を行う。

本論文の第 2 章では、本研究の前提となる表面実装における作業と現状のシステム概要について述べる。第 3 章では、部品画像から部品種別を分類する手



法について述べる。部品種別の自動分類法は既に提案されている手法があるが、その手法での問題点を整理し、部品中でリードが存在する可能性の高い領域と、その領域におけるリードの周期的な並びの特徴に注目した提案手法の有効性について述べる。第4章では、コネクタ部品におけるリード形状データの自動生成手法について述べる。特定物体認識手法である AdaBoost を用いたリードの先端検出法と、リードの周期的な並びに注目したリード領域の抽出手法を組み合わせた提案手法について、評価実験を通して有効性を示す。最後の第5章は全体のむすびである。

## 第2章 表面実装作業

本章では、研究の前提となる表面実装における現状について述べる。

### 2.1 表面実装における作業

現在、様々な電子製品に用いられているプリント基板の多くは、表面実装技術(SMT : Surface Mount Technology)を用いて作成されている。プリント基板の生産に用いられるマウンタは、生産効率向上のためにその速度(スループット)や部品供給の自動化など様々な改良が加えられている。近年、多量商品種の生産形態のみならず、ユーザーのカスタマイズに応じるため少量多品種の生産形態も多く、マウンタに対しても速度だけでなく多様な要求がある。

マウンタでの生産現場では、生産準備にかかる時間と実際に生産にかかる時間(実生産時間)があり、その合計が実際にプリント基板の生産時間(図 2-1)となる。生産準備は生産に必要なデータを用意するための時間であり、一般的に表 2-1 のような作業が発生する。

これらの生産準備作業は段取り作業とも呼ばれ、生産するプリント基板の種類が切り替わるたびに行わなくてはならないため、少量多品種の生産形態へ移行すればするほど相対的に段取り作業時間が増加し、生産効率を下げる原因となる。

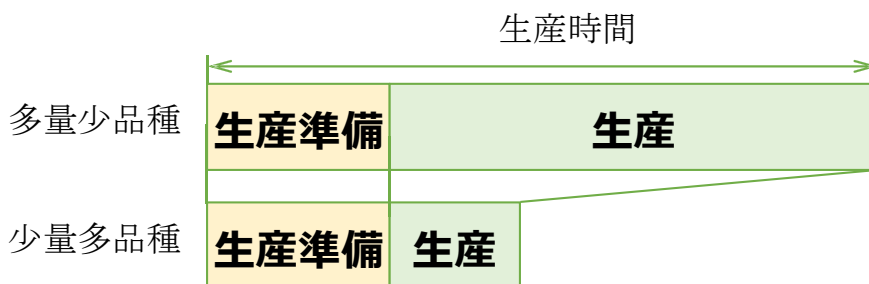


図 2-1 多量少品種生産と少量多品種生産

表 2-1 生産準備作業

作業	内容
装置レイアウト作業	マウンタの配置など
実装データ作成作業	表面実装で必要なデータの作成
供給部品の準備作業	プリント基板に搭載する部品の準備、及び供給装置の準備

表面実装部品（SMD：Surface Mount Device）をプリント基板上へ装着して最終製品を製造するためには、予め生産に関する様々なデータを作成した上で、マウンタに対してデータを登録しておかなくてはならない。生産には表 2-2 の様な様々なデータが必要となる。

これらのデータのうち、部品実装座標データはプリント基板上のどの位置へ対象部品を装着するかというデータであり、プリント基板の設計データから自動的に作成されることが一般的である。部品供給データは使用する部品の供給形態により決まり、部品搬送データは部品の形状によりどのようなノズルで吸着するのかが決定できるため、データの作成には時間はかからない。しかし、部品形状データ及び画像処理アルゴリズムは基板上に装着する部品 1 つ 1 つに対して作成する必要があるため、データ項目数も多いため入力には時間がかかる。これらのデータは過去の生産に使用した部品のデータを流用することが一般的だが、初品は設計図データを元にした数値入力や、部品画像を予め撮像し画像を元に作成することが多い。

通常、生産時に使用される形状データと画像処理アルゴリズムは、設計図データを元にマウンタによる基板生産前に作成、決定しておくことが一般的である。機密保持等の理由により設計図が入手できない場合などは専門の人員が、専用の撮像装置などを用いて部品画像を撮像し、画像から形状データの作成、及び画像処理アルゴリズムの決定を行う。しかし、それらの作業には専門性が必要となるため省人化の障害となっている。SMD の種類が決まれば、画像処理アルゴリズムは一意に決定できるため、SMD の種類を自動認識することは、段取り工数を削減するために必要な処理である。

表 2-2 実装データ

データ名称	内 容
部品実装座標データ	プリント基板上へ装着する位置、角度データ ・装着座標、装着角度
部品形状データ	画像処理で使用する部品の形状を表すデータ ・ボディ寸法、色 ・リードの幅、長さ、ピッチ、本数 ・バンプの幅、ピッチ、個数 ※リード、バンプの数により増減する
画像処理アルゴリズム	部品形状データを処理する画像処理アルゴリズム 種の選択 ・画像処理アルゴリズム
部品供給データ	部品を供給装置に関するデータ ・供給装置の設置場所 ・供給装置の幅、送りピッチなど
部品搬送データ	部品を供給装置からプリント基板上へ搬送するためのノズルに関するデータ ・ノズル径 ・ノズル形状 ・搬送スピードなど

## 2.2 撮像システム

図 2-2 に部品撮像システム構成を示す。この構成は先行研究[16][17]のものと同様であり、コンピュータとカメラ及び撮像用光源で構成されている。カメラは 2000×2000[pixel]で、8bit 階調のグレースケール画像を撮像することができる。部品による高さの違いを吸収できるようにするため、ステージには部品にフォーカスが合う位置へ上下させる機構が搭載されており、高さ調節を行えば被写界深度内での撮像が可能となる。撮像環境を均一にするため光源は 2 種類用意されており、1 つはハーフミラーを介してカメラの光軸上から部品上

面を投光する落射光源で、もう一つは部品の斜め方向から投光する側射光源である。通常は落射光源と側射光源の両方を使用するが、BGA 等の突起形状の電極がある場合には、側射光源のみを使用する。側射光源のみを使用した場合、突起物のみが輝度が高く写り、BGA の配線パターンなど認識に用いない部分が映らないため認識には都合が良い。

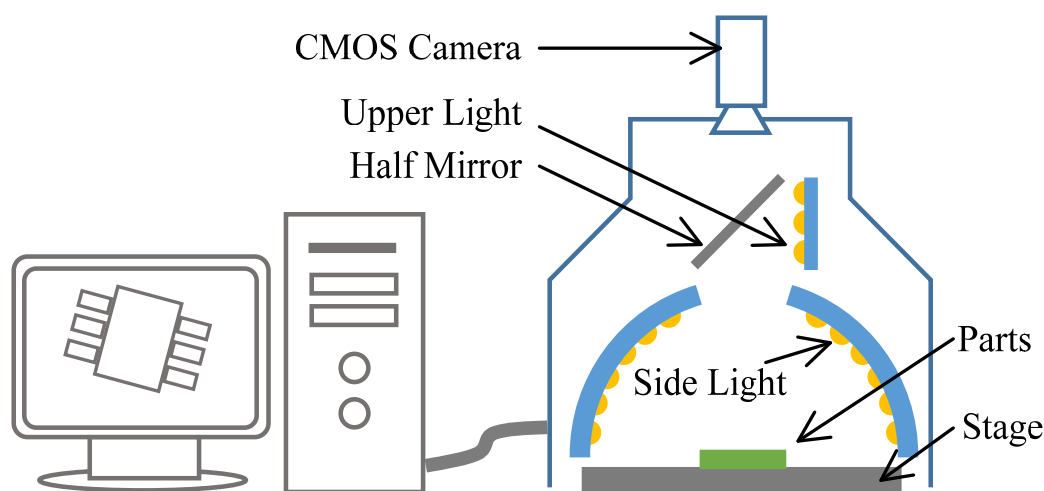


図 2-2 部品撮像システム構成

## 2.3 表面実装部品の種別

マウンタで装着する部品種類は様々であり、その種類ごとに必要となる画像処理アルゴリズムが異なるため、種類を判別した上でアルゴリズムを選択する必要があることは先に述べた通りである。種類の判別対象となる表面実装部品の種別を次頁より表として示す。表中には図 2-2 で示したシステムで撮像した取込画像を例として添付する。BGA 部品画像のみ側射光源で撮像した画像例であり、その他の部品については落射光源と側射光源を併用して撮影された画像である。部品は大きく分けて、リードやはんだボールなどの電氣的な接点となる電極部分と、本体であるボディ部、部品特有の機能を有する部分に分けられる。チップ部品は、電極を持つ矩形の部品であり、黒もしくは白色に撮像されるボディ部の左右に電極を持つことが多い。IC 部品は、黒いボディの周囲に短冊状のリードを有する部品である。BGA 部品は、CPU などに多く見られる部品であり、はんだボールがボディの上に格子状に並んでいることが特徴をもつ。撮

像時に落射光源を使用しないことからボディが映らないことも多い。MINI モード部品は、ボディ周辺に短冊状のリードを有する部品であり、IC 部品に比べてリードがボディに対して大きく、リードの数が少ないことが特徴である。コネクタ部品は、IC 部品と類似する部品ではあるが、IC 部品に比べてボディ上に様々な構造を有することが特徴である。異形部品は、上記以外の構造を有する様々な部品の総称である。

表 2-3 部品の種別 (チップ部品)

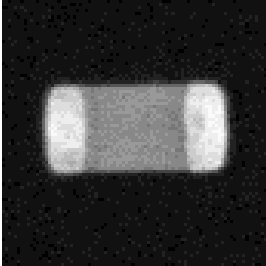
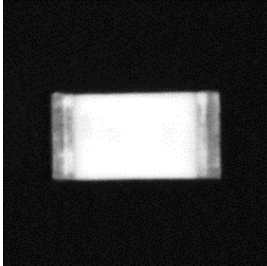
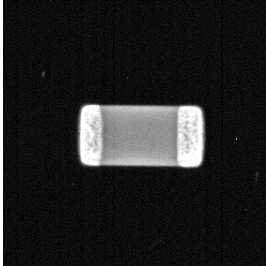
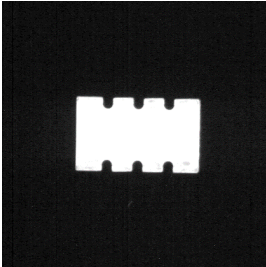
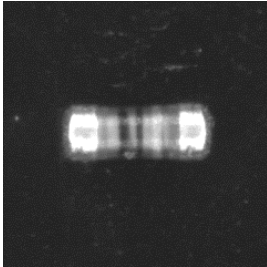
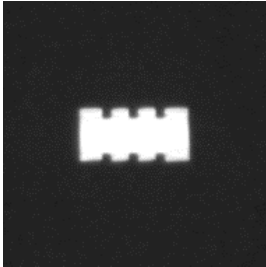
種別	取込画像および特徴		
チップ部品 (Chip)			
			
<ul style="list-style-type: none"> <li>・ 矩形形状</li> <li>・ ボディ部分が黒く映る場合がある</li> </ul>			

表 2-4 部品の種別 (IC 部品)

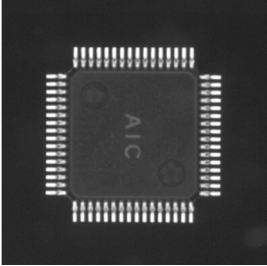
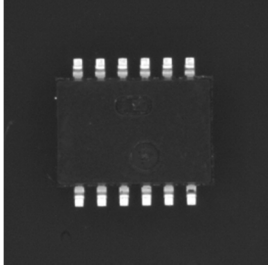
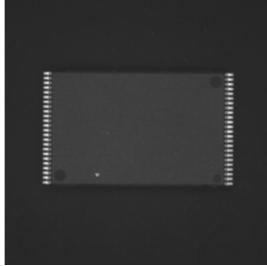
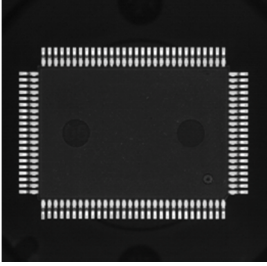
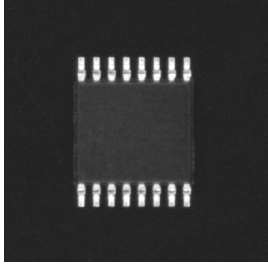
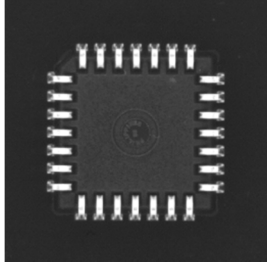
種別	取込画像および特徴
IC 部品 (IC)	<div style="display: flex; flex-wrap: wrap; justify-content: space-around;">    </div> <div style="display: flex; flex-wrap: wrap; justify-content: space-around; margin-top: 10px;">    </div> <ul style="list-style-type: none"> <li>・ 短冊状のリードが並んでいる</li> <li>・ ボディは黒く写らない</li> </ul>

表 2-5 部品の種別 (BGA 部品)

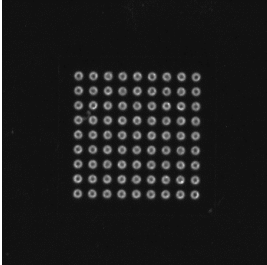
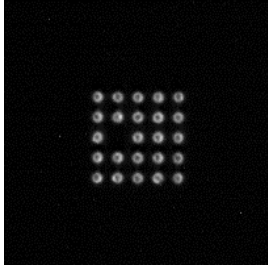
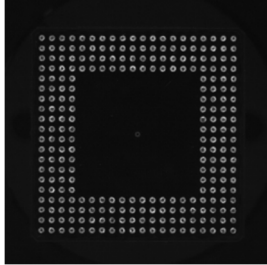
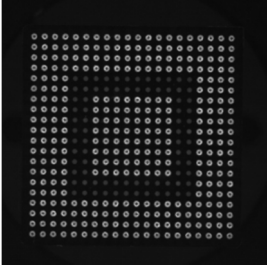
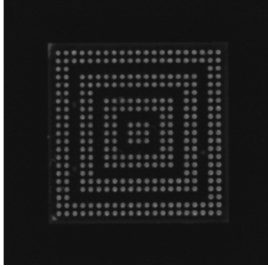
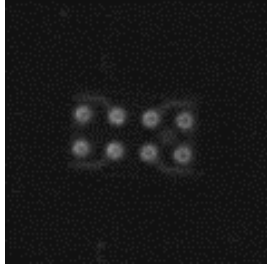
種別	取込画像および特徴
BGA 部品 (BGA)	<div style="display: flex; flex-wrap: wrap; justify-content: space-around;">    </div> <div style="display: flex; flex-wrap: wrap; justify-content: space-around; margin-top: 10px;">    </div> <ul style="list-style-type: none"> <li>・ はんだボールが格子状に並んでいる</li> <li>・ 千鳥配置など複雑な配置の場合もある</li> <li>・ 電極の形状は円形に写る</li> </ul>

表 2-6 部品の種別 (MINI モールド部品)

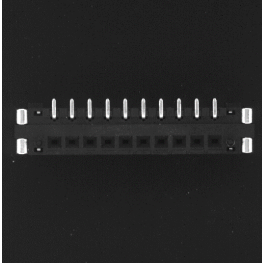
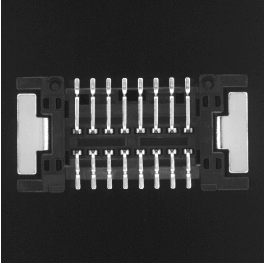
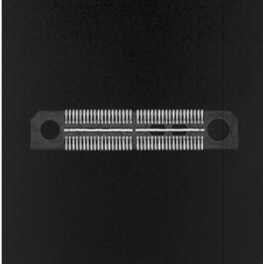
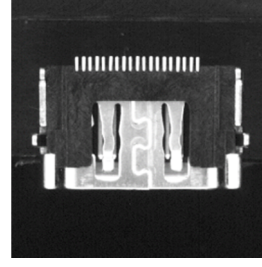
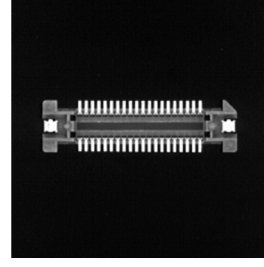
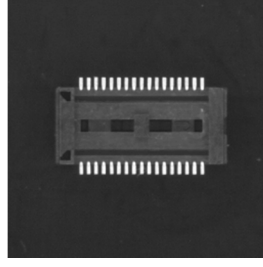
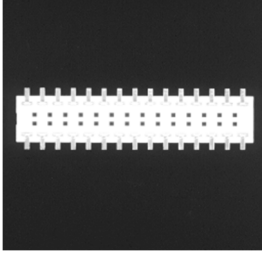
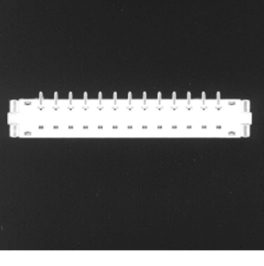
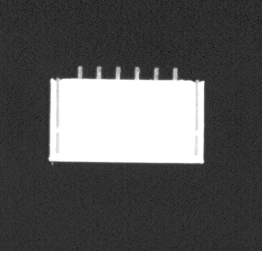
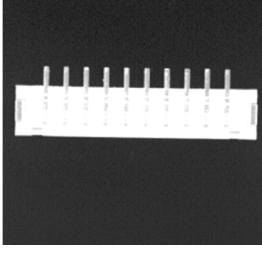
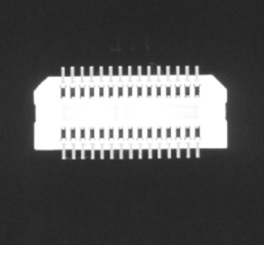
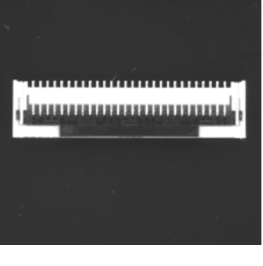
種別	取込画像および特徴		
MINI モールド部品 (Minimold)			
<ul style="list-style-type: none"> <li>・ 短冊状の電極がある</li> <li>・ 少数リード部品</li> </ul>			

表 2-7 部品の種別 (異型部品)

種別	取込画像および特徴		
異型部品 (Odd form)			
<ul style="list-style-type: none"> <li>・ 上記カテゴリ、コネクタのいずれでもない部品</li> <li>・ シールド部品、モーター、コイルなど</li> </ul>			



表 2-8 部品の種別 (コネクタ部品)

種別	取込画像および特徴		
コネクタ部品 (Connector)			
			
			
			
	<ul style="list-style-type: none"> <li>・ 基本的に短冊状の電極が並んでいるが、IC 部品よりも形状にばらつきが多い</li> <li>・ ボディ以外にも白く写るものがある</li> <li>・ ボディが白い場合もある</li> </ul>		

## 第3章 コネクタ部品の特徴抽出法

本章では、部品画像からその部品種別を分類する手法について述べる。部品種別の自動分類法は先行研究などにより既に提案されている手法があるが、その手法での問題点を整理した上で、提案手法の有効性について述べる。

### 3.1 まえがき

SMD は目的に応じて様々な形状のものが作成されている。抵抗器やコンデンサなどは角チップと呼ばれる矩形形状をした部品で作成されることが多く、トランジスタや IC などはリードと呼ばれる短冊状の電極を配置したものが多い。また CPU などは、はんだボールを電極とした BGA (Ball Grid Array) と呼ばれるパッケージで提供され、電極の形状は球体である。それぞれカメラで撮像した場合、リードは長方形形状が並んで撮像される場合が多く、バンパは円形に撮像される。

これらの SMD はマウンタを使用して部品供給装置から電子基板(PCBs)上に搭載される際に搬送中の保持姿勢をカメラにより撮像され、位置決め処理や検査処理が行われる。位置決め処理、検査処理は通常、上記の電極の配置や寸法をデータ化した形状データを用いて処理を行う。電極形状やボディの特徴にあわせて画像処理アルゴリズムが SMD の種類ごとに用意されており、適切なアルゴリズムを選択する必要がある。

通常、生産時に使用される形状データと画像処理アルゴリズムは、設計図データを元にマウンタによる基板生産前に作成、決定しておく必要がある。機密保持等の理由により設計図が入手できない場合などは専門の人員が、専用の撮像装置などを用いて部品画像を撮像し、画像から形状データの作成、及び画像処理アルゴリズムの決定を行う。しかし、それらの作業には専門性が必要となるため省人化の障害となる。SMD の種類が決まれば、画像処理アルゴリズムは一意に決定できる。

本章では図 2-2 で示した撮像装置により撮像した部品画像からの SMD の自動分類を目的とし、これまでに提案されている電子部品の自動分類手法である層型の分類システム[16][17]における問題点の整理を行い、特に分類性能の低

いコネクタ部品に対する分類性能の向上を行うことを目的とする。コネクタ部品は 2.3 節で示したように、同じリードを持つ IC 部品とは異なり、リード形状、色などが多岐にわたる。そのため従来手法では分類が困難な部品であり、部品形状データ自動作成システムとしては部品の分類性能を向上することが重要な課題となる。

## 3.2 従来の自動分類システムと問題点

基板作成工程における省人化のため、電子部品の種類の自動認識を行い、適切な画像処理アルゴリズムを選択するシステムが提案されている[16][17]。はじめにこの従来手法を用いて BGA、IC、MINI モールド(Minimold)、コネクタ(Connector)、チップ(Chip)、異型部品(Odd form)の各部品種に対する自動分類器を作成して評価を行う。使用した画像は 748 枚であり、予め人が分類したカテゴリに対し、自動分類装置が分類するカテゴリの正解率を計測した。結果を表 3-1 に示す。コネクタ以外の部品に対しては 95[%]以上の正解率があるが、コネクタ部品に対しては 40[%]と低い。コネクタ部品は電極と電極を接続するために用いられる部品であり、最終製品の大きさやデザインに合わせ多種多様な形状で作成される。

先行研究[16][17]では階層型の部品自動分類システムが提案された。第 1 階層では、輝度の高い部分の真円係数相対ヒストグラムを用い、BGA 部品とそれ以外の部品を分類し、第 2 階層では部品領域全体での X・Y 各軸方向への投影輝度、ゼロクロス頻度を求め、各部品カテゴリのリファレンスデータとの DP 距離を計測し特徴量に用いていた(図 3-1)。

本章で扱う誤分類の多いコネクタ部品は第 2 階層で分類されることになるが、部品の内部構造などの輝度変化により、リファレンスデータとの DP 距離が増大してしまうことが考えられる。結果的に部分空間法において、リファレンスの部分空間への距離が増大してしまうために正しく分類分けが出来ない。

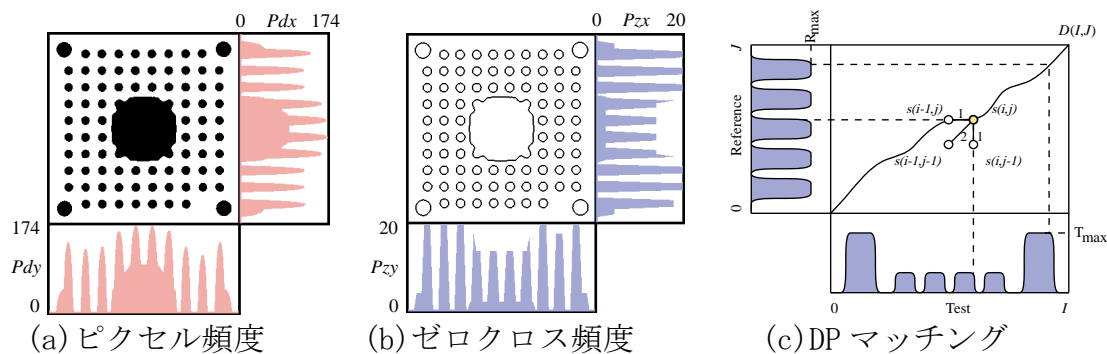
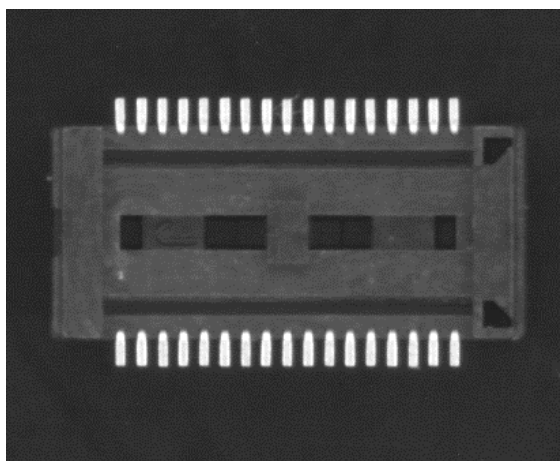


図 3-1 ピクセル頻度とゼロクロス頻度

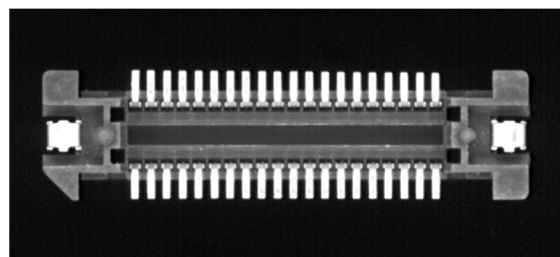
表 3-1. 従来手法による分類結果

Part categories		Result						Total	Accuracy rate [%]
		Chip	Minimold	IC	BGA	Connector	Odd form		
Input	Chip	162					1	163	99.4
	Minimold	4	139				3	146	95.2
	IC			100			1	101	99.0
	BGA	2			104			106	98.1
	Connector			4		46	65	115	40.0
	Odd form		1				116	117	99.1
Total							748	89.2	

図 3-2 に正常に分類されたコネクタ部品の画像を、図 3-3 に誤分類されたコネクタ部品画像の例を示す。以下の画像からもわかるように、コネクタ部品は多様な形状で作られている。IC 部品のように常にボディ部分が黒くなるわけでもなく、ボディ形状も複雑であり「一般的なコネクタ部品形状」を規定することは困難である。また、従来手法で使した部品全体のピクセル頻度およびゼロクロス頻度を使用した手法で部品分類性能を向上させようとした場合、多数の部品画像を部分空間法におけるリファレンスとして用意する必要があるため、効率的ではない。

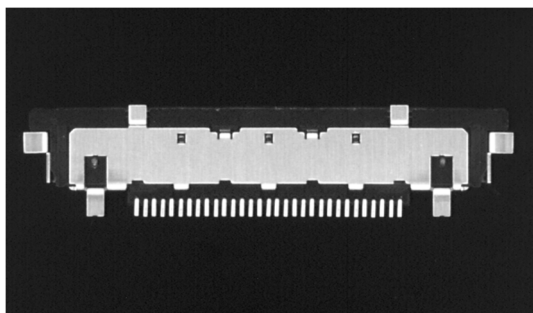


(a) コネクタ部品 A

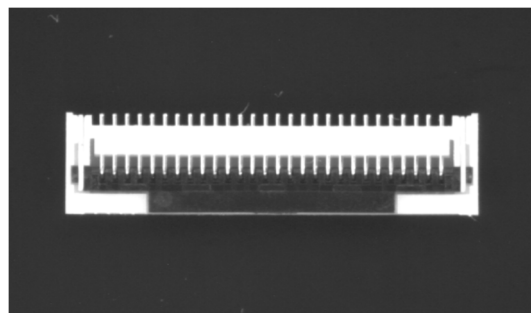


(b) コネクタ部品 B

図 3-2 正常に分類されたコネクタ部品画像



(c) コネクタ部品 C



(d) コネクタ部品 D

図 3-3 異型部品と誤分類されたコネクタ部品画像

本章では、特に分類性能の低いコネクタ部品を対象を限定し、コネクタ部品がもつ共通の特徴を得ることにより、コネクタ部品と異型部品を判別する特徴量の抽出法を提案する。そして提案手法を従来手法に組み込むことにより自動分類器の分類性能改善を行う。

### 3.3 信号解析基礎理論

本章で扱う部品の自動分類性能の向上には、音声解析などで用いられ得る周波数解析の手法を用いる。音声分析における信号解析において、音声信号を 1 次元の時系列データとしてとらえた場合、それをそのまま利用する方法と周波

数成分を利用する方法では、周波数成分を利用する場合の方が外部環境の変化に強いという理由から音声分析、音声認識の分野では主流となっている。本章で扱う特徴量の抽出にあたり、音声認識分野で使用している信号解析の理論を画像へ応用した。以下に信号解析理論について述べる。

### 3.3.1 フーリエ変換

フーリエ変換とは、音声のような波形について、どのような周波数成分が含まれるのかを求めるために用いる手法である。そのため変換後の関数は実データの周波数領域表現と呼ばれることがある。

連続時間信号 $s(t)$ を考えた場合、 $\omega = 2\pi f$ とすると基本周波数の正弦波成分と余弦波成分が元の信号にどの程度含まれているかを求めるには、 $s(t)$ と $\sin \omega t, \cos \omega t$ との相関を求めればよい。

$$\begin{aligned} \text{基本余弦成分} &= \int_{-\infty}^{+\infty} s(t) \cos \omega t \, dt \\ \text{基本正弦成分} &= \int_{-\infty}^{+\infty} s(t) \sin \omega t \, dt \end{aligned} \quad (3.1)$$

となる。しかし、この式は積分区間が $-\infty$ から $+\infty$ までとなっているため、実際の計算はこのままでは不可能である。そこで、 $s(t)$ および $\sin \omega t, \cos \omega t$ が周期 $T$ の波形となっていることを利用して、その成分である一周分きの波形を切り出して、次式のように相関を求めることで計算が可能となる。一周期に対する基本余弦波成分を $a_1$ 、基本正弦成分を $b_1$ とすると、

$$\begin{aligned} a_1 &= \int_0^T s(t) \cos \omega t \, dt \\ b_1 &= \int_0^T s(t) \sin \omega t \, dt \end{aligned} \quad (3.2)$$

同様にして、基本周波数の2倍、3倍、…の周波数に対する成分 $a_2, b_2, a_3, b_3, \dots$ が求まる。したがって、基本周波数の $n$ 倍の周波数に対する成分を $a_n, b_n$ とすると、上式は次式のように一般化できる。

$$\begin{aligned}
 a_n &= \int_0^T s(t) \cos(n\omega t) dt \\
 b_n &= \int_0^T s(t) \sin(n\omega t) dt
 \end{aligned}
 \tag{3.3}$$

また、 $a_0$ として、一周期における信号 $s(t)$ の平均値を次のように求める。

$$a_0 = \int_0^T s(t) dt
 \tag{3.4}$$

(3.3), (3.4)式を一般にフーリエ変換といい、その係数 $a_0, a_n, b_n$ をフーリエ係数という。ここで、余弦波成分を実数、正弦波成分を虚数とし、複素数で表現すると次式のように定義される。

$$S(n\omega) = \int_0^T s(t) e^{-jn\omega t} dt
 \tag{3.5}$$

### 3.3.2 離散フーリエ変換

コンピュータで信号を扱う場合、アナログ信号をそのまま扱うのではなく、デジタル信号として入力する。式では無限の範囲を扱うためこのままでは計算不能である。コンピュータでは離散データを扱うことになるため、積分を近似的に計算する必要がある。離散信号 $f(n)$ に対する離散時間のフーリエ変換は以下の式で表される。

$$F(e^{j\omega}) = \sum_{n=-\infty}^{\infty} f(n) e^{-j\omega n}
 \tag{3.6}$$

実際の計算では、無限の範囲では計算出来ないため、 $f(n)$ の有限の範囲でのみに着目して計算する必要がある。有限な $M$ 点の信号が $0, 1, \dots, M-1$ の範囲に存在するとした場合、(3.6)式は、

$$F(e^{j\omega}) = \sum_{n=0}^{M-1} f(n)e^{-j\omega n} \quad (3.7)$$

となる。しかし、コンピュータでは連続スペクトルの値をすべての $\omega$ について計算することは出来ない。そこで離散的な $\omega$ についてスペクトル値を計算する必要がある。いま、スペクトル1周期を $N$ 等分し以下の定義をする。

$$\omega_k = 2\pi k/N, \quad k=0,1,2,\dots,N-1 \quad (3.8)$$

(3.8)式を(3.7)式に代入すると、

$$F(e^{j2\pi k/N}) = \sum_{n=0}^{M-1} f(n)e^{-j2\pi nk/N} \quad (3.9)$$

となる。表現を簡潔にするため、

$$F(k) = F(e^{j2\pi k/N}) \quad (3.10)$$

$$W_N = e^{-j2\pi/N} \quad (3.11)$$

とおくと、(3.9)は以下のようになる。

$$F(k) = \sum_{n=0}^{M-1} f(n)W_N^{nk} \quad (3.12)$$

これが $N$ 点の離散フーリエ変換 (DFT :Discrete Fourier Transform) である。



### 3.3.3 高速フーリエ変換

ここで、離散フーリエ変換の計算量に注目する。仮に $N = 4$ を仮定した場合、(3.12)式は次のような行列表記が出来る。

$$\begin{bmatrix} F(0) \\ F(1) \\ F(2) \\ F(3) \end{bmatrix} = \begin{bmatrix} W_4^0 & W_4^0 & W_4^0 & W_4^0 \\ W_4^0 & W_4^1 & W_4^2 & W_4^3 \\ W_4^0 & W_4^2 & W_4^4 & W_4^6 \\ W_4^0 & W_4^3 & W_4^6 & W_4^9 \end{bmatrix} \begin{bmatrix} f(0) \\ f(1) \\ f(2) \\ f(3) \end{bmatrix} \quad (3.13)$$

1つの $F(k)$ を求めるために、4回の乗算と3回の加算を行う必要があることがわかる。この計算を一般化すると、 $N$ 点のDFTを得るためには複素数演算として $N^2$ の乗算と $N(N-1)$ 回の加算が必要になる。 $N$ が512の場合では、 $N^2 = 262144$ となり今日の高速な演算処理が可能なコンピュータを用いても無視出来る計算量とはならない。

この演算を高速に実行するために、 $e^{j2\pi nk/NT}$ の周期性を利用して考察されたのが高速フーリエ変換(FFT: Fast Fourier Transform)である。この手法を使用することにより、必要な複素乗算の回数は $N \log_2 N$ 回ですむことになる。したがって、DFTに比べて $N \log_2 N / N^2 = \log_2 N / N$ の比率で少なくてすむ。例えば、 $N = 128$ として計算量を比較すると $\log_2 N / N = 7/128$ 、 $N = 256$ として計算量を比較すると $\log_2 N / N = 8/128 = 1/32$ となり、 $N$ が大きい程この高速化の効果は顕著になる。

次にFFTの計算原理について述べる。処理の前提として、処理する信号の標本値 $N$ を2のべき乗( $N = 2^m$ )であるとする。まず、信号 $s(t)$ の標本値の系列 $\{s(n)\}, n = 0, 1, \dots, N-1$ を偶数番目の標本値の系列 $\{g_l\}$ と奇数番目の標本値の系列 $\{h_l\}$ に分ける。

$$\begin{aligned} \{g_l\}: s_0, s_2, \dots, s_{N-2} \{g_l\} &= \{s_{2l}\} \\ l &= 0, 1, \dots, \frac{N}{2} - 1 \\ \{h_l\}: s_1, s_3, \dots, s_{N-1} \{h_l\} &= \{s_{2l+1}\} \\ l &= 0, 1, \dots, \frac{N}{2} - 1 \end{aligned} \quad (3.14)$$

この2つの系列は共に項数が $N/2$ であり、周期が $NT/2$ の系列とみなすことができる。したがって、そのDFTは、

$$\begin{aligned}
 G(k\Omega) = G_k &= \sum_{l=0}^{\frac{N}{2}-1} g_l e^{-jw2\pi\frac{2}{NT}kl} = \sum_{l=0}^{\frac{N}{2}-1} g_l (W^2)^{kl} \\
 H(k\Omega) = H_k &= \sum_{l=0}^{\frac{N}{2}-1} h_l e^{-jw2\pi\frac{2}{NT}kl} = \sum_{l=0}^{\frac{N}{2}-1} h_l (W^2)^{kl} \\
 \text{ただし, } W &= e^{-jw2\pi\frac{1}{NT}}
 \end{aligned} \tag{3.15}$$

一方、 $\{s_n\}$ のDFTは、

$$\begin{aligned}
 F(k\Omega) = F_k &= \sum_{i=0}^{N-1} s_n e^{-jw2\pi\frac{1}{NT}ki} \\
 &= \sum_{l=0}^{\frac{N}{2}-1} \{s_{2l} W^{2lk} + s_{2l} W^{(2l+1)k}\} \\
 &= \sum_{l=0}^{\frac{N}{2}-1} \{g_l W^{2lk} + h_l W^{(2l+1)k}\} \\
 &= G_k + W^k H_k \\
 k &= 0, 1, \dots, N-1
 \end{aligned} \tag{3.16}$$

ここで、 $\{G_k\}, \{H_k\}$ は周期が $NT/2$ であることを使えば、

$$F_k = \begin{cases} G_k + W^k H_k & k = 0, 1, \dots, \frac{N}{2} - 1 \\ G_k - \frac{N}{2} + W^k H_k - \frac{N}{2} & k = \frac{N}{2}, \frac{N}{2} + 1, \dots, N-1 \end{cases} \tag{3.17}$$

上式は、 $N$ 点のDFTの計算が、 $N/2$ 点のDFTの計算2回に分解できることを示している。例えば $N=8$ の時にこの処理を繰り返していけば、 $2N \log_2 N$ 回の計

算で $N$ 点の DFT が求まることになる。しかも、各点での積和の計算のうち、一方は必ず乗算の係数が 1 であり単なる加算の演算になり計算が大幅に最適化される。

このような FFT の演算手法は、Cooley-Tukey 型のアルゴリズムとして知られている [18]。この手法を用いれば、必要な複素乗算回数は $(N/2)(\log_2 N - 1)$ 、複素加算回数は $N\log_2 N$ 回となりデータ点数が 2 の階乗であるという条件であれば、高速にフーリエ変換の演算することが可能となる。FFT に関しては他にも様々なアルゴリズムや実装が存在する [19]–[21] が、本研究では大浦 FFT [22] を利用した。

### 3.4 特徴量抽出

画像を用いた自動識別に関する先行研究では HOG 特徴を用いた人検出 [5]–[8] や、Haar-Like 特徴を使用した顔検出手法 [10] が知られている。HOG 特徴は輝度勾配情報を特徴量としており、例えば人の着ている服の色などに対してロバストな特徴量として報告されている。Haar-Like 特徴は画像中における輝度差を特徴量としており、人の顔の持つ共通の輝度差（目と頬など）を記述するのに有効であることが報告されている。複数の HOG 特徴量を組み合わせた Joint HOG 特徴量や複数の Haar-Like 特徴量を組合せ得た Joint Haar-Like 特徴も提案されており、車両検出 [23] [24] や顔検出の性能向上 [25] などが報告されている。

今回対象とする電子部品のリード部には通常リードが周期的に並んでおり、部品ごとにリードの幅は異なる。エッジ位置や相対位置の濃淡に基づく HOG 特徴や Haar-Like 特徴では、幅の違いにより記述される特徴が異なるため、幅の変動に対する不変特徴とはならない。取り付けられているリード本数やピッチも部品ごとに異なるため、HOG 特徴や Haar-Like 特徴ではリード単体の検出は可能でも、リード部全体の周期的な特徴を得ることは困難である。3.2 節で示したように先行研究 [16] [17] の手法で分類できない部品、すなわちコネクタ部品に注目し、周期性に基づいた判別手法を用いて異型部品とコネクタ部品を分類する手法を提案する。

### 3.4.1 リード群の判別領域の特定

コネクタ部品が共通に持つ特徴は、リードと呼ばれる短冊状の電極が部品に取り付けられていることである。電子部品の電極としては、他にもはんだボールという BGA などに用いられる球状の電極があるが、コネクタにはほぼ用いられていない。

リードは通常、部品のボディ部分から外部方向へ取り付けられている。そのため、部品領域の外周付近に存在する確率が高く、逆に部品中央部にリードが存在する確率が低いことが予想される。リードは同一形状のものが等間隔に数本から数十本、同一方向に向いて並んでいる場合が多い。この同一方向に並んだリードのまとまりをリード群とし(図 3-4)、予めリード群が定義されている部品画像から、部品領域内におけるリード群の存在する確率の高い領域を求める。

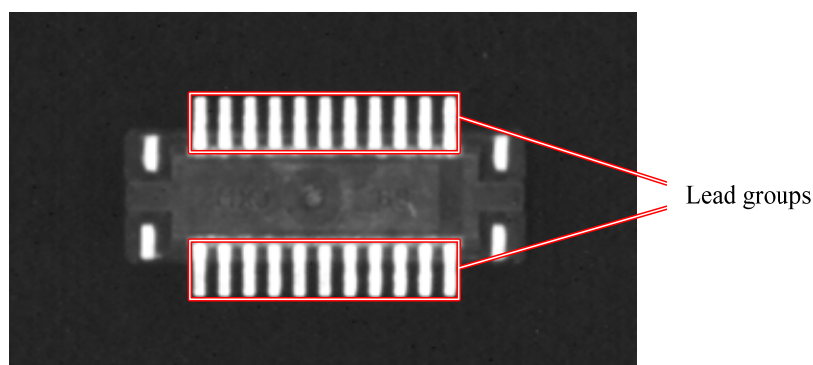


図 3-4 リード群の例

部品は  $0^\circ$ 、 $90^\circ$ 、 $180^\circ$ 、 $270^\circ$  のいずれの向きで供給されるのかはユーザーの運用方法で決まるため、どの方向が正しい部品の向きかは画像からは決定できない。そこで部品画像を4方向に回転させ、人手で定義したリード群を画像中に投票してリード群の存在確率を表す画像(以下、リード群領域画像と呼ぶ)を生成した。部品領域は縦横比が1:1(512×512 pixel)の正方形領域になるように正規化し、リード群の投票最大数が輝度255になるように正規化してリード群が存在する領域を画像化した。使用した画像は全215枚のリードを持つ部品画像である。図3-5および図3-6はリード群領域画像の作成手順と得られた平均画像を示したものである。この画像より、リード群は部品領域の外周部分に集中していることが確認できる。得られた平均画像を判別分析法によ

り2値化したものが図 3-7 である。2 値化の結果から図中のように1 辺の 1/8 の長さの領域でリード群判別領域を取得し、この領域で切り抜いた画像をリード群判別領域とした。これを上下左右の4 箇所から取得し、それぞれの領域からリード群の有無を判別する。

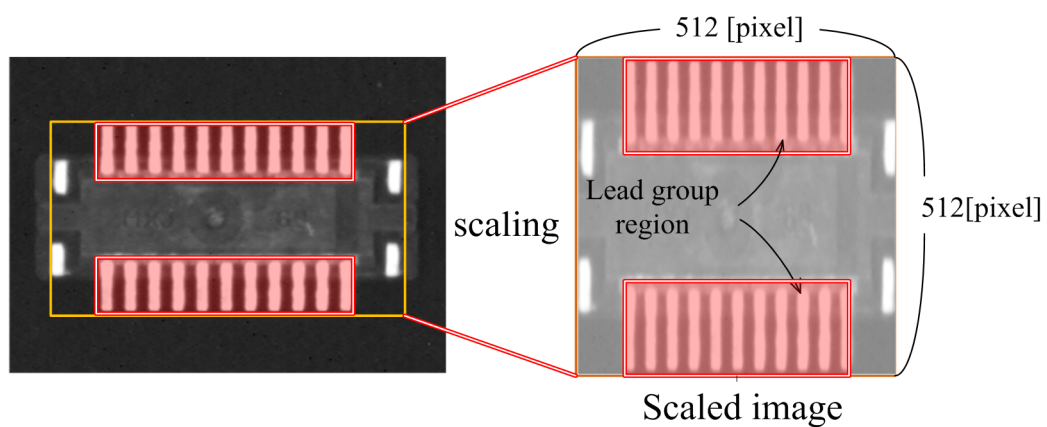


図 3-5 部品領域の正規化

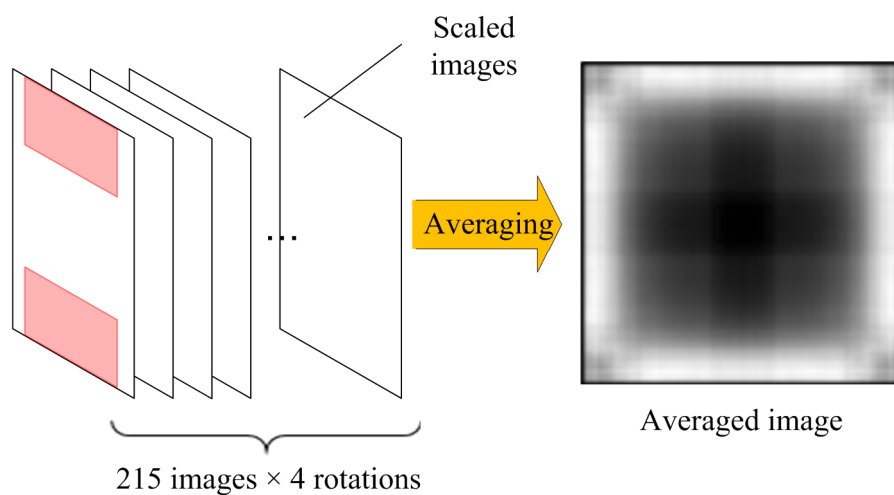


図 3-6 リード群領域の画像化

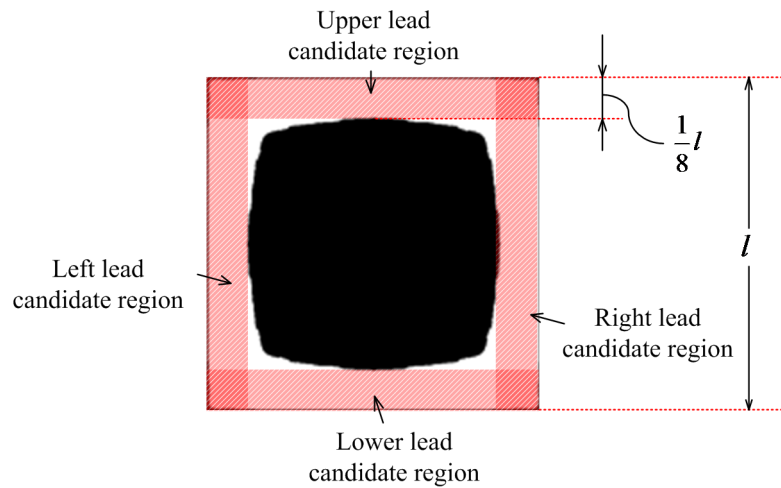


図 3-7 判別分析法による二値化とリード群判別領域

この手法により得られたリード群の判別領域画像の例を図 3-8 に示す。切出したリード群判別領域画像は、すべて部品の外側が上向きになるように回転補正して表示した。

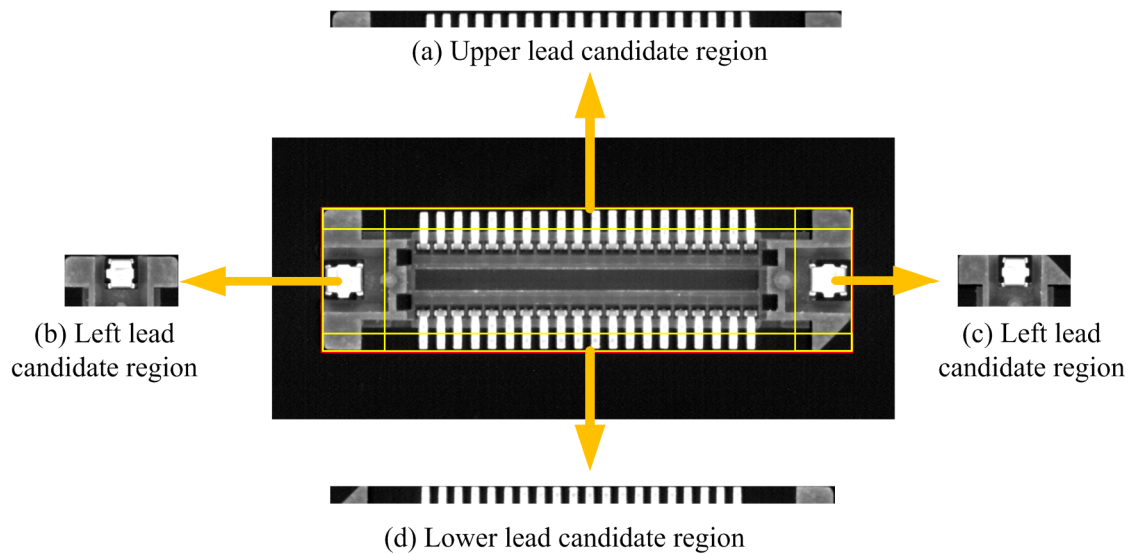


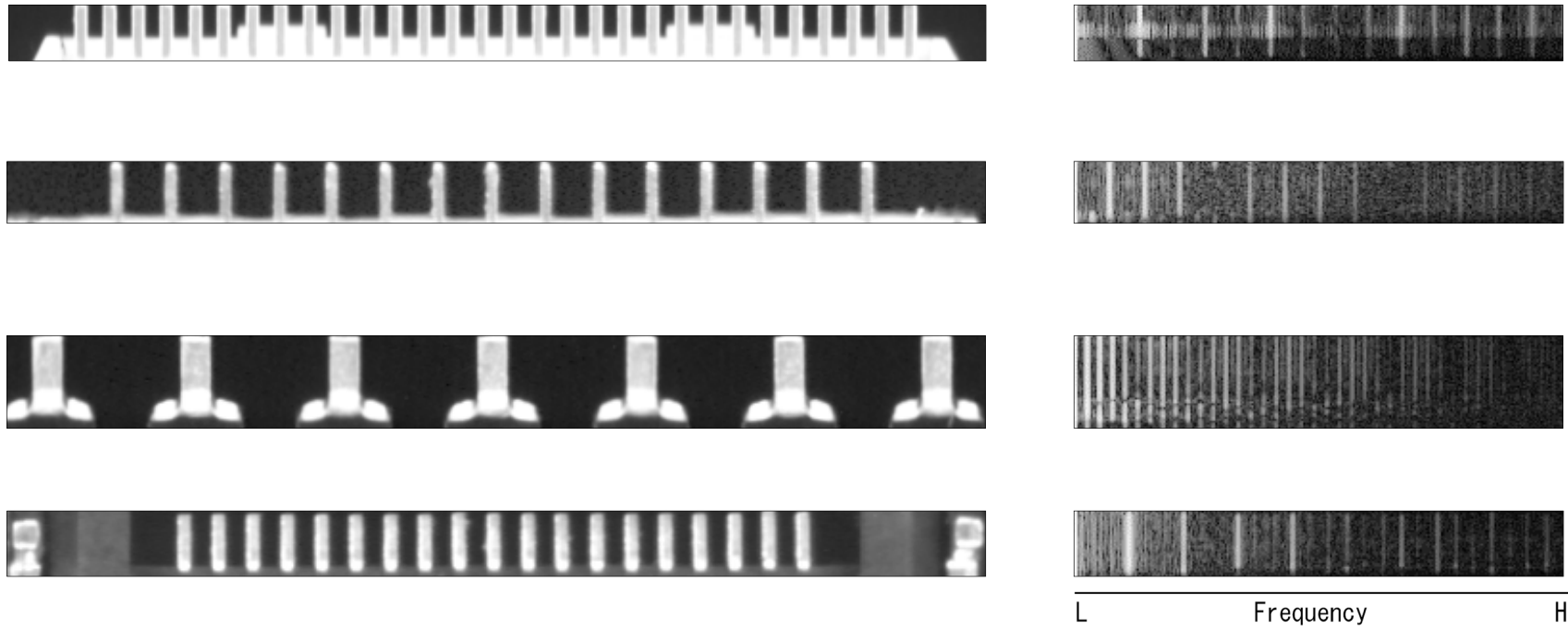
図 3-8 リード群判別領域の抽出例

### 3.4.2 リード群判別領域画像への FFT の適用

3.4.1 項の手法で抽出したリード群判別領域画像から、周期性の特徴を得る

ための特徴抽出法について述べる。リード群判別領域には、実際にリード群が存在するサンプル(Positive)とリード群が存在しないサンプル(Negative)がある。これらを目視により分類し、Positive サンプル 444 枚、Negative サンプル 737 枚を得た。取得したリード群判別領域画像の例を図 3-9 (a) 図 3-10 (a)に示す。画像はすべて部品領域の外側が上向きになるように回転補正をかけて表示した。

図 3-9 (a)に示す画像はリードが含まれる領域の例であり、図 3-10(a)に示す画像はリードが含まれない領域の例である。図 3-9 (a)に関しては水平方向に規則的な輝度変化が認められる。そこでこの領域画像に対し、水平方向 1 ラインごとに FFT をかけて画像化しパワースペクトル画像を取得した。FFT をかける際に 2 の階乗のデータ数が必要となるため、図 3-9 (a)、図 3-10 (a)に示すリード群判別領域画像は、幅方向をすべて 512[pixel]となるように伸長して作成してある。また、FFT をかける際に Hamming 窓を適用した。得られたパワースペクトル画像を図 3-9 (b)、図 3-10 (b)に示す。リード群を含む画像ではパワースペクトル画像において周期的な縦線が確認できるが、リードを含まない領域の画像では周期的な縦線が確認できない。縦縞の出現位置は、512[pixel]への画像伸張や、リード間距離 (ピッチ) によって異なる。そのため、縦縞の特徴を得るためには位置に不変な特徴量を使用することが有効であると考えられる。

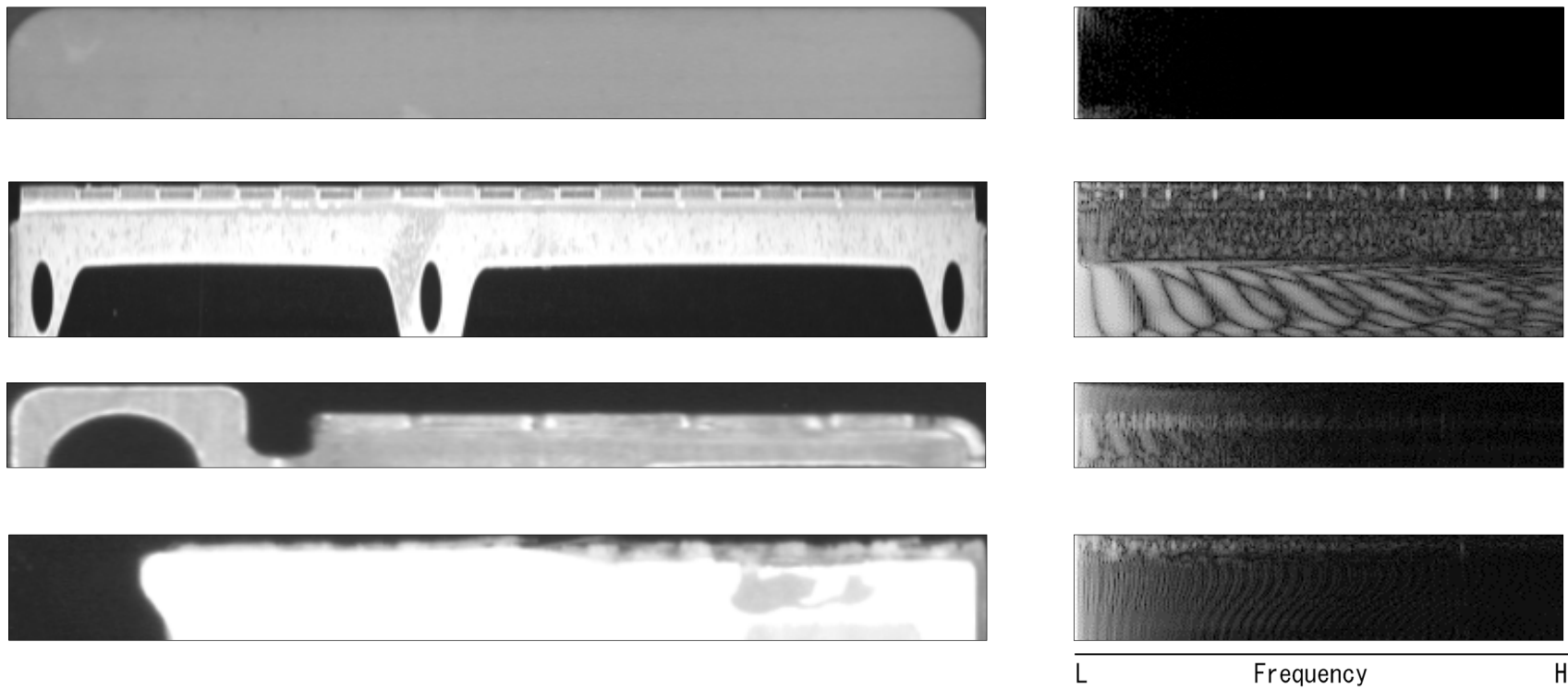


(a) リード群判別領域画像

(b) パワースペクトル画像

図 3-9 Positive リード群判別領域画像の例





(a) リード群判別領域画像

(b) パワースペクトル画像

図 3-10 Negative リード群判別領域の例

### 3.4.3 高次自己相関特徴 (HLAC)

パワースペクトル画像中に縦縞が出現するかどうかを判断するため、画像中から高次自己相関 (HLAC : Higher-order Local AutoCorrelation) 特徴[26]を取得し、これを特徴量とすることでリード群を持つ部品かどうかを判定する手法を提案する。HLAC はテクスチャ認識や文字認識、顔認識、物体認識といった様々な画像処理で使用されており[27]–[35]、汎用性の高い特徴量として提案されている。2次元のHLACは一次の自己相関関数を、画像(二次元データ)でも扱えるように拡張したものである。一般的に一次の自己相関関数は(3.18)式のようにあらわされる。

$$X(a) = \int f(r)f(r + a_N)dr \quad (3.18)$$

ここで $f(r)$ はある一次元信号中の位置 $r$ での実関数(データ)値、 $a$ は自己相関を計算する際の平行移動量(相関幅)、 $X(a)$ は位置 $r$ から $a$ だけ平行移動した際の自己相関量である。このように自己相関関数はある信号中の任意の位置 $r$ での関数の値と、 $r$ から $a$ だけ平行移動した場合の値との積であらわせられ、その間の相関を示す。この $X(a)$ のフーリエ変換は原波形のパワースペクトルを示す。

自己相関関数を画像などの二次元波形に用いるために $N$ 次に拡張した高次局所自己相関は大津により提案された。 $N$ 次の高次自己相関関数は $N$ 個の変位 $a_i(i = 1, \dots, N)$ に対して以下のようにあらわされる((3.18)式に示す自己相関関数は $N = 1$ の場合として含まれる)。

$$x(a_1, a_2, \dots, a_N) = \int I(r)I(r + a_1) \cdots I(r + a_N)dr \quad (3.19)$$

(3.19)式は画像中の注目点 $r$ での輝度値と $N$ 個の変位点 $a_i$ での輝度値との積をとり、画像全体で総和をとることを意味している。

自己相関関数を(3.19)式のように高次に拡張することで二次元画像中において様々な組み合わせによる相関を求めることができる。そのため求めた特徴量は、組み合わせが多ければ多いほど画像のテクスチャパターン固有のものに

なるとされている。これまでもこの特徴量を用いて画像中の文字認識や顔抽出などが行われている。

しかし自己相関関数を高次に拡張した場合、変位の組み合わせによって得られる特徴の数が $N$ に対して指数関数的に膨大となる(特徴量の最大数= $N^{w \times h - 1}$ )ので、実際の応用では $N$ や変位 $a_i$ がとりうる範囲を限定する。

たとえば変位 $a_i$ の範囲を、注目画素位置 $r$ を中心とした $3 \times 3$  格子状の8方向に限定すると、変位 $a_i$ がとりうる方向は図 3-11 のようになる。この場合考えられる最高次数は $N = 8$ であるが、ここで $N = 2$ 次までと次数を限定した場合、自己相関をとりうる組み合わせは図 3-12 に示す 25 種類となる(平行移動等価となるものは除く)。この図では中心の黒塗り部が注目画素 $r$ であり、注目画素の輝度値 $f(r)$ とその他の黒で示す画素(参照点)の輝度値 $f(r + a_i)$ との積を取ることを示している。特徴量は、ある注目画素において局所領域内で求めた自己相関の値を求め、注目画素を移動させながら自己相関の値を画像全体で累積することで求める。つまり $N = 2$ 次まで、格子(マスク)サイズを $3 \times 3$ とした場合、(3.19)式は図 3-12 であらわす 25 種類のマスクパターンそれぞれで、画像全体を走査し積分することを示している(図 3-13)。その結果 2 次元画像中から 25 次元の特徴量ベクトルが生成される。

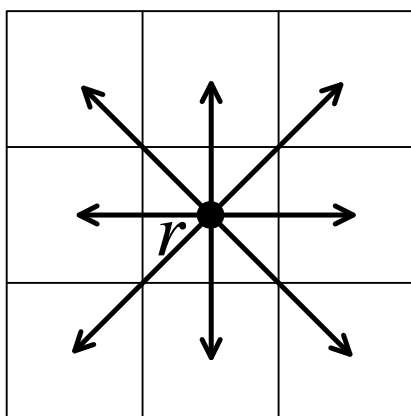


図 3-11 局所格子内での変位方向

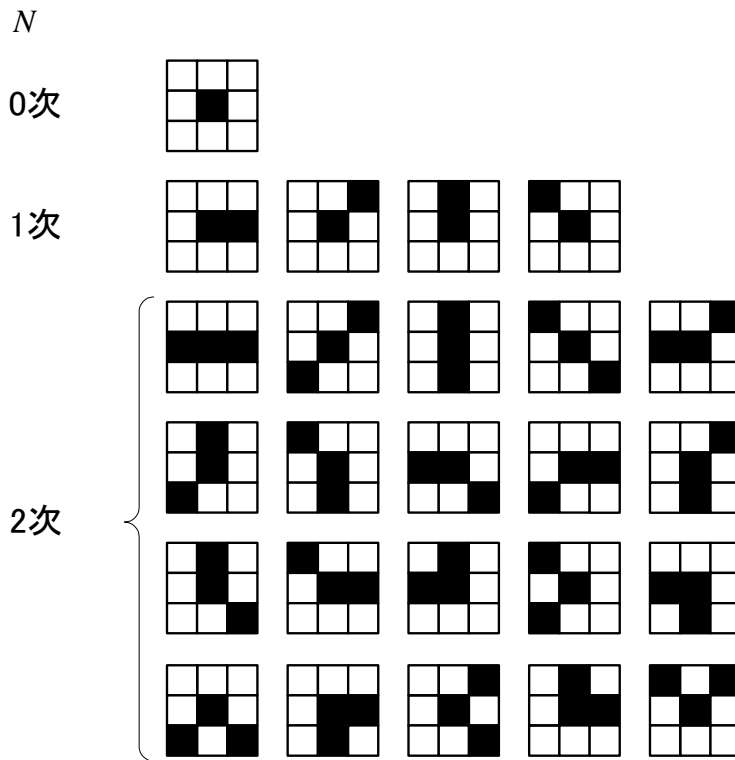


図 3-12  $N = 2$  までの変位組み合わせ

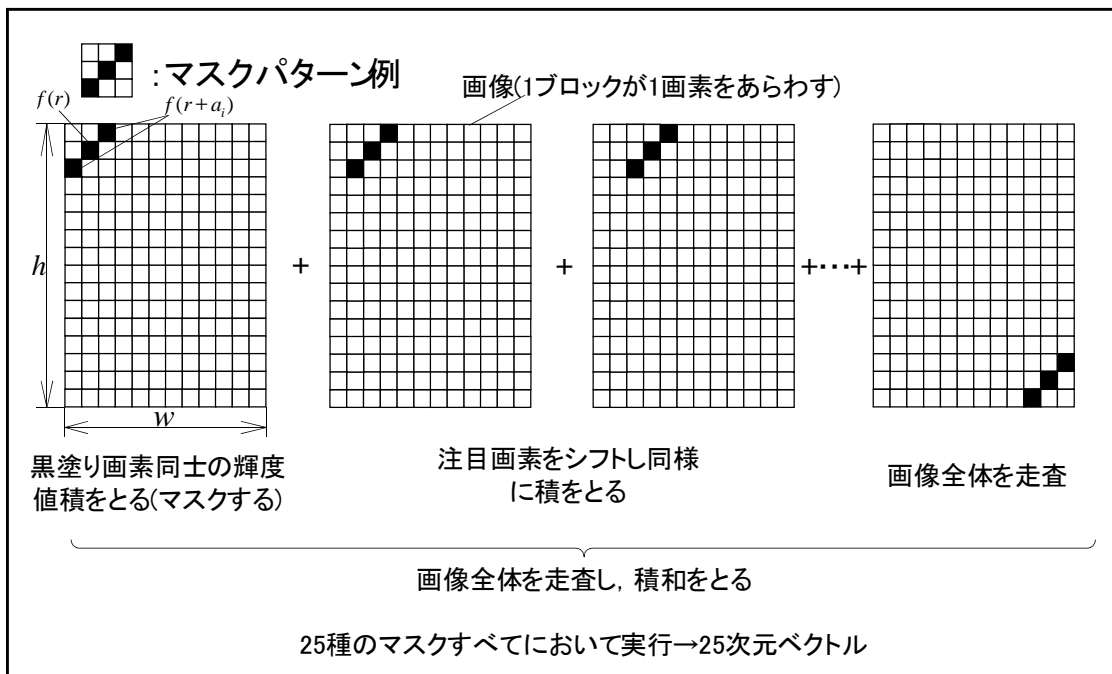


図 3-13 高次局所自己相関の導出

なお  $3 \times 3$  の格子サイズ(マスク)から抽出される特徴は画像中の高周波成分

からなり、用いる格子サイズを  $5 \times 5$ ,  $7 \times 7$  などへ拡大することで低周波成分の特徴も抽出可能となる[36]。ただし、格子サイズを拡大した場合は特徴量の組み合わせ数が  $N^{w \times h - 1}$  となることから、組み合わせが膨大な数となる。また対象が画像である場合に注目点とその近傍画素の局所的な相関関係が形状などの重要な特徴情報である場合が多いため  $3 \times 3$  の局所領域内で高次自己相関を求める場合が多い。また、計算コストの関係から  $N$  は 2 までとすることが多い。

なお、局所領域を  $5 \times 5$  や  $7 \times 7$  など大きくする場合は特徴量の組み合わせ数を減らすために参照点数の制限(次数  $N$  の制限)を導入する場合が多い[37]。低周波成分と高周波成分どちらも抽出するために、[37]のように小サイズマスクと大サイズマスク両方から特徴量を抽出する場合や複数解像度の画像を使用する場合もある[28][38]。

同様に  $3 \times 3$  の局所領域で  $N = 2$  に限定すると、グレースケール画像に対しては 35 次元の特徴ベクトルが得られる。本研究では濃淡画像を対象として 35 次元の特徴ベクトルを使用する。35 の組合せパターンを図 3-14 に示す。

HLAC 特徴は、加法性、位置不変性という特徴を持つ。そのため、リード間ピッチや 512[pixel] への画像伸長により縦縞の位置が変化するパワースペクトル画像に対して有効な特徴量であると考えられる。

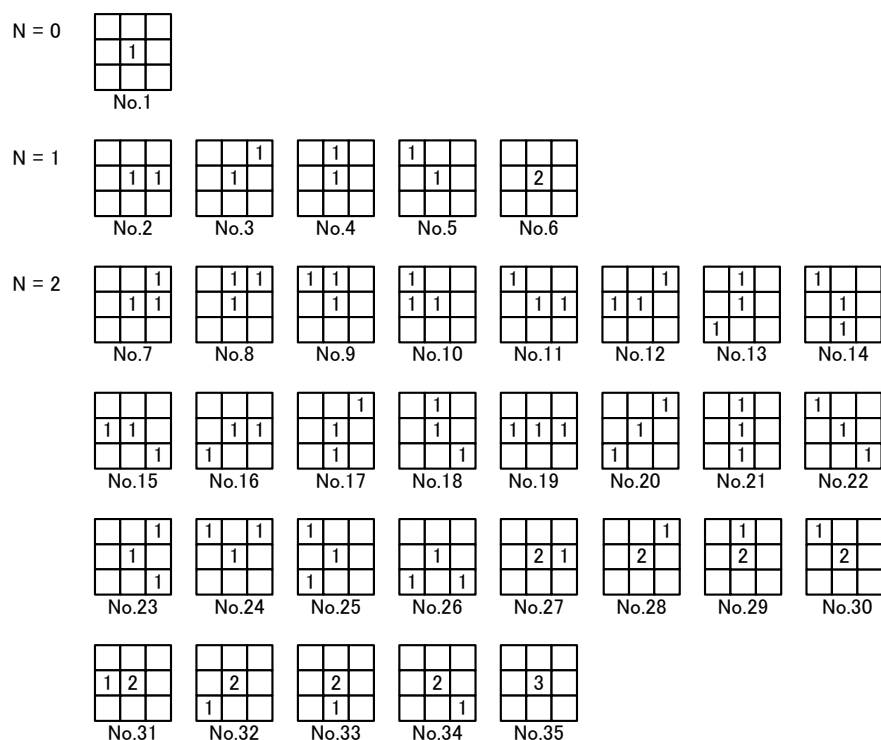


図 3-14 グレースケール画像での HLAC マスクパターン

### 3.4.4 Local Binary Pattern (LBP)

Local Binary Pattern はテクスチャ認識のために T. Ojala らにより提案された特徴抽出法である [39]。LBP は照明変動に対する頑健性や計算コストの低さという利点があり、顔認識、表情認識等に応用されている [40]–[44]。HLAC と同様、ヒストグラム表現となるため位置不変性の特徴が得られる。LBP は注目点と周囲の画素値との大小を見る特徴量である。その大小を 0 と 1 に 2 値化し、その値を 2 進数と考えることで 10 進数の数値を算出する。その結果、0-1 の変動パターンから 0 から 255 の値を得る。そのような演算を 16×16 などの範囲で行い、一つのヒストグラムにまとめることで特徴量とする。図 3-15 に LBP 取得の例を示す。

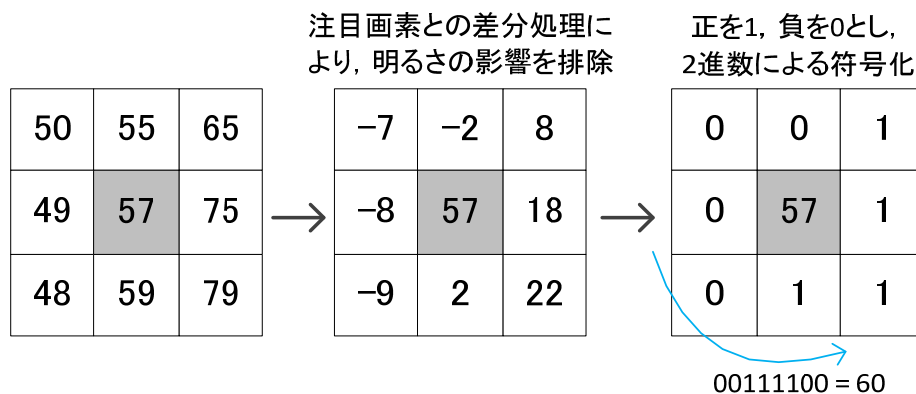


図 3-15 8 近傍画素における LBP の取得

この場合、LBP = 60 と計算される。

### 3.4.5 Local Ternary Pattern (LTP)

Local Ternary Pattern も LBP と同様にテクスチャ認識のために X. Tan らにより提案された特徴抽出法である [45]。LBP と同様に様々な検出処理のための特徴量として利用されている [46] [47]。LTP は注目画素値そのままを周囲と比較せずに、オフセットを設けることでノイズの影響を軽減している。例えば、注目画素値が 57 で、オフセットが 5 の場合  $57-5=52$  と  $57+5=62$  が新たな閾値となり、52 以下を -1、62 以上を 1、それ以外を 0 とする (図 3-16)。これら

を二値化するため、-1 を 0 とする、あるいは 1 を 0 とし、-1 を 1 に書き換える処理を行う。その結果、二つの LTP 特徴量を作成できる。前者を LTP-Positive、後者を LTP-Negative と呼ぶ。LTP も LBP と同様に周囲の画素値と注目点との差を計算するため、明るさの変動にロバストな特徴となる。

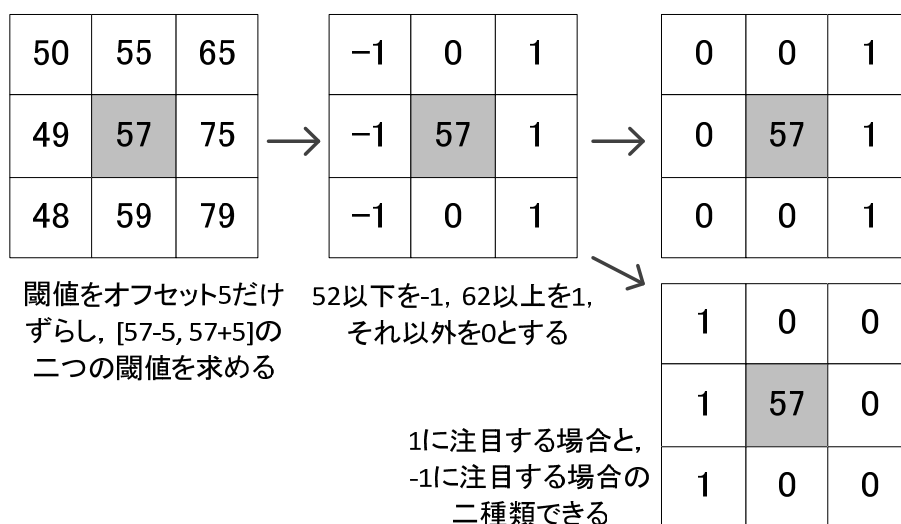


図 3-16 LTP の導出方法

### 3.5 サポートベクターマシン

2クラスの分類問題を解くにあたり、その目的は有効なサンプルから2クラスを分けるための関数を得ることにある。サポートベクターマシンは、Vladimir N. Vapnikらにより1960年代に行われた最適超平面による識別法の提案とその汎化能力に関する解析に期限を持つ。1990年代にはカーネル学習法と組み合わせた非線形の識別手法へと拡張され、高次元のデータの2値判別器として、様々な認識や判別において使用されている[48]–[55]。

まず、図 3-17(a)の例を考える。●と×を線形分離するための分類境界は複数存在する(図中の直線  $l, m, n$ )。マージンを分類境界と最も近くのデータ点までの距離として定義し、マージンを最大化するように最適化することで、分類境界は一意となる。これは○印で示したような一部の訓練データ点(サポートベクター)により決定される。

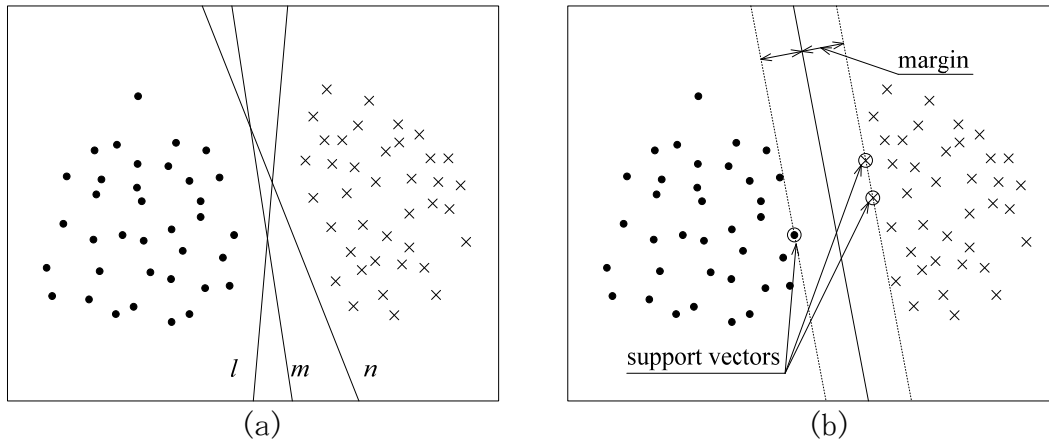


図 3-17 超平面を用いた 2 クラスの分類

(a) 有効な分類境界 (b) 点線により最大マージンで最適化された分類境界

線形モデルを用いて 2 値分類を解く場合 SVM の識別関数は以下のように表される。

$$y(\mathbf{x}) = \mathbf{w}^\top \phi(\mathbf{x}) + b \quad (3.20)$$

ここで  $\phi(\mathbf{x})$  はある固定された特徴空間関数であり、 $b$  はバイアスパラメータである。訓練データは  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$  の  $N$  個の入力ベクトルとそれぞれに対応する目標値  $t_1, t_2, \dots, t_N (t_N \in \{-1, 1\})$  からなり、未知のデータ点  $\mathbf{x}$  は  $y(\mathbf{x})$  の符号に応じて分類される。これらの訓練データと目標値からパラメータ  $\mathbf{w}$  と  $b$  を求めることが目的である。訓練データが全て線形分離可能であるとした場合、分類関数 (3.20) 式が  $t_n = +1$  である点については、 $y(\mathbf{x}_n) > 0$ 、 $t_n = -1$  である点については、 $y(\mathbf{x}_n) < 0$  が成立する。そのため  $t_n y(\mathbf{x}_n) > 0$  と定義できる。したがって、分類境界から点  $\mathbf{x}_n$  までの距離は次のように表すことができる。

$$\frac{t_n y(\mathbf{x}_n)}{\|\mathbf{w}\|} = \frac{t_n (\mathbf{w}^\top \phi(\mathbf{x}) + b)}{\|\mathbf{w}\|} \quad (3.21)$$



マージンは訓練データと分類境界の最短距離であり、今求めたいのはそのマージンを最大化する $\mathbf{w}$ と $b$ を求めることなので、解は次の最適化問題を解くこととなる。

$$\max_{\mathbf{w}, b} \left\{ \frac{1}{\|\mathbf{w}\|} \min_n [t_n(\mathbf{w}^\top \phi(\mathbf{x}) + b)] \right\} \quad (3.22)$$

分類境界に最も近い点について $t_n(\mathbf{w}^\top \phi(\mathbf{x}_n) + b) = 1$ とすると、すべてのデータについて次の制約式が成立する。

$$t_n(\mathbf{w}^\top \phi(\mathbf{x}) + b) \geq 1 \quad (3.23)$$

この制約はすべてのデータは $-1 < y < +1$ の間にはないことを示している。つまりマージンができる事となる。この制約のもとで最適化問題は $\|\mathbf{w}\|^{-1}$ を最大化する問題となる。つまりこれは $\|\mathbf{w}\|^2$ を最小化する問題と等価となり、以下の解を(3.20)式の制約のもとで求めれば良いことになる。

$$\min_{\mathbf{w}, b} \frac{1}{2} \|\mathbf{w}\|^2 \quad (3.24)$$

通常2クラスのカテゴリを行う場合、両クラス間に重なりがあることが考えられる。この場合、訓練データを完全に分離する解が必ずしも汎化能力に優れているとは限らない。これまですべての訓練データが線形分離可能であると仮定してきたが、線形分離不可能である場合マージン境界からの距離に応じたペナルティを与えることで、間違った分類を許すように式を修正する。具体的にはスラック変数 $\xi_n \geq 0$ を導入する。スラック変数は各訓練データごとに定義される変数で、データが正しく分類された場合には $\xi_n = 0$ 、それ以外の場合には $\xi_n = |t_n - y(\mathbf{x}_n)|$ として定義される。したがって分類境界上にあるデータについては、 $\xi_n = 1$ 、誤分類されたデータについては $\xi_n > 1$ が成り立つ。 $0 < \xi_n < 1$ の値を持つデータは、マージン内部にあるが正しく分類されていることを示している。このスラック変数を用いて(3.23)式を以下のように修正する。

$$t_n(\mathbf{w}^\top \phi(\mathbf{x}) + b) \geq 1 - \xi_n, \quad n = 1, \dots, N \quad (3.25)$$

目的は、マージン境界に対して誤った分類をしたデータに対してペナルティを与えながら、マージンを最大化することにある。そこで、以下の様に(3.24)式の目的関数を修正する。

$$\min_{\mathbf{w}, b, \xi} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{n=1}^N \xi_n \quad (3.26)$$

$C$ はスラック変数を用いて表されるペナルティとマージンの大きさの間をトレードオフを制御するパラメータである。 $C$ が大きいほど誤分類に対してペナルティが大きくなり、 $C \rightarrow \infty$ ではすべてのデータが線形分離可能なハードマージンSVMの最適化問題と同等となる。ハードマージンSVMに対し、「ソフトに」ペナルティを与えることからソフトマージンSVMと呼ばれる。

## 3.6 評価実験

一般的にSVMに使用する特徴量は高次元になることが多い。この問題に対し線形カーネルを用いて計算すると、内積の計算量が大きくなり処理時間がかかる。通常は多項式カーネルやRBF (Radial Basis Function, ラジアル基底関数) カーネルの計算に置き換えて計算する手法が用いられる (カーネルトリック) [56] [57]。今回の実験では、RBFカーネルを用いて評価を行う。RBFカーネルは $\gamma$ をパラメータとして以下の式で表される。

$$K(\mathbf{x}, \mathbf{x}') = \exp(-\gamma \|\mathbf{x} - \mathbf{x}'\|^2) \quad (3.27)$$

ソフトマージンSVMで使用するパラメータ $C$ とRBFカーネルのパラメータ $\gamma$ が判別能力に影響をあたえるため、 $C, \gamma$ を変化させ10-fold cross validationを

行い、訓練データに最適な $C, \gamma$ を決定する。

### 3.6.1 グレースケール画像とパワースペクトル画像に対する評価と考察

3.4.2 項で得られたリード群の判別領域画像とそのパワースペクトル画像から HLAC および LBP、LTP を用いて特徴を抽出し、10-fold cross validation の精度を測定することで周波数領域からの特徴抽出性能とグレースケール画像からの特徴抽出性能の比較を行う。実験で使用したのは、3.4.2 項で得た Positive サンプル 444 枚、Negative サンプル 737 枚である。結果を図 3-18 に示す。

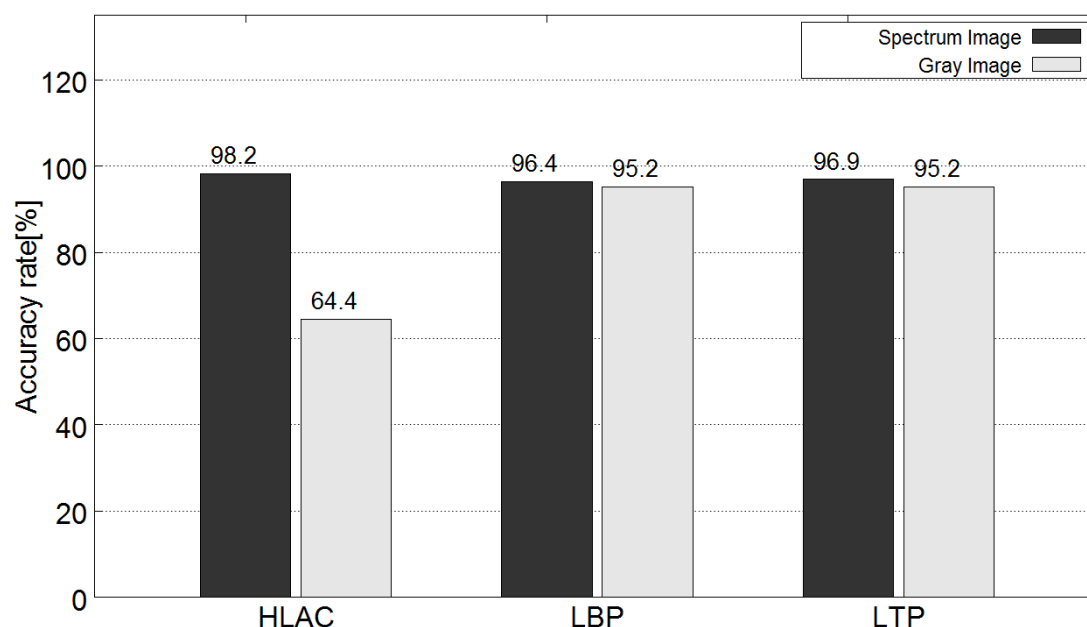


図 3-18 リード判別領域画像に対する判別性能

この結果より、HLAC、LBP、LTP のいずれを用いた方法でも周波数領域での特徴を得ることが有効であることがわかる。また HLAC と LBP および LTP を比較をすると、スペクトル画像に対する正解率は、HLAC を用いた場合に 98.2[%]で最も高く、LBP を用いた場合(96.4[%])、LTP を用いた場合(96.9[%])よりも高い正解率が得られた。HLAC 特徴量は 35 次元であり、LBP/LTP の 256 次元に比べ低い次元数での特徴量で高い正解率が得られている。またこの時のサポートベクターマシンにより判別境界として使用されたサポートベクターの数の比較を行うと図 3-19 の結果が得られた。

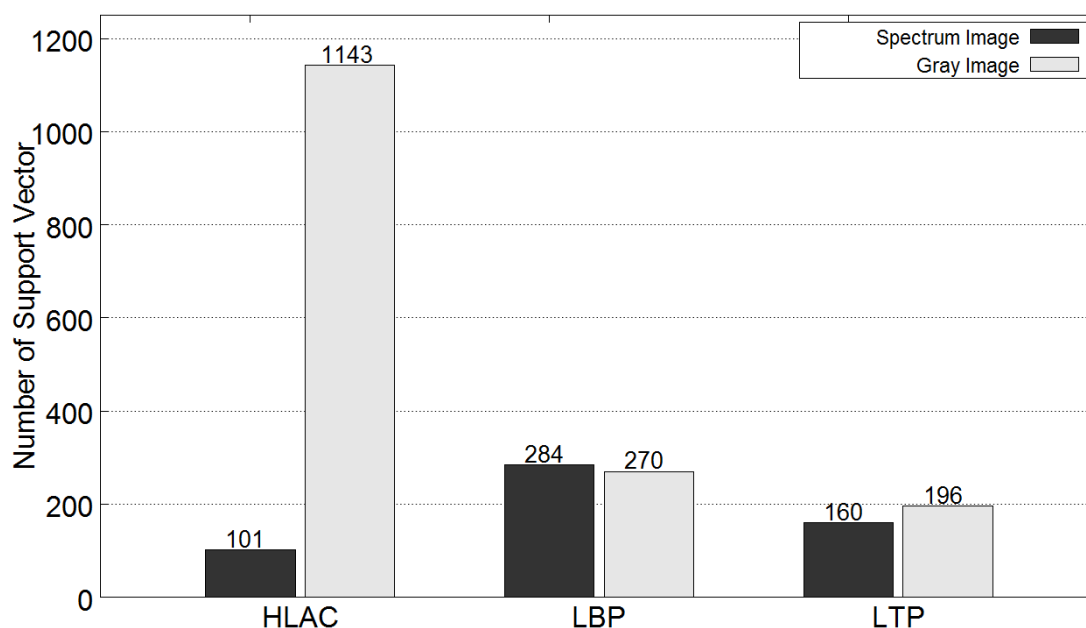


図 3-19 サポートベクターの数の比較

このグラフからも分かるように、パワースペクトル画像から HLAC 特徴を抽出した場合に最も少ないサポートベクターで分類できていることから、パワースペクトル画像に HLAC 特徴を用いた場合がより効果的であることがわかる。

### 3.6.2 リード幅、ピッチに対するロバスト性評価と考察

3.4.2 項で述べたように、パワースペクトル画像中の縦縞の出現位置はリード間ピッチや本手法における画像伸長により変化する。またリードの幅の変化に対してもパワースペクトル画像への影響があると考えられる。そこでリードの特性（幅、ピッチ）の違いへのロバスト性評価のため、まずリード幅、リード間ピッチで Positive サンプルを k-means クラスタリングを行い 3 クラスへ分類し、それぞれのクラスに対して作成した判別器を他クラスへ適用した。それらの正解率を求めることでリード特性の違いへの有効性を確認する。クラスタリングの結果得られた各クラスのリード間ピッチの平均を求めるとクラス番号順に、20.3[pixel]、56.4[pixel]、132.2[pixel]となり、クラス1とクラス3の間には、リード間ピッチで約 6.5 倍の差があることがわかる。クラスタリング結果を図 3-20 に示す。

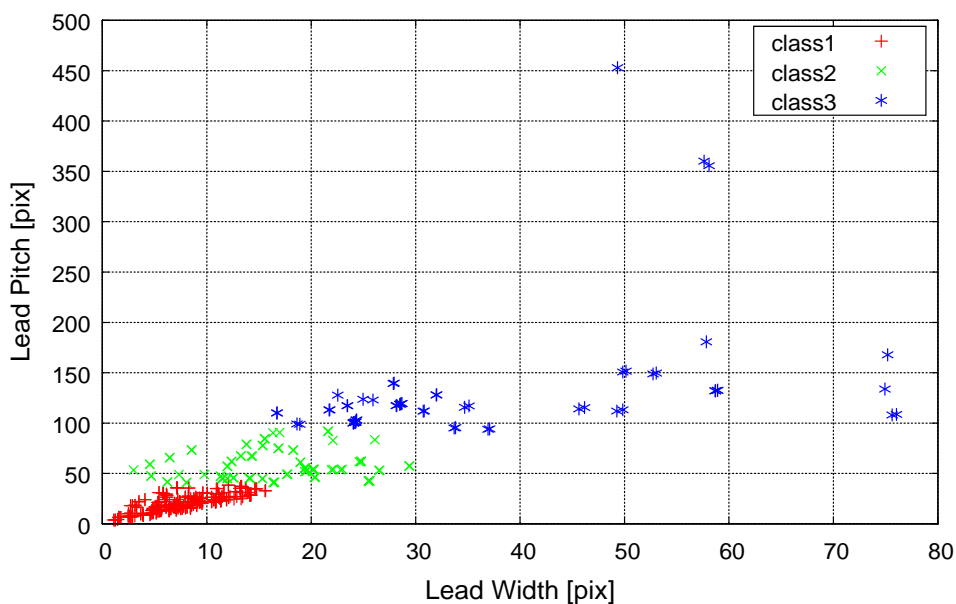


図 3-20 リード幅、ピッチによるクラスタリング結果

次に各クラスの Positive サンプルを使用し、SVM により 3 つの分類器を作成し、それぞれ別のクラスに対する True Positive Rate を計測した。3.6.1 の実験結果より、LTP に関しては LBP と大きな差が認められなかったため、HLAC と LBP についてのみ検証した。計測結果を図 3-21、図 3-22 に示す。これらのグラフから分かるように、HLAC では他クラスに対しても高い判別性能があるのにもかかわらず、LBP の場合は自クラスへの正解率が高いものの、他クラスへの True Positive Rate が低いことがわかる。LBP は 256 次元の特徴量を得ることができるが、各クラスの局所解に特化した判別器が生成された可能性がある。

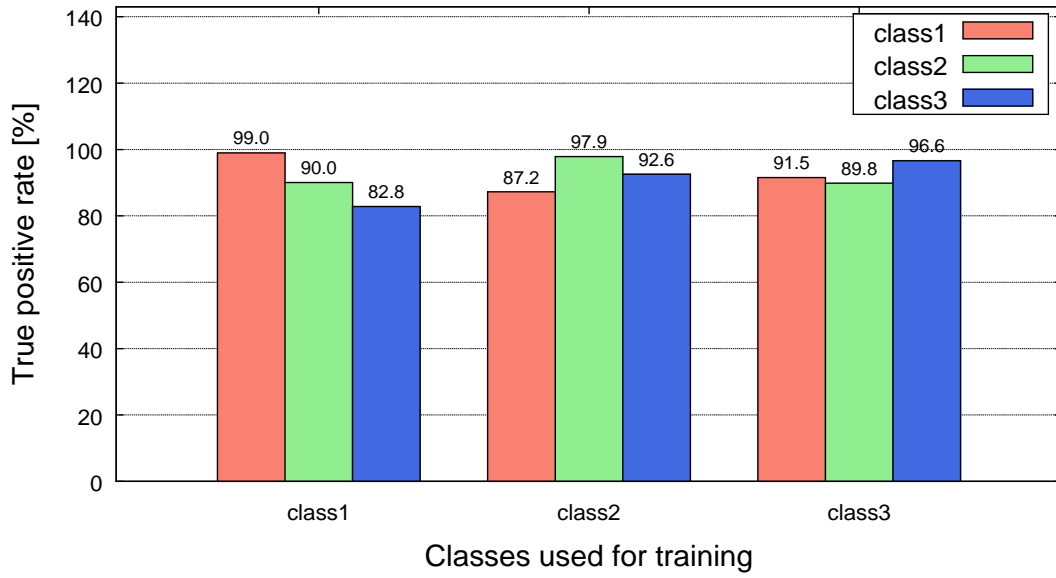


図 3-21 HLAC を使用した場合のクラス毎の True Positive Rate

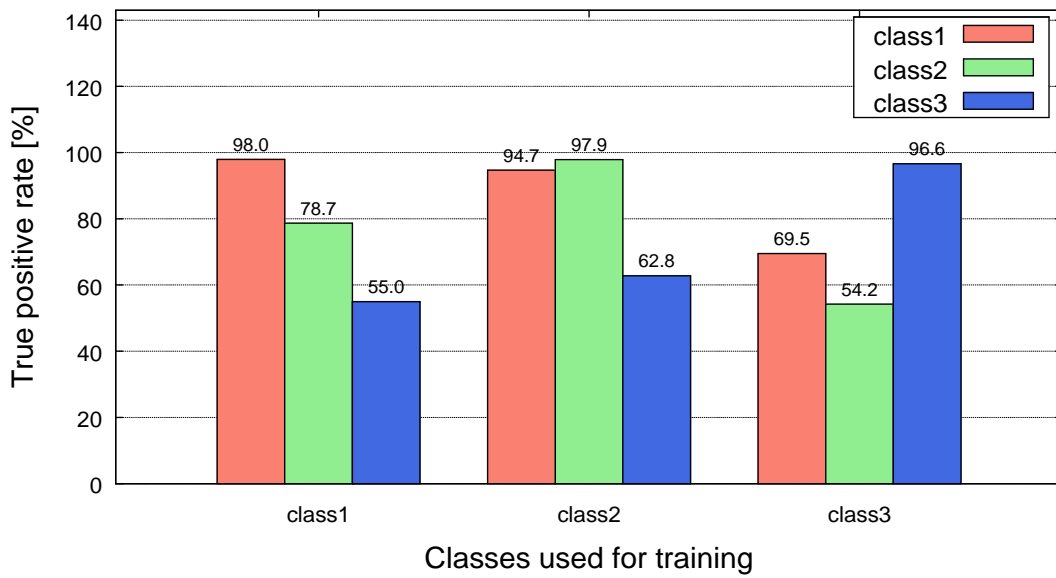


図 3-22 LBP を使用した場合のクラス毎の True Positive Rate

### 3.6.3 特徴量抽出時間

リード群判別領域画像のパワースペクトル画像に対して HLAC 特徴と LBP 特徴を用いた場合の、特徴抽出時間を計測した。計測に使用した画像は、従来手法で異型部品と分類された 310 画像で、計測に使用したのは Core i7@2.1GHz の PC である。結果を図 3-23 に示す。

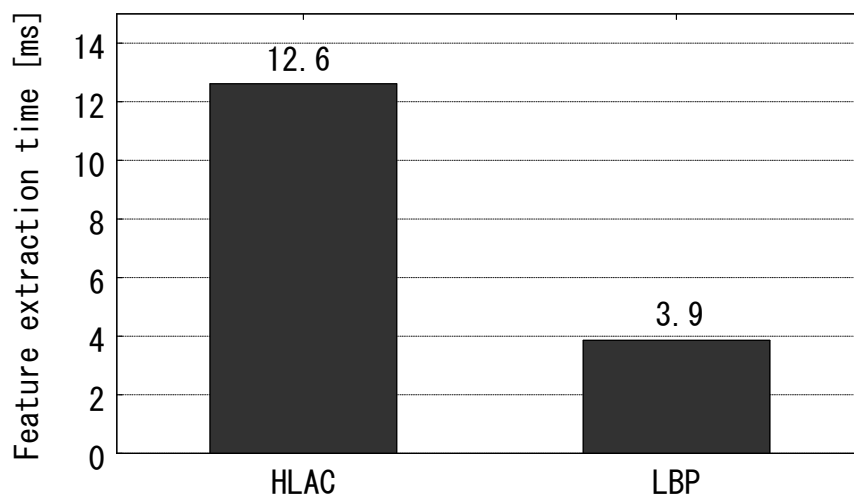


図 3-23 特徴抽出時間

この結果から、HLAC は LBP よりも特徴抽出に約 3 倍の時間がかかることがわかる。しかし、抽出時間は 12.6[ms]程度と体感速度としてはほぼ増加がわからない程度であるため、問題にはならないと考えられる。最終的に検出性能の高い HLAC を使用した手法を分類器に組み込み、分類器としての処理速度を計測する。

### 3.6.4 自動分類器への組み込み

これまでにコネクタ部品か否かを判別するための特徴抽出法について述べた。この手法を従来手法の分類器に組み込み、電子部品の自動分類器としての分類性能の改善を行う。処理フローを図 3-24 に示す。従来手法で異型部品と分類された部品画像に対し、提案手法を用いて特徴抽出を行い、SVM を用いた判別器で Positive と判別された画像に対してはコネクタ部品として分類する処理（再分類処理）を追加した。

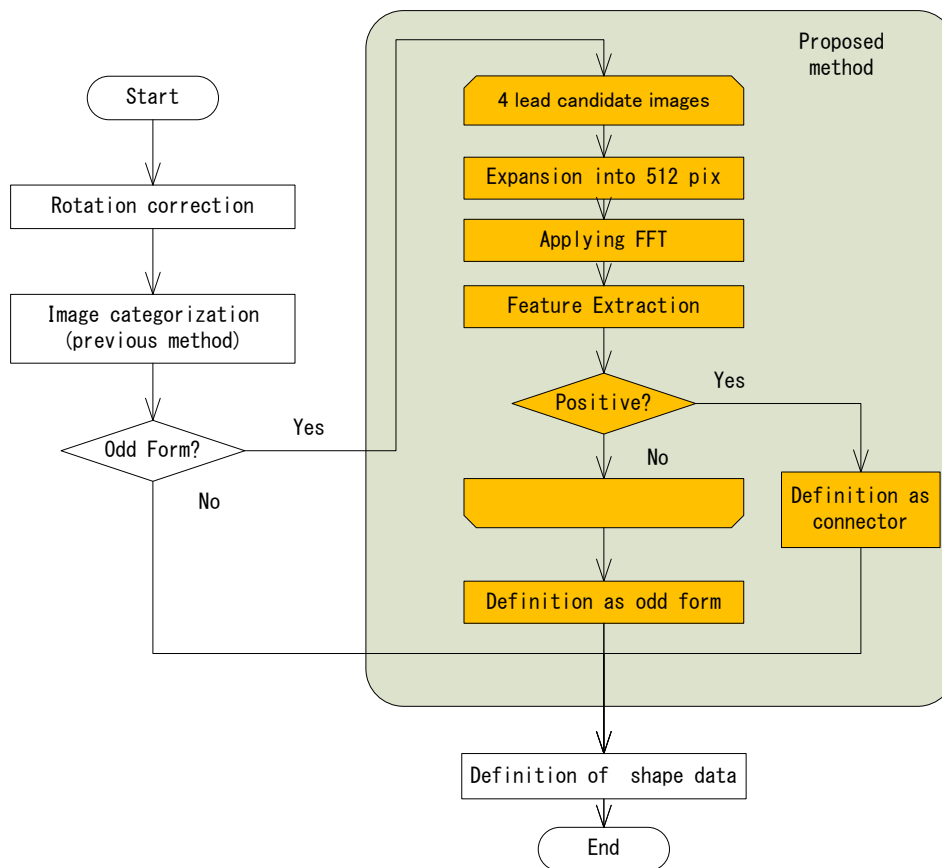


図 3-24 処理フロー

### 3.6.5 提案手法を用いた自動分類器の評価と考察

3.6.4 項で示した分類器を用いた分類結果を表 3-2 に示す。再分類処理では 1 画像あたり 4 方向のリード群判別領域画像が得られ、それらのいずれか一つが Positive (リード群あり) と判別された場合はコネクタ部品として分類する。



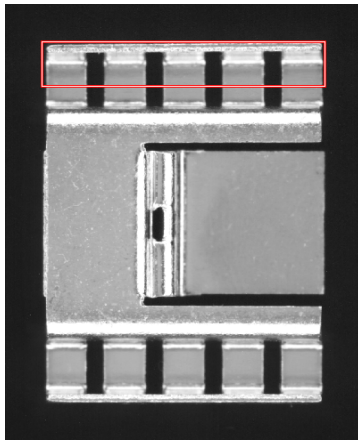
表 3-2. 提案手法による分類結果

Part categories		Result					Total	Accuracy rate [%]	
		Chip	Minimold	IC	BGA	Connector			Odd form
Input	Chip	162					1	163	99.4
	Minimold	4	139				3	146	95.2
	IC			100			1	101	99.0
	BGA	2			104			106	98.1
	Connector			4		105	6	115	91.3
	Odd form		1			8	108	117	92.3
Total								748	96.0

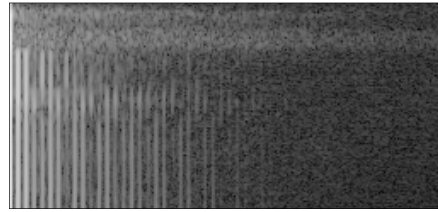
提案手法によりコネクタ部品において正解率が 40.0[%]から 91.3[%]に向上している。IC 部品への誤分類は依然として残されているが、IC 部品およびコネクタ部品で使用される位置決め処理は両方共リードを元に位置決め、検査をするため、この誤分類は問題にならない。

一方で本来異型部品である部品に対し、コネクタ部品と誤分類するケースが増加した。誤分類した部品の例を図 3-25、図 3-26 に示す。この原因は、異型部品の部品領域外周部に周期的に並ぶ形状が現れるものが存在していたためであり、このような周期的な形状が現れる異型部品に対しては誤分類してしまう。このような部品に関しては、後工程でのリードの識別処理において非リードであると認識されるため、異型部品として分類することができる。

また、コネクタ部品に関して 8.7[%]の異型部品への誤分類が残っている。誤分類した画像の例を図 3-27、図 3-28 に示す。図 3-27 中からわかるようにリード群はリード群判別領域画像中の半分程度の領域にしか含まれておらず、周期的な縦縞模様の特徴が低下したためと考えられる。この誤分類については、リード群判別領域の見直しなど改善する余地がある。図 3-28 では検出すべきリード群がリード群判別領域中にもあるにも関わらず誤分類している。この例におけるリードはその配置が周期的ではなく、周期的な特徴を捉えることが出来ずに誤分類している。周期的ではないにせよ同一形状のリードが配置されているため、周期的にリードが配置されていない部品に関しては、形状の類似性などを考慮した検出方法を検討する必要がある。

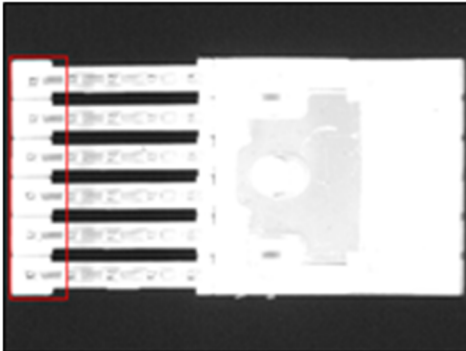


(a) 部品画像全体  
(赤枠: リード群判別領域)

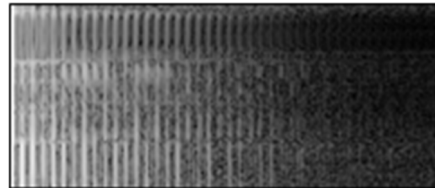


(b) パワースペクトル画像

図 3-25 False Positive の例

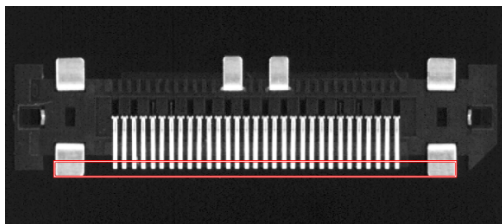


(a) 部品画像全体  
(赤枠: リード群判別領域)



(b) パワースペクトル画像

図 3-26 False Positive の例



(a) 部品画像全体  
(赤枠: リード群判別領域)



(b) パワースペクトル画像

図 3-27 False Negative の例

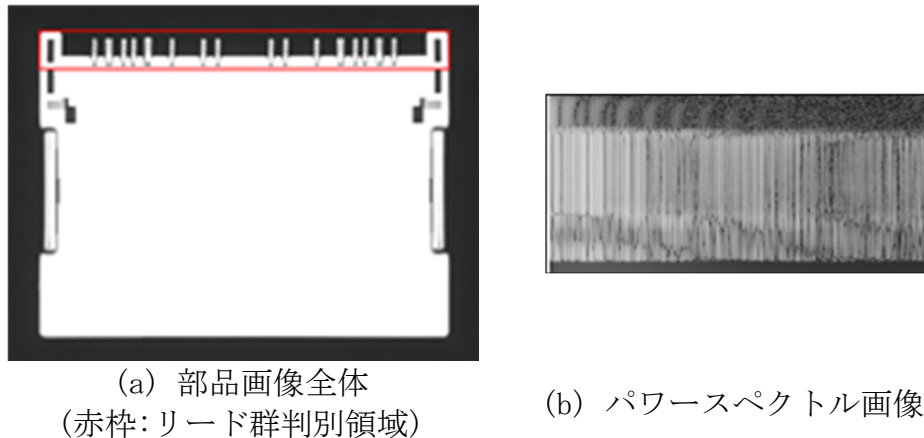


図 3-28 False Negative の例

再分類処理の処理時間と従来の分類処理の処理時間の比較を表 3-3 に示す。使用したコンピュータは特徴量抽出時間を計算したものと同様に、Core i7@ 2.1GHz の CPU を搭載している。この処理時間には、HLAC 特徴量の抽出時間だけでなく、パワースペクトル画像を生成する時間も含まれている。従来処理では分類処理に平均で 289[ms]かかっていた。この処理に提案手法による再分類処理を追加しても平均で約 49[ms]の増加に過ぎず、このシステムを使用する作業員の体感速度としてはほぼ変わらず、運用上問題にはならない。

表 3-3. 再分類処理に要した時間

	従来処理 (748 samples)	再分類処理 (186 samples)
Average [ms]	289.0	49.1
Max [ms]	2509.3	236.6
Min [ms]	11.0	2.5

### 3.7 まとめ

本章では従来手法で分類性能が低いコネクタ部品に焦点を絞り、コネクタ部品の持つ特徴に着目した手法を提案した。はじめに部品画像中からリードの存在する可能性が高い領域を抽出し、リード群の判別領域を特定した。次にこの領域から得られた画像に対し、FFT を適用しパワースペクトル画像を作成した。

リード群判別領域にリード群が存在する場合は、画像中に縞模様が出現し、逆にリード群が存在しない場合には縞模様が出現しないことが確認できた。パワースペクトル画像から縞模様の出現を判別する手法として HLAC 特徴量および LBP、LTP を使用した特徴抽出を行い、サポートベクターマシンを使用して判別する方法を提案した。

実験では、パワースペクトルを抽出することの効果を検証するため、グレー画像から特徴を抽出する場合と、パワースペクトル画像から特徴を抽出する場合を比較した。HLAC/LBP/LTP のいずれの特徴量を使用した場合でもパワースペクトル画像から特徴を抽出した場合に判別性能が高いことが確認できた。また、判別境界として使用されたサポートベクター数についても比較した。その結果 HLAC 特徴を使用した場合が少ないサポートベクター数で高い判別性能が得られたことが確認できた。HLAC 特徴は LBP/LTP に比べて次元数も低く、この識別には有効な特徴量であるといえる。

次に、リード幅、ピッチに対するロバスト性を検証した。実験ではリードの幅、ピッチから k-means クラスタリングを行い 3 クラスへ分類し、それぞれのクラス内のサンプルを用いて判別器を作成し、他のクラスへ適用させて判別性能を HLAC と LBP で比較した。その結果、HLAC 特徴を用いた場合のほうが他のクラスへの判別性能が高いことが確認できた。HLAC は LBP に比べ少ない次元数の特徴量だが、パワースペクトル画像から効果的な特徴を抽出出来たといえる。

最後にこの判別器を従来手法の自動分類器の処理フローに組み込み、部品自動分類器の分類性能向上を行った。先行研究[16][17]で異型部品と分類された部品画像に対し、提案手法を用いることでコネクタ部品に対する正解率が 40.0[%]から 91.3[%]に向上することを示した。部品全体としては、89.2[%]から 96.0[%]に正解率を向上することが出来た。従来手法に提案手法を加えても平均で約 49[ms]の増加であり、システムを使用する作業員の体感速度はほぼ変わらず問題にはならない。

しかしコネクタ部品であるにもかかわらず、リード群が部品の中央部に存在する部品に対し、異型部品と誤分類するケースが残されている。リード群判別領域の見直しや、リードの形状認識を組み合わせるなどの方法でより精度の高い分類器を構築することが今後の課題である。

# 第4章 リード部品形状データ自動生成

本章では、リード部品の形状データの自動生成手法について述べる。

## 4.1 まえがき

第3章でも述べたように、電子部品をマウンタでプリント基板上に装着するためには、部品種ごとに応じた画像処理アルゴリズムと部品の形状データを予め用意しておく必要がある。第3章では画像処理アルゴリズムを決定するために、部品種の自動判別のための特徴抽出について述べた。形状データは、抵抗器やコンデンサの代表的な形状である角チップの場合、図4-1(a)に示すような矩形形状で定義され、矩形のサイズ (SizeX, SizeY) が必要な形状データとなる。ICやコネクタに代表される、リードという短冊状の電極を持つリード部品(図4-1(b))では、リードの位置、幅、長さおよび間隔(ピッチ)が必要な形状データとなる。この形状データの定義を行うことで、部品をプリント基板上へ装着する前に位置決め処理および、リードの欠損、曲がりを現行の画像処理システムを用いて測定することができる。

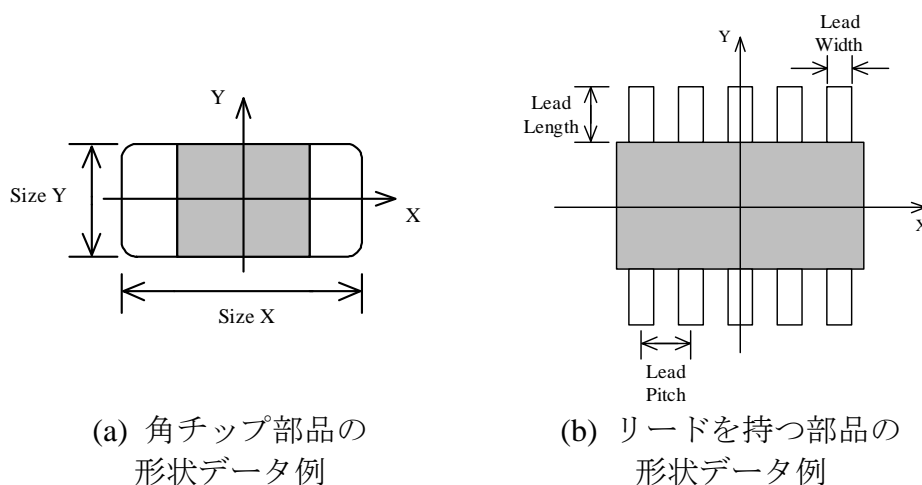


図 4-1 形状データの定義例

これらの形状データは多くの場合、専門の人員が手作業で作成している。紙面による設計図データがある場合には、データを手入力することで形状データ

を作成する。機密保持等の理由により設計仕様書が入手できない場合や、現品のみが発注元からの支給により提供されるような場合には、部品や部品を撮像した画像をノギスなどで採寸することで、形状データを作成する。形状データ作成には、多くの工数がかかるため自動化が強く求められている。

本章では、1枚の良品サンプル画像から表面実装部品のリード部分の形状データを自動生成することを目的とする。特に表面実装部品の最も代表的な電極の一種であるリードは、図 4-2、図 4-3 に示すように様々な部品で使用されており、その位置やサイズは高精度な位置決め処理には重要である。特に第 3 章で述べたコネクタ部品では、リードを電極として用いた部品が多い。これまでに江寄らによる研究で、IC のリード形状を自動的に生成するシステム[58]が提案されている。これらのシステムでは図 4-2(a) (b) に示す一般的な形状の IC 部品において効果的な結果が得られている。従来手法では、画像全体からリードを 1 本ずつ検出しているわけではなく、自動分類システム[16]による自動分類手法の分類結果に基づいた事前情報を利用してリードの形状データを生成している。しかし、図 4-3(a) (b) に示されるような複雑な形状の部品では形状データが生成できない。形状データが生成できない理由として、従来手法[58]では図 4-2(a) (b) のようにリード 1 本ずつが独立して、画像中の輝度の高い領域として撮像されることを前提としているためである。

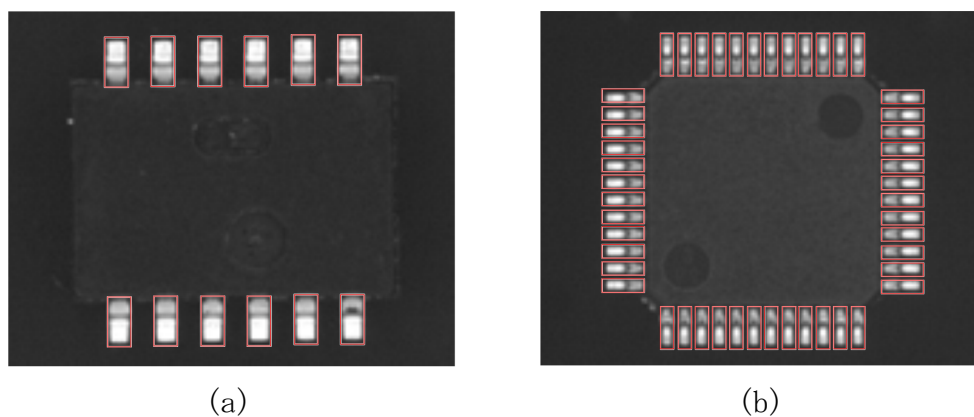


図 4-2 一般的な IC 部品の例 (赤枠：リード)

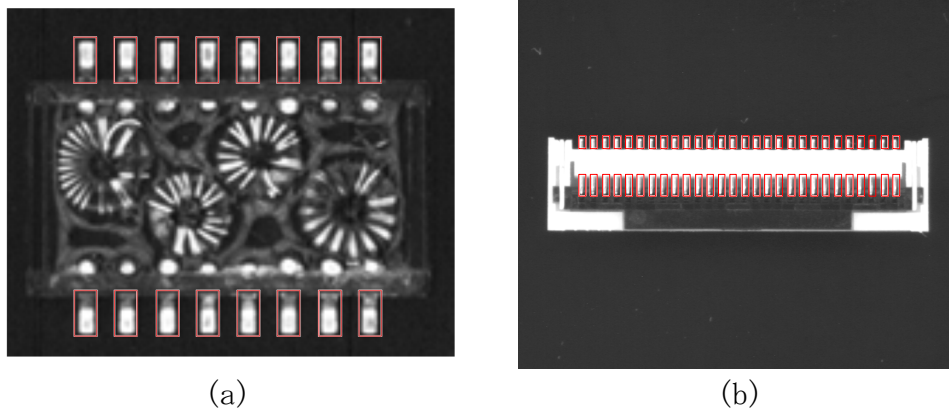


図 4-3 複雑なリード部品の例 (赤枠：リード)

本章では画像中からリードそのものを直接検出する手法を用いて、リード形状データを生成する手法を提案する。リードを検出する手法は、顔検出や人検出などにおいて使用されている AdaBoost を用いた検出方法[5]–[8], [10]を応用した。さらにリード電極の配置が持つ周期性に注目し、周期的な配置のある領域に限定してリードを検出することで有効な形状データが生成できることを示す。周期性の判断は音声関係の研究で用いられるピッチ検出の手法[59][60]を応用した。

## 4.2 AdaBoost を用いたリード先端抽出

本章で扱う検出対象リードは、一般的に短冊状の電極形状をしている。電極の長さ、幅、ピッチ、先端形状は様々であり、電極らしいものを検出するために顔認識や人認識で効果的な手法である AdaBoost を用いた手法を応用した。はじめに AdaBoost の基本的な理論から述べる。

### 4.2.1 AdaBoost

Boosting アルゴリズムは、教師あり学習を行うための機械学習アルゴリズムであり、あまり性能の高くない単純な識別器（弱識別器）を複数用意し、それらをまとめることでより性能の高い識別器（強識別器）を生成するアルゴリズムである。その中でも AdaBoost (Discrete AdaBoost) は Y. Freund と R. Schapire により提案された機械学習アルゴリズム[61]であり、顔検出や人検出というア

アプリケーションで使用されている [5]–[8], [10] [62]。

Real AdaBoost は Discrete AdaBoost を発展させた学習アルゴリズムである。Discrete AdaBoost の弱識別器の出力が、 $\{+1, -1\}$  という 2 値の出力であるのに対し、Real AdaBoost では弱識別器の出力が特徴量の分布に応じて実数値化 (real) しているという特徴がある [63]。Discrete AdaBoost のアルゴリズムを図 4-4 に示す。

1. 初期化 以下の  $N$  個のサンプル  
 $(x_1, y_1), \dots, (x_N, y_N)$   
*where*  $x_i \in X, y_i \in Y = \{+1, -1\}$
2. 学習サンプルの重み  $D_1$  の初期化  

$$D_1(i) = \frac{1}{N}, i = 1, 2, \dots, N$$
3. For  $t = 1, 2, \dots, T$ :  $T$  回の学習  
 For  $m = 1, 2, \dots, M$ :  $M$  個の弱識別器
  - ① 弱識別器  $h_t$  を学習する  

$$h_t = \begin{cases} 1 & u \cdot v(x) > u \cdot \theta \\ 0 & \text{otherwise} \end{cases}$$
 学習サンプルに対する誤り率、  

$$\epsilon_t = \sum_{i: h_t(x_i) \neq y_i}^m D_t(i)$$
 が最小となる  $h_t(x)$  を選択する
  - ② 誤り率から信頼度  $\alpha_t$  を算出する  

$$\alpha_t = \frac{1}{2} \log \left( \frac{1 - \epsilon_t}{\epsilon_t} \right)$$
  - ③ サンプルの重みを更新する  

$$D_{t+1}(i) = D_t(i) \exp[-\alpha_t y_i h_t(x_i)]$$
  - ④ サンプルの重みの和が 1 になるように正規化  

$$D_{t+1}(i) = \frac{D_{t+1}(i)}{Z_{t+1}}$$
 ここで、  

$$Z_{t+1} = \sum_{i=1}^m D_{t+1}(i) = \sum_{i=1}^m D_t(i) \exp[-\alpha_t y_i h_t(x_i)]$$
4. すべての弱識別器を信頼度で重み付けして強識別器を作成  

$$H(x) = \text{sign} \left( \sum \alpha_t h_t(x) \right)$$

図 4-4 Discrete AdaBoost アルゴリズム



$u$ は不等号の向きを決定する符号であり、+1 もしくは -1 をとる。 $v$ はサンプル $x$ における特徴量であり、 $\theta$ は閾値である。最終的に強識別器 $H(x)$ は各弱識別器の重み付けの多数決を取る形となる。

Real AdaBoost は Discrete AdaBoost とは異なり、弱識別器の出力が+1, -1ではなく、確率密度分布 $W_+$ ,  $W_-$ の差に応じて連続量を出力することが可能になる。図 4-5 に Real AdaBoost のアルゴリズムを示す。

1. 初期化 以下の  $N$  個のサンプル  
 $(x_1, y_1), \dots, (x_N, y_N)$   
 where  $x_i \in X, y_i \in Y = \{+1, -1\}$
2. 学習サンプルの重み  $D_1$  の初期化  

$$D_1(i) = \frac{1}{N}, i = 1, 2, \dots, N$$
3. For  $t = 1, 2, \dots, T$ :  $T$  回の学習  
 For  $m = 1, 2, \dots, M$ :  $M$  個の弱識別器
- ① 弱識別器  $h_t$  の確率密度分布  $W_+, W_-$  の作成  

$$W_+^j = \sum_{i: j \in J_i^y = +1} D_t(i)$$

$$W_-^j = \sum_{i: j \in J_i^y = -1} D_t(i)$$
- ② 弱識別器候補  

$$h(x) = \frac{1}{2} \ln \left( \frac{W_+^j + \varepsilon}{W_-^j + \varepsilon} \right)$$
 弱識別器選択のための評価値を算出  

$$Z_m = 1 - 2 \sum_j \sqrt{W_+^j W_-^j}$$
- ③ 最も  $Z$  が大きい弱識別器を選択  

$$h_t = \arg \max Z_{t,m}$$
- ④ サンプルの重みを更新する  

$$D_{t+1}(i) = D_t(i) \exp[-\alpha_t y_i h_t(x_i)]$$
- ⑤ サンプルの重みの和が 1 になるように正規化  

$$D_{t+1}(i) = \frac{D_{t+1}(i)}{Z_{t+1}}$$

$$Z_{t+1} = \sum_{i=1}^m D_{t+1}(i) = \sum_{i=1}^m D_t(i) \exp[-y_i h_t(x_i)]$$
4. すべての弱識別器を信頼度で重み付けして強識別器を作成  

$$H(x) = \text{sign} \left( \sum_{t=0}^T \alpha_t h_t(x) \right)$$

図 4-5 Real AdaBoost アルゴリズム

Discrete AdaBoost とは異なり、弱識別器の重み  $\alpha$  が存在せず、弱識別器の

線形和となっている。3ステップの②における $\epsilon$ は分母が0になることを防ぐ係数である。強識別器 $H(x)$ は Discrete AdaBoost と同様、各弱識別器の重み付けの多数決を取る形となる。

## 4.2.2 カスケード構造

AdaBoost を用いた識別器の高速化手法として、Viola&Jones の手法ではカスケード構造を使用した手法が提案されている [10] [64]。

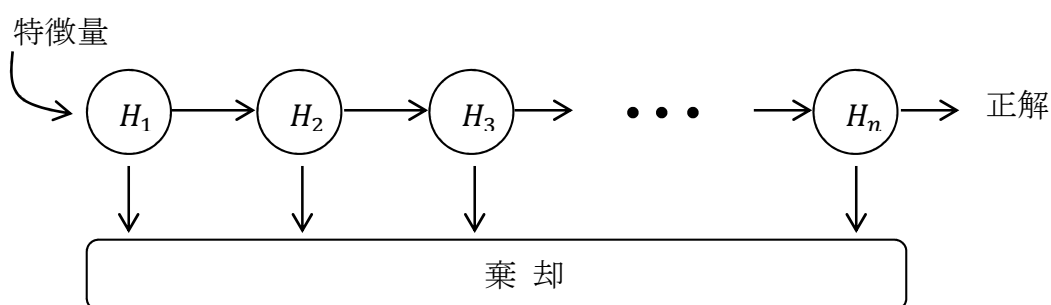


図 4-6 カスケード構造

前段の強識別器は、できるだけ早い段階で明らかに不正解となる画像を棄却するための仕組みである。図 4-6 における強識別器 $H_1, H_2$ の役割は、できるだけ少数の弱識別器を用いて単純な背景画面等を棄却することが求められる。少数の弱識別器で棄却できればよく、少ないコストで正解を検出することが可能となる。

## 4.2.3 Haar-Like 特徴量

Haar-like 特徴は高速に顔検出ができる特徴量として知られている [10] [62]。この特徴量は、人の顔の目、口、頬、眉の相対的な輝度差を表す特徴量として用いられており、単純な矩形領域の輝度差を特徴量とする (図 4-7)。

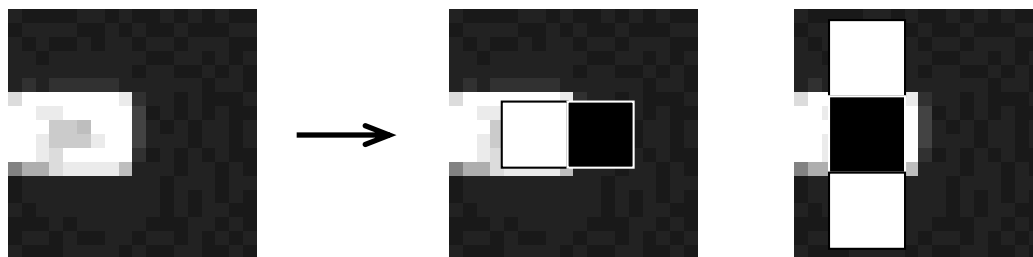


図 4-7 リード画像から選択される Haar-Like 特徴の例

Haar-Like 特徴は、図 4-7 に示すような特徴的な 2 つの領域、白領域 $r_1$ と黒領域 $r_2$ の平均輝度の差を(4.1)式より算出し特徴量とする。

$$Haar(r_1, r_2) = S(r_1) - S(r_2) \quad (4.1)$$

ここで $S(r)$ は、領域 $r$ の平均輝度を算出する関数である。2 つの領域パターンには様々なものが提案されているが、一般的には図 4-8 に示すパターンが用いられることが多い。これらのパターンは縦方向の輝度差、横方向の輝度差、中央と周辺の輝度差等を捉えることが可能なパターンである。他にもこのパターンを  $45^\circ$ 回転させたパターンも紹介されている[65]が、リード検出のために用いる Haar-Like 特徴パターンは一般的に使用されるものと同様、図 4-8 に示した 8 パターンを使用した。以下の 8 パターンを  $18 \times 18$ [pixel]の画像に適用して特徴量を抽出した場合、66618 次元の特徴量が得られる。

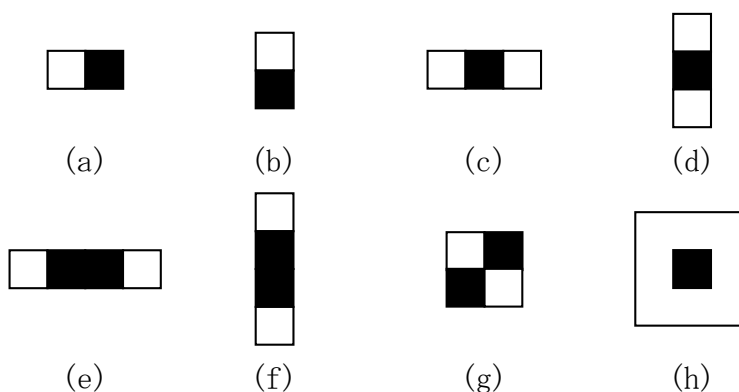


図 4-8 使用した Haar-Like 特徴パターン

Haar-Like 特徴を使用した物体検出では、検出ウィンドウを対象画像に対してラスタスキャンさせ、検出ウィンドウより得られる特徴量を用いて検出を行う。そのため非常に多数の検出ウィンドウから特徴量を算出することになり、高速に特徴量算出を行う方法が必要となる。高速化の手法として積分画像を使用する方法が提案されている [64]。

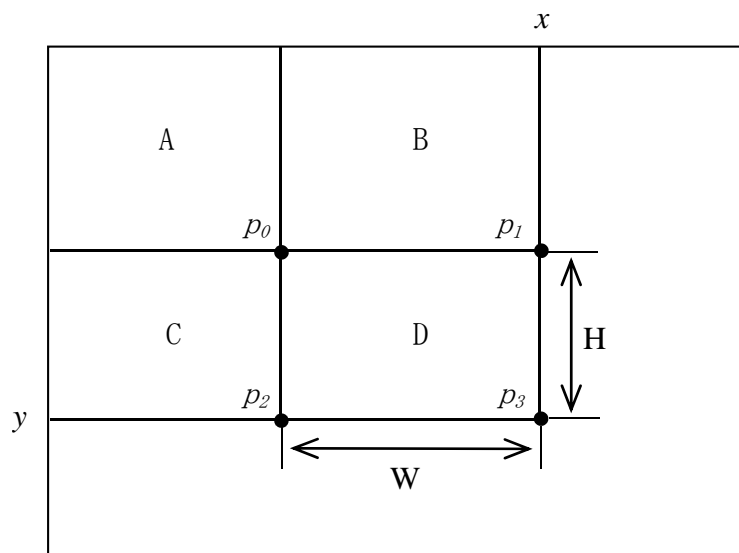


図 4-9 積分画像

積分画像は画素の値が、画像の左上からの輝度値を累積した値であり、入力画像  $i(x, y)$  から求められる積分画像は (4.2), (4.3) 式で表される。

$$s(x, y) = s(x, y - 1) + i(x, y) \quad (4.2)$$

$$ii(x, y) = ii(x - 1, y) + s(x, y) \quad (4.3)$$

$s(x, y)$  は行の画素の総和、 $ii(x, y)$  は列の画素の総和を表す。 $(x, y) = (0, 0)$  の場合に画像の範囲外へのアクセスとなるため、 $s(x, -1) = 0, ii(-1, y) = 0$  とする。この時、図 4-9 における D の領域の平均輝度値を求める場合は (4.4) 式より算出することができる。

$$S(D) = \frac{(ii(x, y) + ii(x - W, y - H)) - (ii(x - W, y) + ii(x, y - H))}{W \times H} \quad (4.4)$$

積分画像は対象画像の検出処理を開始する前に 1 度だけ作成すればよく、高速化が期待できる。

#### 4.2.4 HOG 特徴量

人検出によく使用される勾配ベースの特徴量として HOG 特徴量が提案されている [5]。HOG 特徴は Haar-Like 特徴のように画像の輝度ベースとした特徴量ではなく、1つの局所領域におけるエッジの方向ごとのエッジ強度に着目した特徴量である。局所領域は図 4-10 に示すようにブロックとセルで表現される。ブロックには複数のセルと呼ばれる更に小さな領域から構成され、特徴量としてはセルのエッジ方向毎のエッジ強度ヒストグラムを結合した多次元ベクトルで表現される。

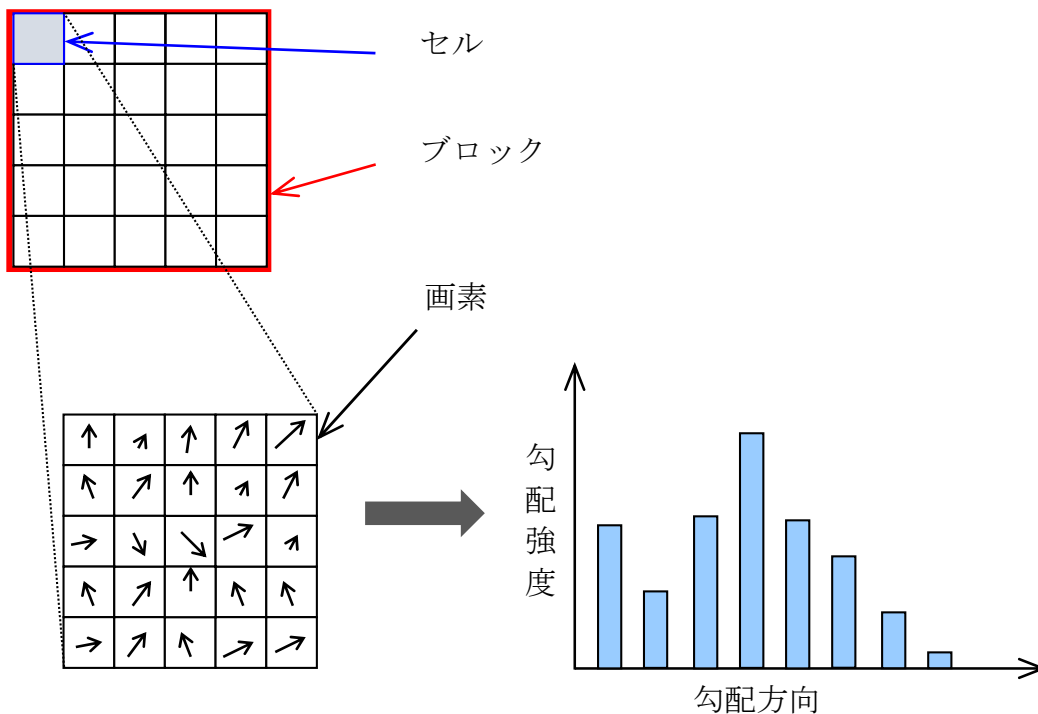


図 4-10 HOG 特徴量におけるセルとブロック

各ピクセルの輝度から、勾配強度 $m$ と勾配方向 $\theta$ を次式より算出する。

$$m(u, v) = \sqrt{f_u(u, v)^2 + f_v(u, v)^2} \quad (4.5)$$

$$\theta(u, v) = \tan^{-1} \left( \frac{f_v(u, v)}{f_u(u, v)} \right) \quad (4.6)$$

$$\begin{cases} f_u(u, v) = i(u, v - 1) - i(u - 1, v) \\ f_v(u, v) = i(u, v + 1) - i(u, v - 1) \end{cases} \quad (4.7)$$

算出された勾配強度と勾配方向を用いて 5×5 ピクセルを 1 セルとした領域において輝度の勾配方向ヒストグラムを作成する。勾配方向は 0° から 360° を 30° 単位で分割した場合、12 方向の勾配ヒストグラムが作成できる。次に、各セルで作成した輝度の勾配方向ヒストグラムを 1 ブロック毎に正規化を行う。

$$v = \frac{v}{\sqrt{(\sum_{i=0}^k v(i)^2) + \epsilon}} \quad (\epsilon = 1) \quad (4.8)$$

ここで $v$ は正規化後の特徴量であり、 $k$ はブロック内の HOG 特徴量の数、 $\epsilon$ は分母が 0 になることを防ぐための係数である。ブロックは 1 セルずつ移動させながら HOG 特徴量の抽出を行う。1 セルが 5×5 ピクセル、1 ブロック=3×3 セル、勾配方向を 30° 間隔(12 方向)で 40×40 ピクセルの画像を対象に特徴量を抽出した場合、 $(40/5-3+1) * (40/5-3+1) * (3*3) * 12 = 3888$  次元の特徴量が得られる。人検出の先行研究では 20° 間隔で 9 次元の特徴量、つまり服と背景の輝度勾配の方向が逆転していても不変の特徴量とするためにあえて 0° ~ 180° の範囲としていたが、リード検出では背景画像はコントロールできるため勾配方向の逆転を考慮する必要が無い。また対象画像は回転補正してあるため、リードの方向は 0°、90°、180°、270° のいずれかの方向に向いている。20° 間隔では 90°、270° では割り切れないため、30° 間隔で 0° ~ 360° の 12 次元の勾配方向特徴量を使用した。HOG 特徴も Haar-Like 特徴同様、積分ヒストグラムを用いた高速化手法が提案されている [66]。HOG 特徴では積分ヒストグラムを

用いて実験を行う。

## 4.2.5 リード検出実験

Haar-Like 特徴と HOG 特徴を使用してリード画像の判別実験を行う。実験は Discrete AdaBoost により教師画像を学習させ、画像中からのリード検出性能の比較を Haar-Like 特徴量と HOG 特徴に対して行う。教師データはリード先端位置が既知の画像からリード先端 (Positive) 画像として 2070 枚、非リード (Negative) 画像として 2879 枚を用いた。教師画像はすべて正方形であり Positive 画像はすべて回転補正し、リード先端が右向きになるようにした。リード幅については 1 辺の 1/3 になるようにして正規化して作成した。作成した教師画像例を図 4-11 に示す。

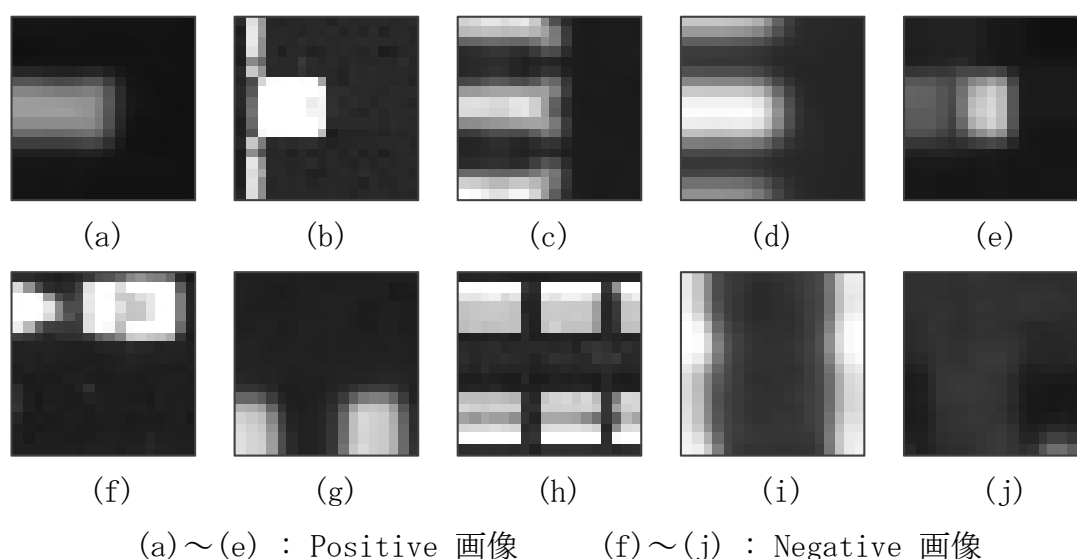


図 4-11 リード先端の教師画像の例

画像サイズは Haar-Like 特徴を使用する場合、 $18 \times 18$  [pixel]、HOG 特徴量を使用する場合は  $40 \times 40$  [pixel] とした。HOG 特徴量を使用する場合、1 ブロック ( $5 \times 5$  [pixel]) で割り切れる  $20 \times 20$  [pixel] の画像を使用したとしても、 $(20/5-3+1) * (20/5-3+1) * (3*3) * 12 = 432$  次元と Haar-Like で得られる特徴量の次元数 66618 次元と比べてあまりにも小さいため、 $40 \times 40$  [pixel] を使用した。 $40 \times 40$  [pixel] の場合、3888 次元の特徴量が得られる。

Haar-Like 特徴量と HOG 特徴量を用いて学習した Discrete AdaBoost のリー



ド検出器を用いて部品画像中からリード検出を行い、実際に定義してあるリードに対する検出率を使用して検証した。右向きリードを検出する学習器を構築しているため、上下左方向のリード検出に対応するため検出器を4方向に回転させてリード検出を行う。検出実験に使用した画像は、予めリード位置が既知のリード部品画像 226 枚である。実験結果を図 4-12 に示す。この結果からわかるように、Haar-Like 特徴量のほうが HOG 特徴量よりも高い検出率が得られている。40×40[pixel]の領域から得られる HOG 特徴量の次元数は Haar-Like 特徴に比べて少なく (1/15 程度)、検出に有効な特徴量が得られなかった可能性がある。

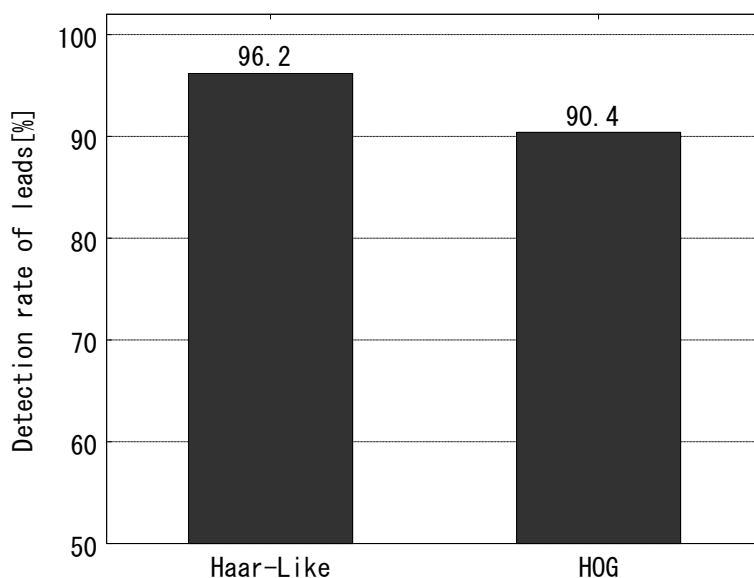


図 4-12 Haar-Like 特徴と HOG 特徴によるリード検出率

また部品画像中から Haar-Like 特徴量を使用した検出器で検出した例を図 4-13 に、HOG 特徴量を使用した検出器で検出した例を図 4-14 に示す。図中の赤枠は上向き、黄色枠は右向き、緑枠は左向き、青枠は下向きのリードを検出したことを示している。

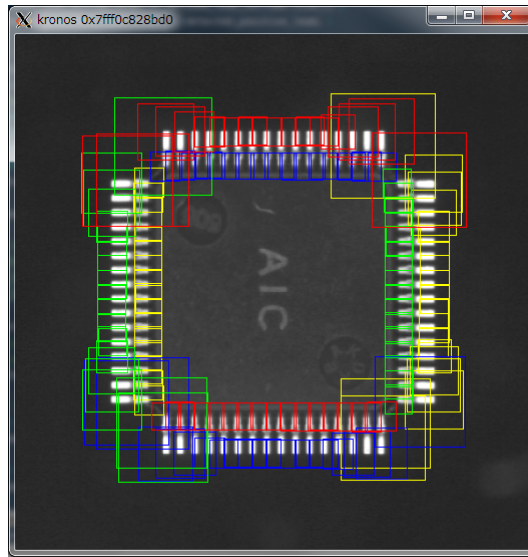


図 4-13 Haar-Like 特徴+Discrete AdaBoost によるリード検出例

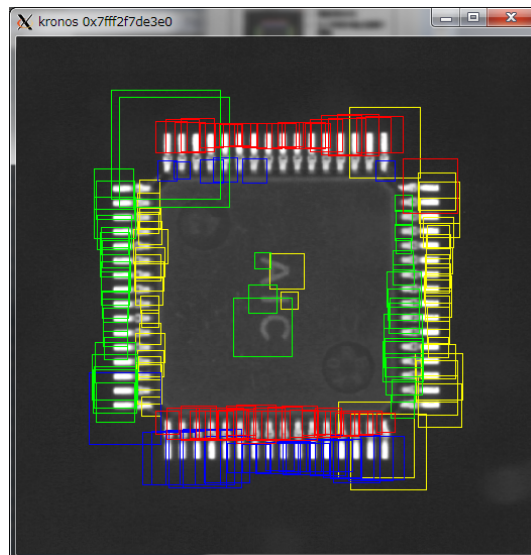


図 4-14 HOG 特徴+Discrete AdaBoost によるリード検出例

これらの結果から、Haar-Like 特徴に比べ HOG 特徴を使用した場合部品中に印刷された文字やリード群のそのものをリードとして誤検出している部分もあり、HOG 特徴よりも Haar-Like 特徴を使用したほうがリード検出には適していると考えられる。

次に Haar-Like 特徴を使用し、Discrete AdaBoost と Real AdaBoost の比較を行う。Real AdaBoost は先行研究[63]にもあるように、Discrete AdaBoost に比べ効果的な結果が得られる可能性がある。実験結果を図 4-15 に示す。この結果より、Real AdaBoost を使用した場合のほうが Discrete AdaBoost を使用

する場合よりも僅かではあるがリード先端の検出率においては良好な結果が得られた。

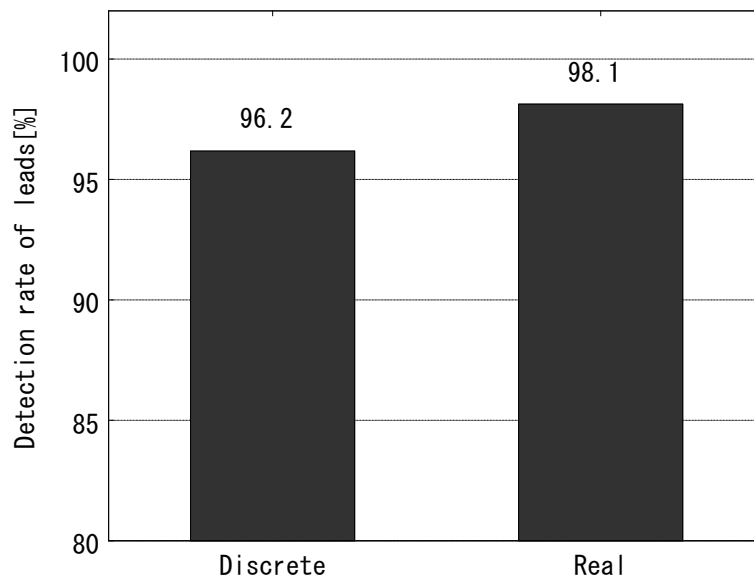


図 4-15 Discrete AdaBoost と Real AdaBoost の比較

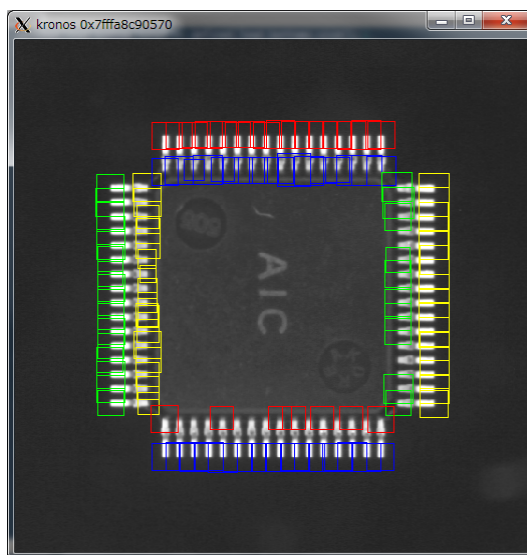


図 4-16 Haar-Like 特徴+Real AdaBoost によるリード検出例

Haar-Like 特徴、HOG 特徴ともに単純な IC 部品であってもリードの付け根部分を誤認識しているが、局所領域でのリード先端形状の検出ではリードの付け

根か先端かを見分けるのは困難であり、AdaBoost を利用した検出方法単体での改善は期待できない。リードの付け根の誤認識に関しては 4.3.7 で述べるように、リードが外側へ向いて取り付けられている性質を利用し、先端と付け根が同一のリードで検出された場合には、外側を先端とすることで回避できる。しかし図 4-17 に示すような複雑な形状をしたコネクタ部品の場合、実際にはリードではないのにもかかわらず局所的にはリードとして見える部分が多く存在する(図 4-17)。そこでリードの周期的な配置に注目して周期的な特徴量抽出によるリード群の領域限定と AdaBoost によるリード抽出手法を組み合わせることで、リード先端の誤検出を低減する手法について検討する。

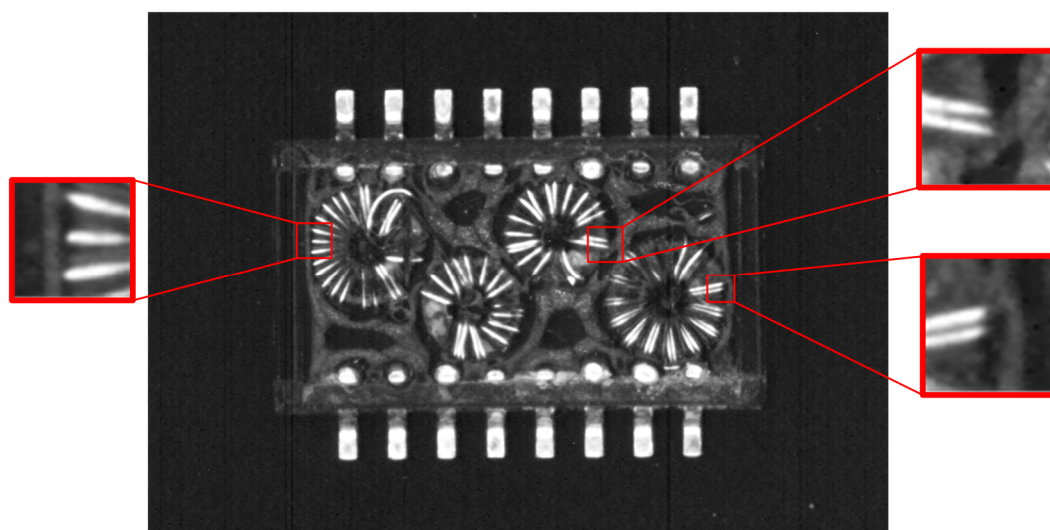


図 4-17 局所的にリード先端と判別される部分の例

### 4.3 リード群領域の抽出法

AdaBoost を用いたリード先端検出法では、リード先端の検出率は低くはない。しかし顔や人ほど特徴的なコントラスト特徴があるわけではなく、部品のフレーム部やコイルなどを誤検出するケースが多い。これはリード先端部の局所領域のみを用いて教師画像を作成し、画像中から検索する手法を使用しているためであり、局所領域だけの特徴から誤検出を削減することは困難であると考えられる。

そこで、局所領域においてリード先端か否かと判断するというアプローチとは別に、巨視的なアプローチとしてリードの並びに注目した。通常、リード部

品は同一形状のリード1本が、等間隔に数本から数十本、同一方向に向いて並んでいる。第3章の図3-4で示したリード群を、この特徴を元に画像中からリード群の領域を抽出する。そして検出されたリード群領域内に対してのみ前述のリード検出を行うことで、誤検出を低減することが可能だと考えた。概略処理フローを図4-18に示す。

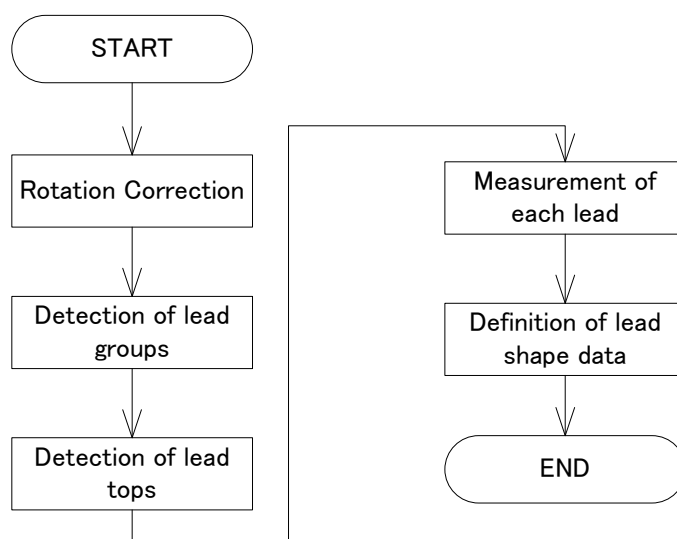


図 4-18 概略処理フロー

周期性の検出には音声の分野でピッチ検出に用いられている正規化二乗差関数 (NSDF : Normalized Square Difference Function) [59][60]を用いた方法を応用した。NSDF の算出法を以下に示す。

### 4.3.1 平均振幅差関数

平均振幅差関数 (AMDF : Average Magnitude Difference Function)は、音声認識の研究ではピッチの検出のための手法の一つとして用いられている。AMDF は以下の式によって定義される [67]。

$$D(\tau) = \frac{1}{W - \tau - 1} \sum_{j=0}^{W-\tau-1} |f(j) - f(j + \tau)| \quad (4.9)$$

$D(\tau)$ は遅れ $\tau$ における平均振幅差関数であり、 $W$ は窓のサイズ、 $f$ は観察する

信号の波形データである。

### 4.3.2 自己相関関数

自己相関関数 (ACF : Auto Correlation Function) は主に 2 つあり、それらを Type I と Type II とする。Type I は以下のように定義される [59]。

$$r(\tau) = \sum_{j=0}^{W-1} f(j)f(j+\tau) \quad (4.10)$$

ここで、 $r(\tau)$  は遅れ  $\tau$  における自己相関関数である。Type II は以下のように定義される。

$$r'(\tau) = \sum_{j=0}^{W-1-\tau} f(j)f(j+\tau) \quad (4.11)$$

この定義では、 $\tau$  の増加とともに積分区間が減少することを示している。

### 4.3.3 二乗差関数

二乗差関数 (SDF : Squared Difference Function) も自己相関関数と同様、2 種類の定義を行う。Type I は以下のように定義される [59]。

$$d(\tau) = \sum_{j=0}^{W-1} (f(j) - f(j+\tau))^2 \quad (4.12)$$

同様に SDF の Type II は以下のように定義される。

$$d'(\tau) = \sum_{j=0}^{W-1-\tau} (f(j) - f(j+\tau))^2 \quad (4.13)$$

ACF の Type II と同様に、 $\tau$ が増加するにつれて積分範囲が減少する。どちらの SDF においても  $\tau = 0$ の時に最小値をとるのに対し ACF では  $\tau = 0$ において最大値を取る。(4.13)式を展開すると SDF の計算の中に ACF の計算が含まれていることがわかる。

$$d'(\tau) = \sum_{j=0}^{W-1-\tau} \{f(j)^2 + f(j+\tau)^2 - 2f(j)f(j+\tau)\} \quad (4.14)$$

ここで以下の(4.15)式を定義し、(4.16)式を得る。

$$m'(\tau) = \sum_{j=0}^{W-1-\tau} \{f(j)^2 + f(j+\tau)^2\} \quad (4.15)$$

$$d'(\tau) = m'(\tau) - 2r'(\tau) \quad (4.16)$$

#### 4.3.4 正規化二乗差関数

正規化二乗差関数(NSDF)は(4.17)式のように定義される[59]。

$$n'(\tau) = 1 - \frac{m'(\tau) - 2r'(\tau)}{m'(\tau)} = \frac{2r'(\tau)}{m'(\tau)} \quad (4.17)$$

#### 4.3.5 NSDF を用いた周期性の抽出

次に、NSDF を利用したリード領域の抽出方法について述べる。多くのリードは前述のとおり、同一形状のものが水平あるいは垂直方向に並んでいるものが多い。そのため画像中の垂直・水平方向の輝度変化を信号の波形と捉え、画像の水平ライン毎および垂直ライン毎に AMDF を取得した。以下、単純にするため水平方向のみについて記す。

AMDF は波形をずらした際の差の絶対値を取るため、常にその値は正となる。これはNSDF を得る際に負の相関が負値とならないため都合が悪い。そこでAMDF

で取得された値を微分し、それに対して NSDF を取ることで正の相関が高い箇所を判定した。

図 4-20 は図 4-19 の画像中における line a と line b における AMDF を得たものである。リード群のある line a における AMDF は周期性が観察できるが、line b では顕著な周期性が無いことが観察できる。

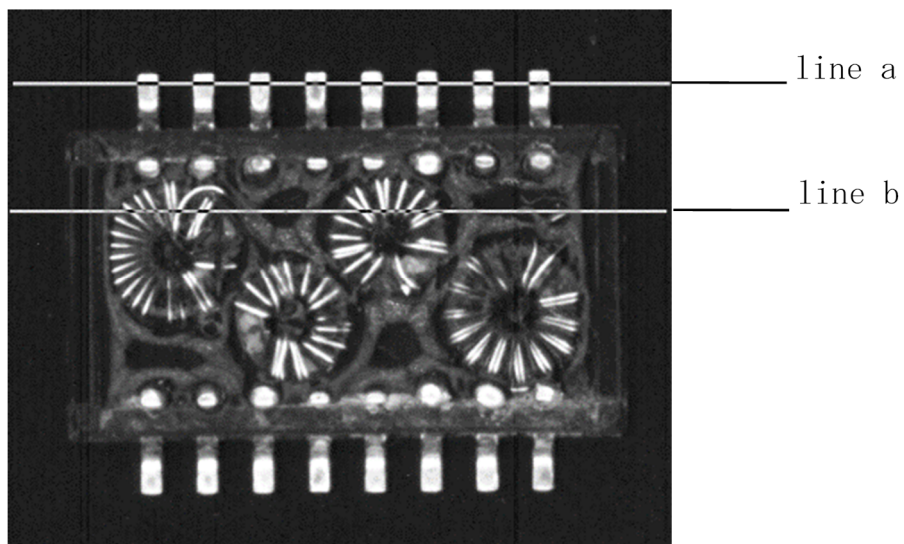


図 4-19 画像例

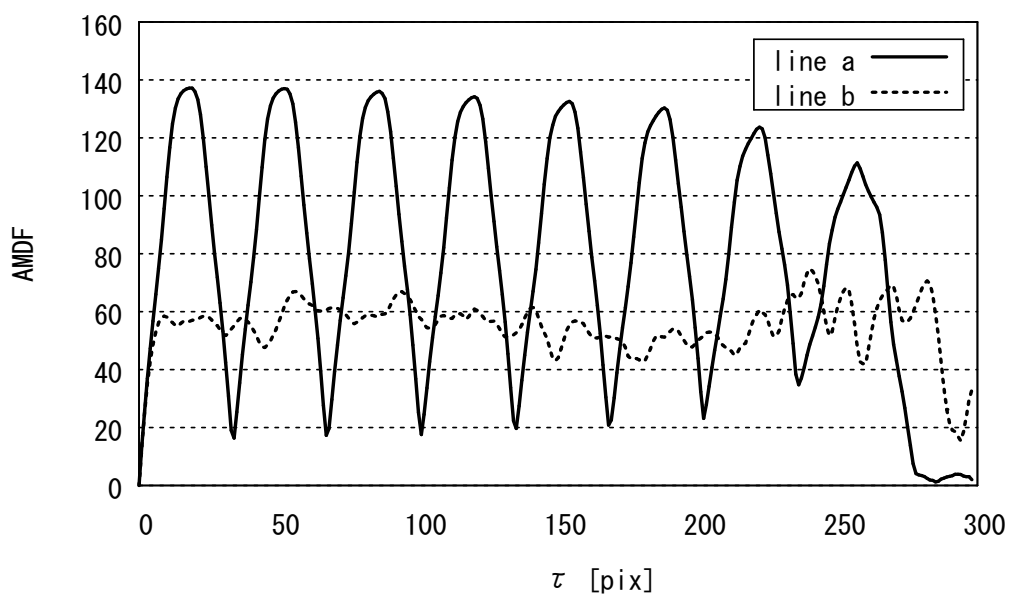


図 4-20 line a, b における AMDF



これを微分したものが図 4-21 であり、この時点でも line a の周期性が確認できている。図 4-21 に対して NSDF を得たものが図 4-22 である。line a においてはリードの間隔毎に安定して高い相関値がピークとして得られていることがわかる。

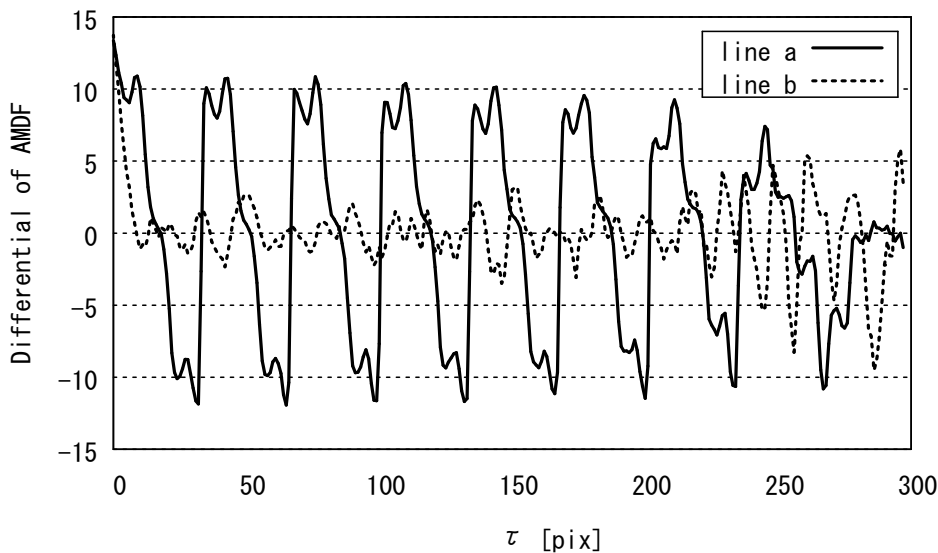


図 4-21 AMDF の微分

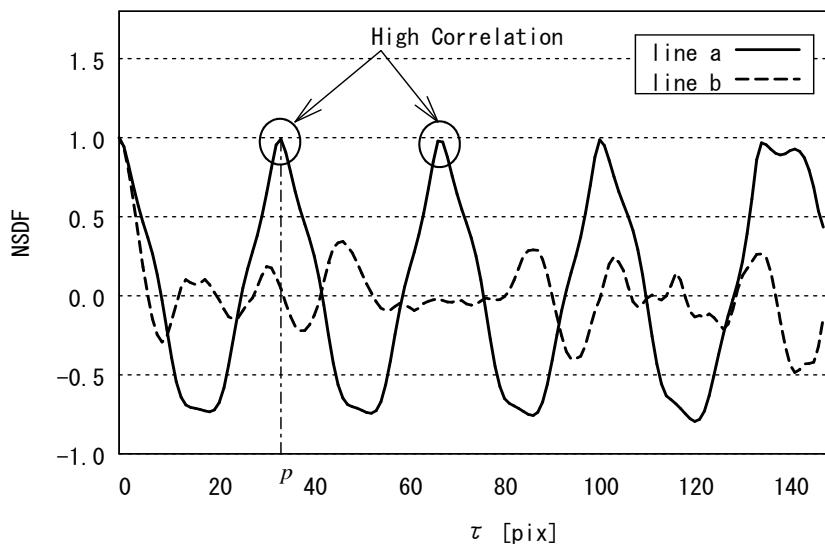


図 4-22 AMDF に NSDF を適用

NSDF は  $\tau = 0$  の場合に最大値 1.0 になるように正規化されているため、 $\tau > 0$  でかつ、極大 NSDF が 0.9 以上の値を持つラインを周期性があるラインだと判

断し、Y 軸方向に連続して周期性がある領域の抽出を行った (図 4-23)。この結果、図 4-24 に示すように、(A)～(D)の領域が周期性のある領域として検出された。検出された 0.9 以上の極大 NSDF が現れた箇所のうち、 $\tau$ が最小のものを検出ピッチ $p$  (図 4-22)とした。

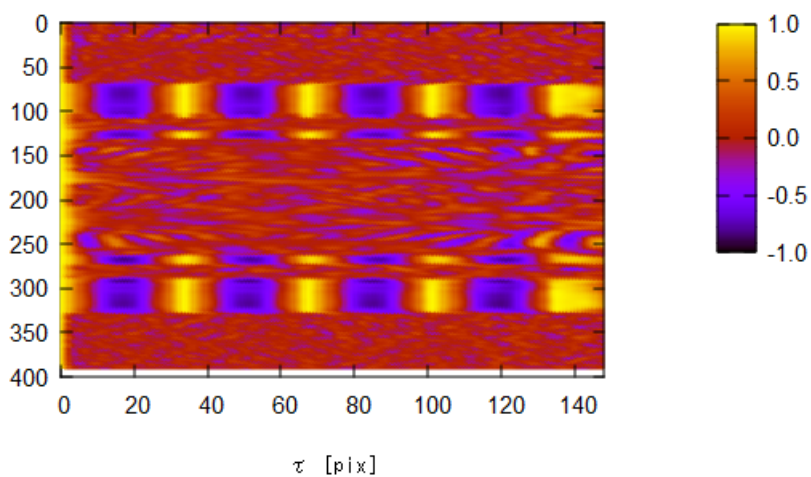


図 4-23 図 4-19 画像例における水平方向の NSDF (画像全体)

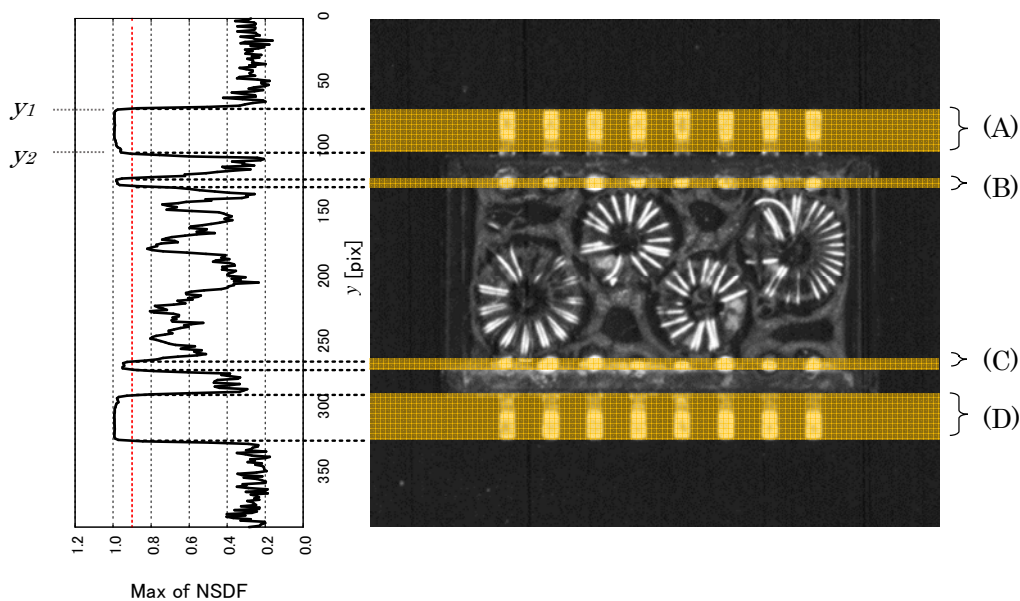


図 4-24 周期性のあるラインの抽出

### 4.3.6 リード群の開始・終了座標の検出

リード群の開始、終了座標を得る手法を説明する。4.3.5 項において連続して周期性があると判断した領域に対して X 軸方向への輝度の投影を行う。平均投影輝度  $T(x)$  は (4.18) 式で表す。

$$T(x) = \sum_{y=y_1}^{y_2} I(x, y) / |y_2 - y_1| \quad (4.18)$$

$I(x, y)$  は座標  $(x, y)$  における画像の輝度であり、 $y_1, y_2$  はそれぞれ連続して周期性があると判断した領域の開始および終了 Y 座標である。この領域はピッチが検出できている領域なので、検出されたピッチ間隔で投影した波形データの標準偏差を取ると、リード群のある区間では標準偏差が高く、そうでない領域では標準偏差が低く得られる。そこで  $T(x)$  に対し検出ピッチ  $p$  による区間標準偏差  $S(x)$  を求めた。

$$S(x) = \sqrt{\frac{1}{p} \sum_{n=x-\frac{p}{2}}^{x+\frac{p}{2}} (T(n) - \bar{T}(x))^2} \quad (4.19)$$

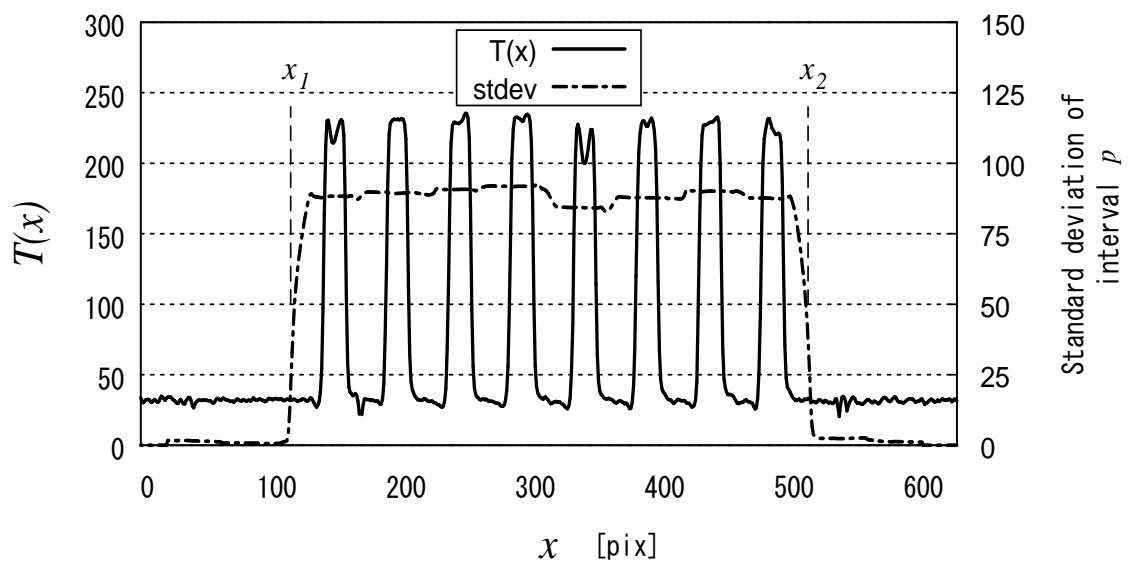


図 4-25 領域(A)における  $T(x)$  及び  $p$  による標準偏差

図 4-25 は図 4-24 における (A) の領域の  $T(x)$  と区間標準偏差  $S(x)$  を得たグラフである。得た区間標準偏差が高い区間だけに領域を限定することで、リード群の開始および終了座標  $x_1, x_2$  が取得できる。リードは部品の上下左右方向へ定義されるため、同様の処理を画像の垂直水平の両方向に対して行い、垂直方向のリード群と水平方向のリード群領域をそれぞれ抽出した。抽出した領域の例を図 4-26 に示す。

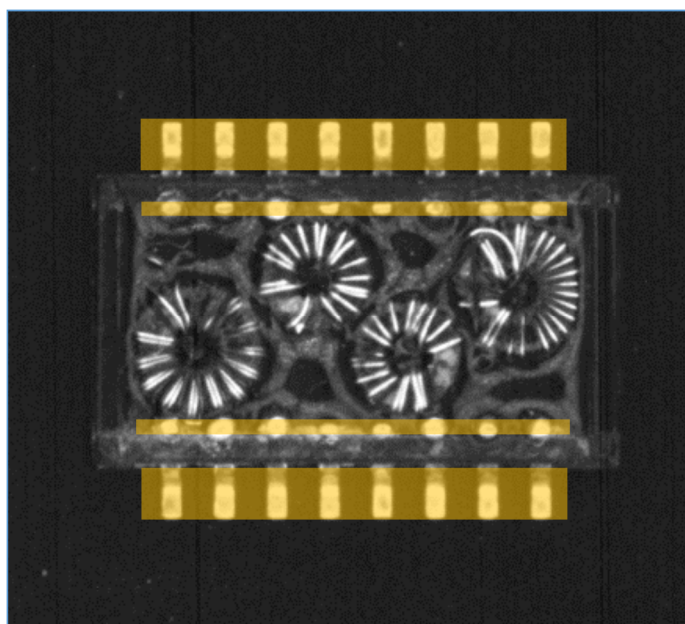


図 4-26 抽出されたリード群領域の例

この抽出したリード群領域内を対象に、Real AdaBoost によるリード先端検出を行うことで False Positive の削減を試みた。最終的にリードの形状データはこの手法でリード先端を検出した後、先行研究[58]において説明されている「エッジ検出によるリード位置およびサイズ計測」の手法を用いて定義した。この手法ではリード先端とリード付け根の位置をエッジ検出して行う。リード先端はこの手法で検出が可能だが、リードの付け根位置に関してはボディ部分と接続されて検出できない場合がある。そこで幅計測を先端から順次行い、幅計測不能になった時点での位置をリード長さとすることで対応した。

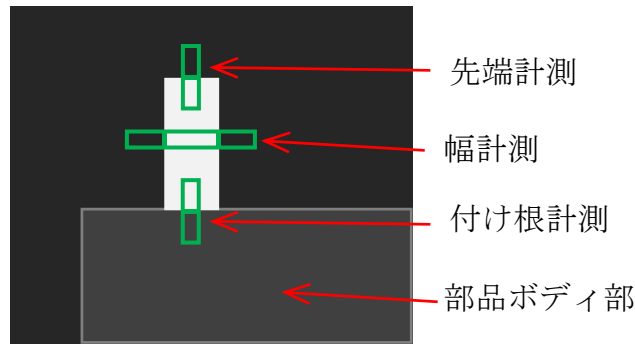


図 4-27 先行研究でのリード計測手法

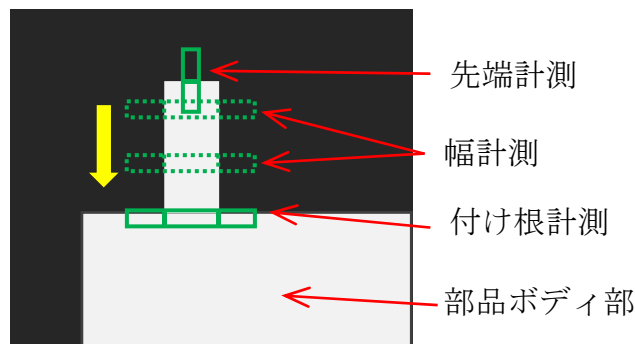


図 4-28 リード計測手法の改良

### 4.3.7 リードの向き決定

部品画像中から周期性の高い領域の抽出ができたが、リードの方向は特定できない。3.4.1 項でも述べたように、リードは通常部品のボディ部分から外側方向へ取り付けられている。水平方向に周期性が得られた領域内では上下方向の、垂直方向に周期性が得られた領域内では左右方向のリードが存在する可能性が高いため、リード検出は4方向行う必要はなく、水平方向の周期性が認められた領域では上下方向のリードを検出すれば良い。しかし、4.2.5 項でのリード検出実験でもわかるように、この領域内でリード先端を AdaBoost による手法で検索した場合、一般的な IC 部品のようにリード1本1本が独立しているリード形状の場合には、リードの先端あるいはリードの付け根のいずれかで検出される（図 4-29 参照）。

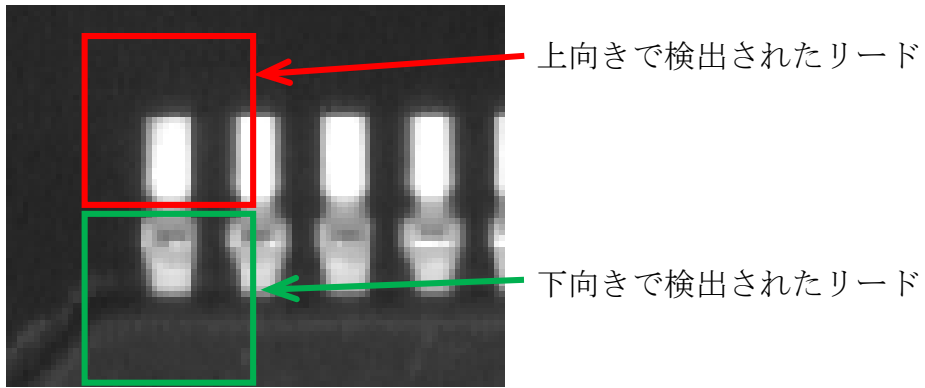


図 4-29 1本のリードで上下両方向に検出されたリード

そこで水平方向に周期性が認められたリード群領域内でリードが検出された場合、検出されたリードの X 座標が同じ場合には、部品中心から外側にある方のリードを残し、内側の検出リードは削除する。同様に垂直方向の周期性が認められた場合には、Y 座標で判断して内側のリードを削除する。

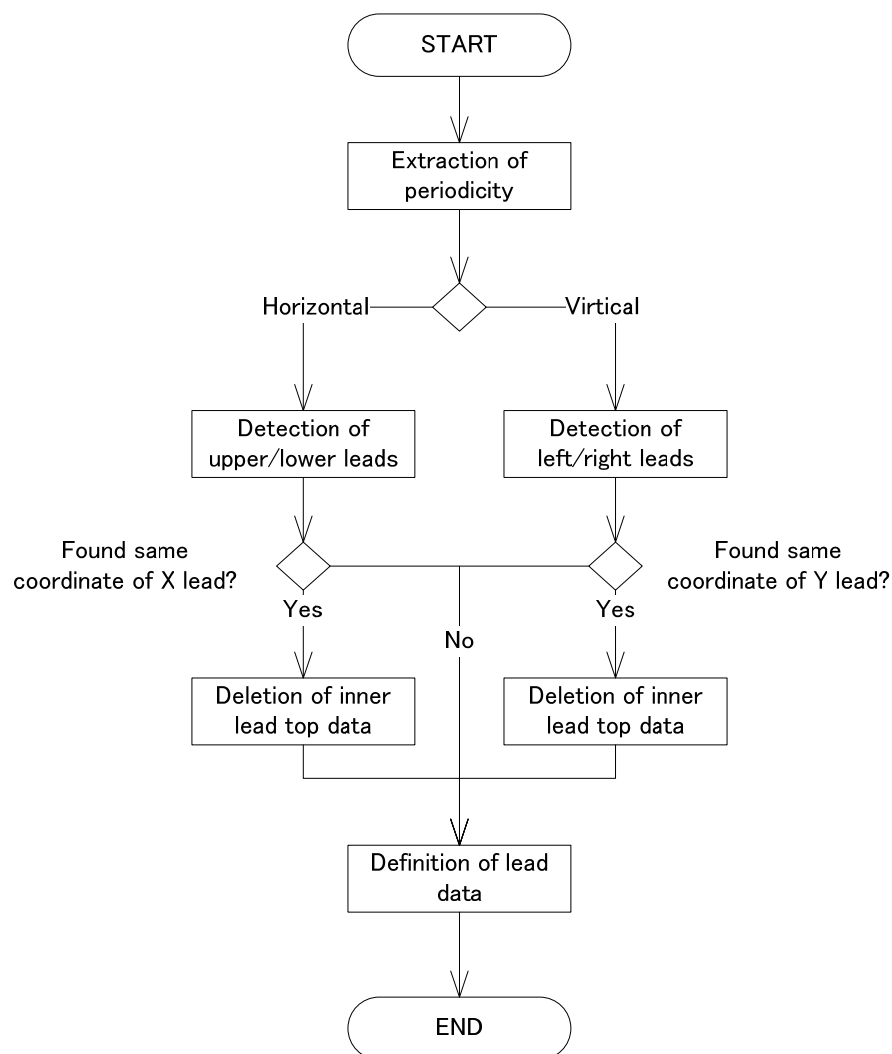


図 4-30 リード群領域からのリード検索フロー

## 4.4 評価実験

### 4.4.1 リード群領域の抽出性能の評価と考察

まず部品画像全体を用いて、NSDF を用いたリード領域抽出の性能を確認する。ここでは比較のため AMDF を用いた場合の実験も行う。性能は提案手法によりリード領域を抽出した際、真のリード本数のうち抽出領域が含んだ本数の割合  $P$  と、部品領域面積における抽出領域面積の割合  $Q$  から計測した。 $P$  と  $Q$  は以

下の式を定義した。使用したサンプルは、一般的な IC 部品 40 サンプル、複雑な形状のリード部品 70 サンプルである。

$$P = \frac{L_{inc}}{L_{true}} \quad (4.20)$$

$$Q = \frac{S_{ext}}{S_{part}} \quad (4.21)$$

ここで $L_{inc}$ は正解リードのうち領域に含まれた本数、 $L_{true}$ は真のリード本数である。また、 $S_{ext}$ は抽出した領域の面積、 $S_{part}$ は部品領域全体の面積である。領域を抽出するための相関値の閾値を変化させ、それぞれの $P$ および $Q$ からリード領域の抽出性能を計測した。結果を図 4-31 に示す。AMDF を用いた検出手法に比べ、提案手法ではより少ない抽出領域で多くの真のリード領域を抽出できていることがわかる。

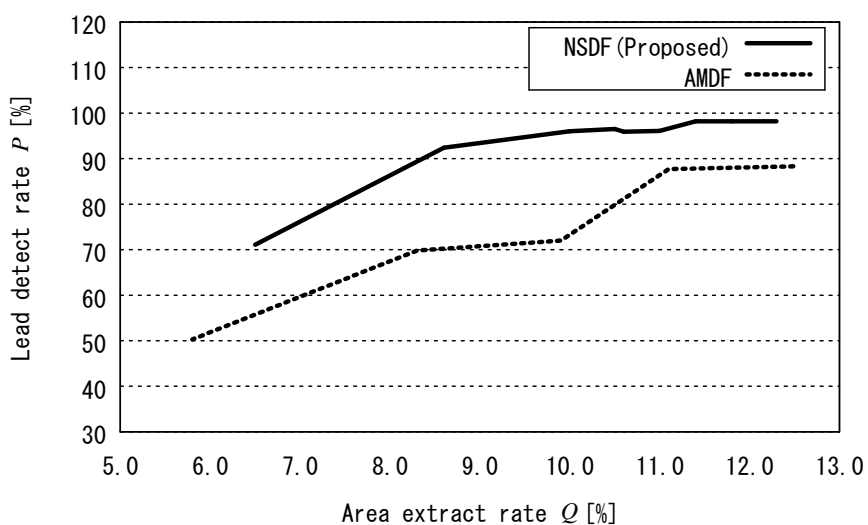


図 4-31 リード群領域の抽出性能

#### 4.4.2 Real AdaBoost と Haar-Like 特徴を用いたリード検出性能の評価と考察

4.2 節で作成した Real AdaBoost + Haar-Like 特徴量を使用したリードの識



別器を用い、部品画像中からリード先端の検出性能を計測した。リードは部品ごとにより様々な幅を持つため、対象となるリード部品画像に対してウィンドウサイズを変えながらラスタスキャンを行い、リードの検出率と誤検出率を評価した。検出率 $D$ として(4.22)式を、誤検出率 $F$ として(4.23)式を定義した。

$$D = \frac{L_{pos}}{L_{true}} \quad (4.22)$$

$$F = \frac{L_{neg}}{L_{total}} \quad (4.23)$$

ここで $L_{pos}$ は正解を検出したリード本数である。また $L_{neg}$ は誤検出したリード本数であり、 $L_{total}$ は総検出リード本数である。一般的にリードは部品中心から上下左右の4方向に向かって配置されている。そのため、学習した右向きのリード識別器を $90^\circ$ 毎に回転させて4方向のリードをすべて検出できるようにした。図4-32および図4-33は一般的なIC部品と複雑な形状の部品とでリードの検出率と誤検出率を比較したものである。一般的なIC部品に対して検出率は高く、誤検出率は低い。しかし複雑な形状の部品の場合、検出率は一般的なIC部品とほぼ変わらないものの、誤検出率が高い。この理由は、一般的なIC部品はボディ部分が黒く、誤検出対象が少ないことがあげられる。しかし複雑な形状の部品では、図4-17のようにコイルの一部やフレームの一部など、局所的に見ればリードと見えてしまう部分を検出しているためである。

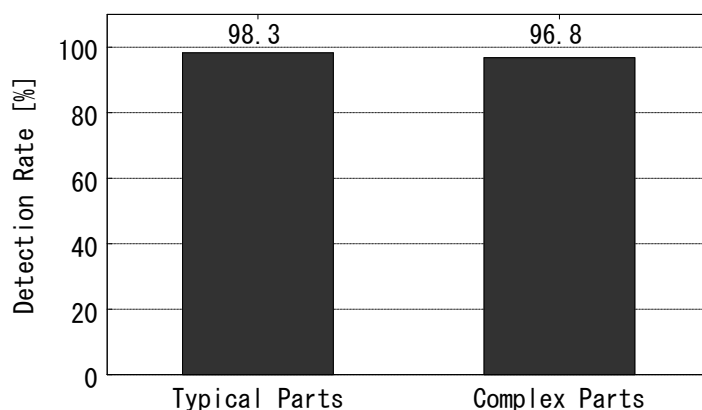


図 4-32 一般的な IC 部品と複雑なリード部品でのリード検出率

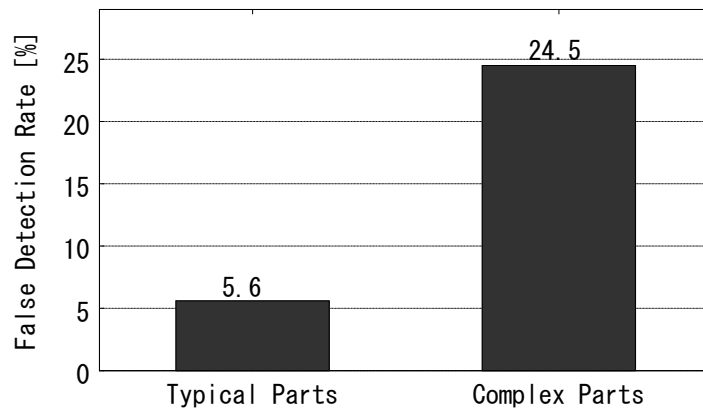


図 4-33 一般的な IC 部品と複雑なリード部品でのリード誤検出率

#### 4.4.3 提案手法によるリード検出実験と考察

NSDF によるリード領域検出手法と AdaBoost によるリード識別手法を組み合わせた場合の性能を評価する。なお、図および表中の手法 1 は 4.4.2 項における複雑な形状の部品の結果を表し、手法 2 は周波数性解析によるリード群検出と AdaBoost によるリード識別を組み合わせたものである。実験に使用した画像サンプルは、複雑な形状のリード部品の 70 サンプルであり、4.4.2 項で用いたものと同様である。図 4-34 および図 4-35 に実験結果を示す。また表 4-1 および表 4-2 には定義すべきリード本数、検出本数、誤検出本数を手法別に示した。

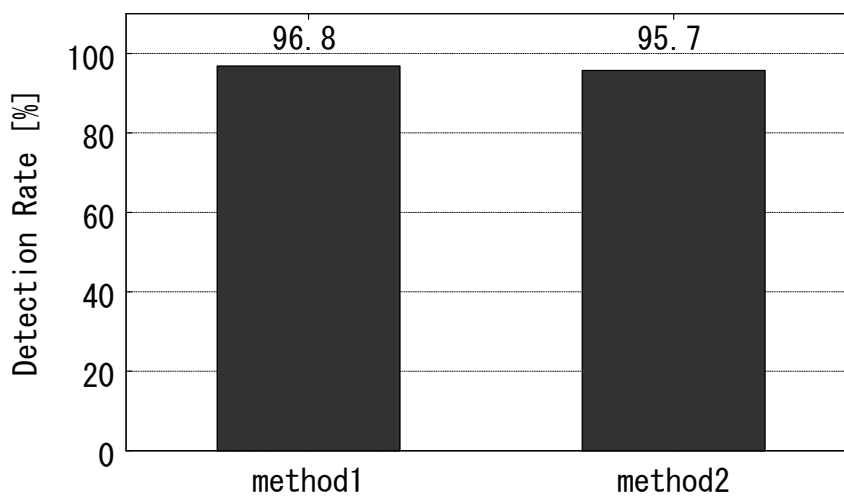


図 4-34 複雑なリード部品でのリード検出率の比較

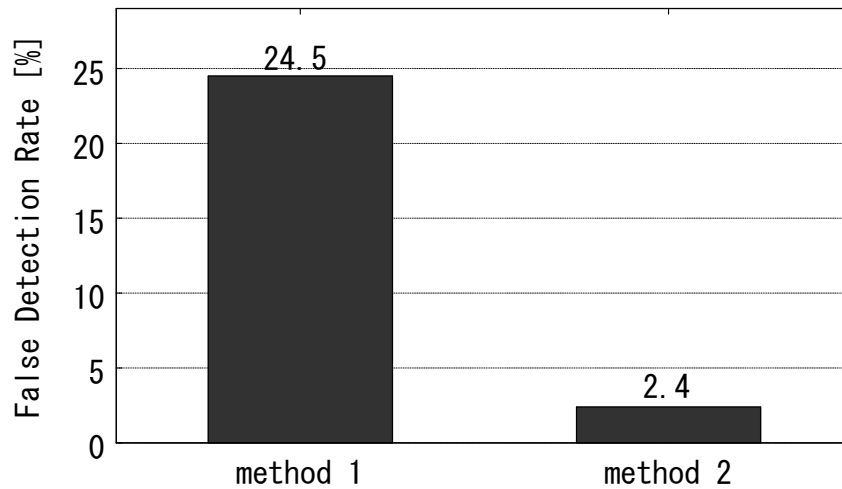


図 4-35 複雑なリード部品でのリード誤検出率の比較

表 4-1 真陽性および偽陽性と判定されたリード本数（手法 1）

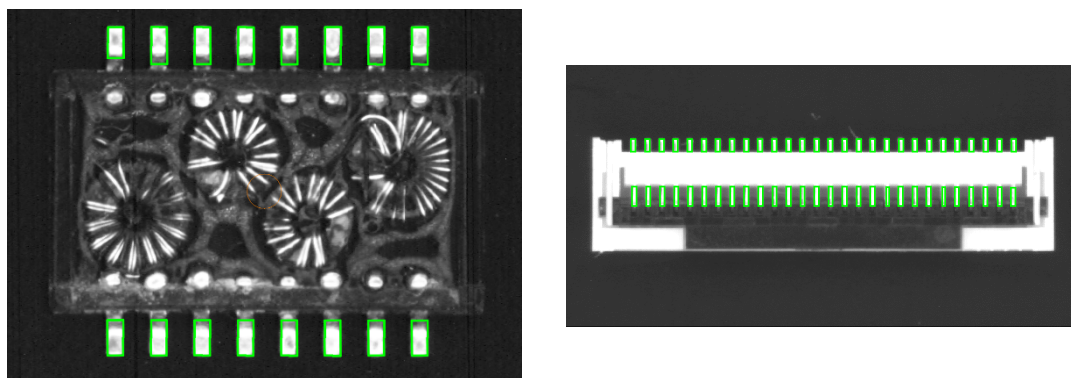
	Ideal leads count	True positive count	False positive count
Typical Parts	1916	1884	143
Complex Parts	2271	2199	714

表 4-2 真陽性および偽陽性と判定されたリード本数（手法 2）

	Ideal leads count	True positive count	False positive count
Typical Parts	1916	1880	30
Complex Parts	2271	2173	54

リードの検出率は、手法 1 と手法 2 でほぼ同等であるものの、手法 2 では誤検出率が 2.4[%]と手法 1 (24.5[%])に比べて大きく低減されており、リード群領域の抽出法が有効であることが確認できた。本手法においても誤検出したリードが確認できるが、この理由は、周期性解析による領域抽出の結果、非リードだが輝度変化が高く区間標準偏差が増大したことでリード領域として抽出され、さらに Real AdaBoost によるリード検出手法でもリードと識別されたも

のが含まれているためである。図 4-36 に最終的に定義されたリードの形状データの例を示す。図中の緑色の矩形が定義されたリードデータを示している。

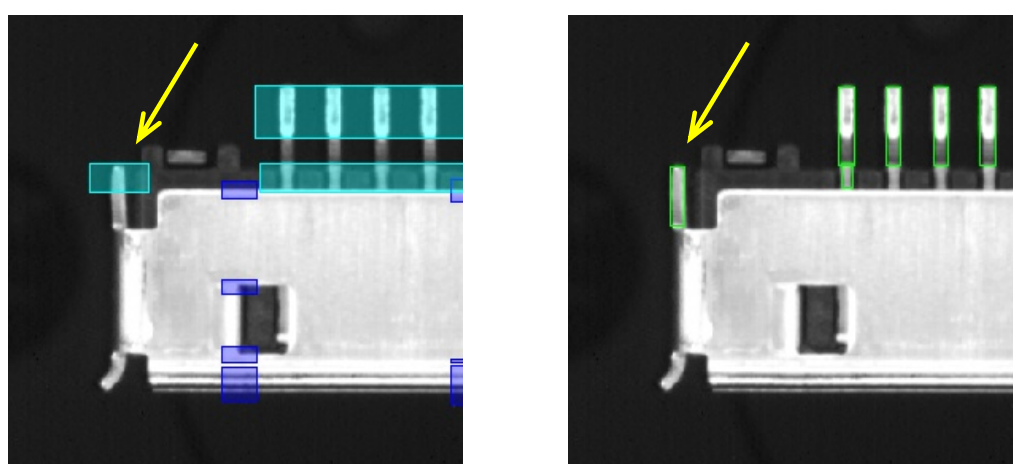


(a) サンプル部品 1

(b) サンプル部品 2

図 4-36 検出されたリードデータの定義例

図 4-37 に本手法での誤検出の例を示す。図 4-37 (a) は誤検出した領域の例、図 4-37 (b) は誤検出した領域からリード検出し、リード定義が出来たために定義ミスとなったリードデータである。(a) の図中の水色の矩形領域は水平方向の周期性検出により検出されたリード群領域、青色の矩形領域は垂直方向で検出されたリード群領域を示しており、(b) の図中の緑色の矩形は、定義されたリードデータである。



(a) 誤検出された領域

(b) 定義ミスしたリードデータ

図 4-37 誤検出例

このようにボディのフレーム部分であるにも関わらず、リードに見えてしまう部分ではリードの誤検出が発生してしまう。同一部品画像内で他に検出されたリード部分との類似度や対称性などの構造的な特徴から削減が可能か検討する余地がある。

一方で検出率が 96.8[%]から 95.7[%]と僅かに減少した原因は、リードがあるにも関わらず、リード本数が少ないなどの理由により NSDF のピーク値が低い部品存在し、これらが周期性解析による領域特定を行うことで除外されたためである。未検出リード群領域の例を図 4-38 に、そのため未定義のリードの例を図 4-39 に示す。また、図 4-38 の画像の水平方向への NSDF を図 4-40 に示す。この例では、図中に示したように NSDF のピーク値が 0.892 であり、閾値である 0.9 以下なのでリード検出領域が未検出となる。

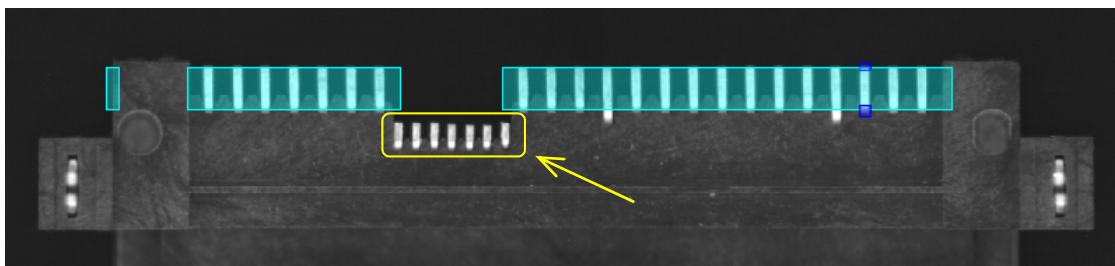


図 4-38 未検出のリード群領域

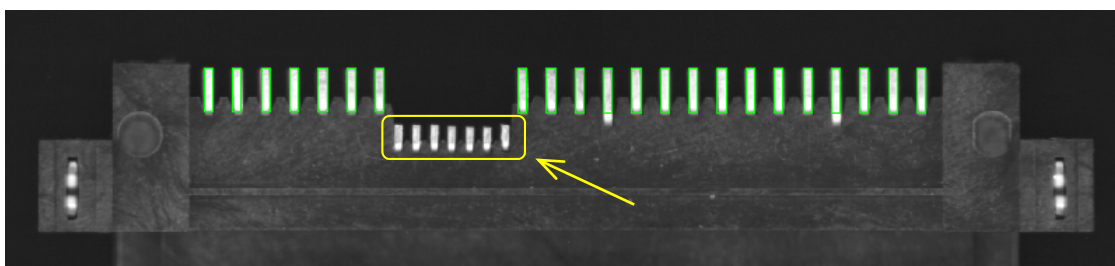


図 4-39 未定義のリードの形状データ

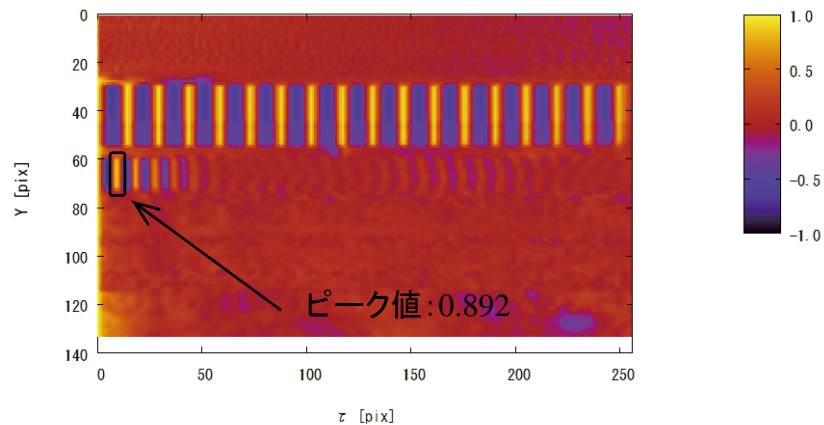
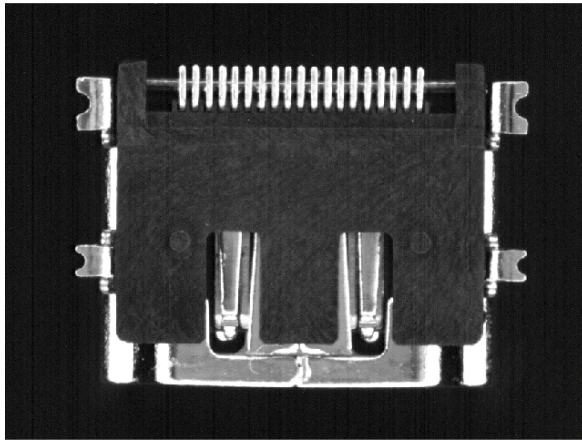


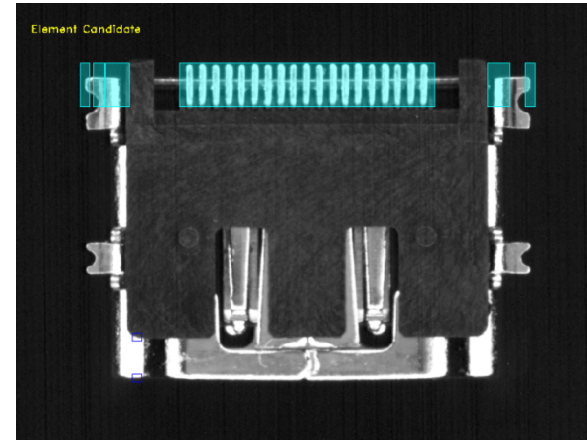
図 4-40 図 4-38 画像の水平方向の NSDF

提案手法を用いて形状データの初期値を作成した場合、未検出リードデータの作成および誤検出リードデータの削除をマニュアル操作で行う。作成作業にリード 1 本あたり約 2 秒、削除作業に約 1 秒かかるとした場合、今回対象とした複雑な形状のリード部品 70 サンプル(リード 2271 本)では形状データの作成に約 76 分程度要する。一方提案手法の場合、未検出 98 本および誤検出 54 本をマニュアル操作で作成・削除することになる。自動生成時間にかかる時間、1 部品あたり約 1 秒(Core i7@2.1GHz)を加えても約 5 分で終了する。すべてマニュアル操作で作成した場合に比べ 93[%]の時間短縮が達成できる。

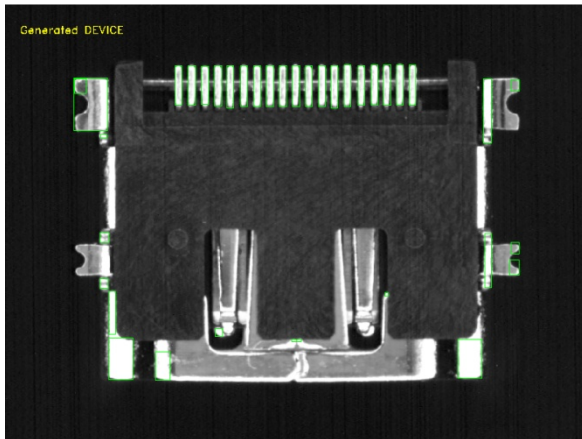
最後に、図 4-41～図 4-50 にリード群領域を特定せずにリード検出した場合と、領域特定後にリード検出した場合の検出例を示す。それぞれの図の(a)は対象部品画像、(b)は抽出したリード群領域、(c)は(a)から直接 Real AdaBoost によるリード検出した場合の検出結果、(d)は(c)のリード群領域を対象にリード検出した結果を表している。(b)における水色の矩形で示された部分は水平方向の周期性検出により検出されたリード群領域を、青色の矩形は垂直方向で検出されたリード群領域を示している。(c), (d)で緑色で示された矩形は検出されたリードデータを示している。



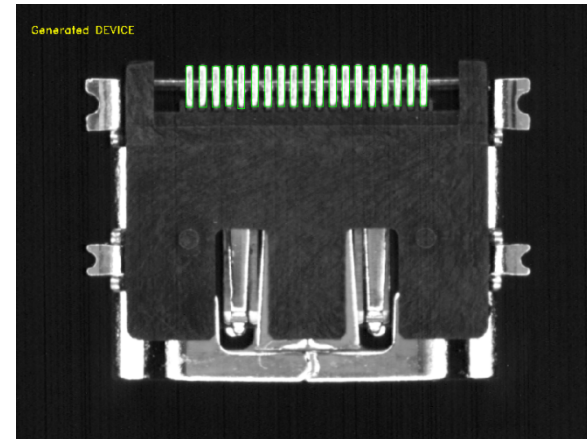
(a) 原画像



(b) リード群領域

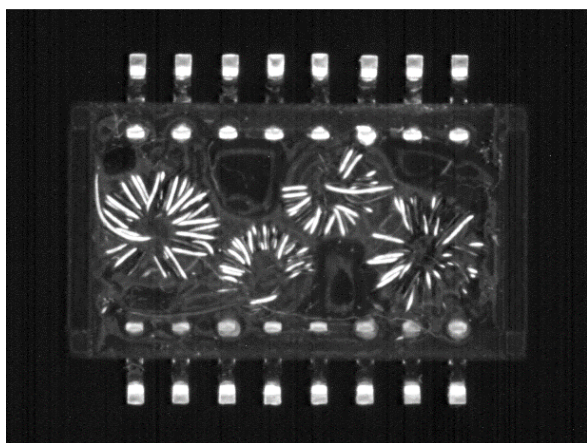


(c) 原画像からのリード抽出

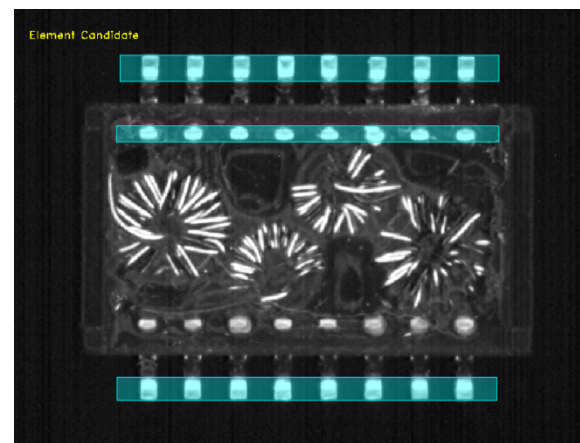


(d) リード群領域からのリード抽出

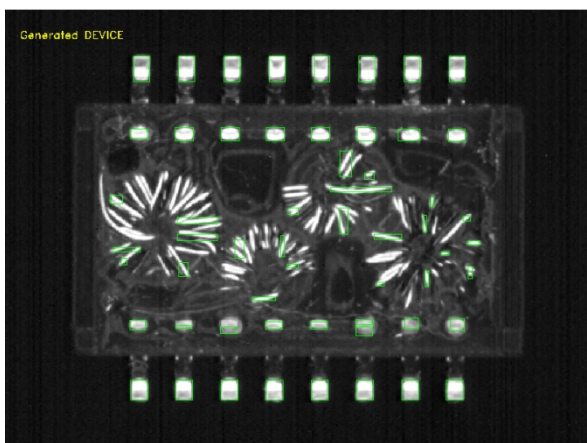
図 4-41 コネクタ部品のリード検出例 1



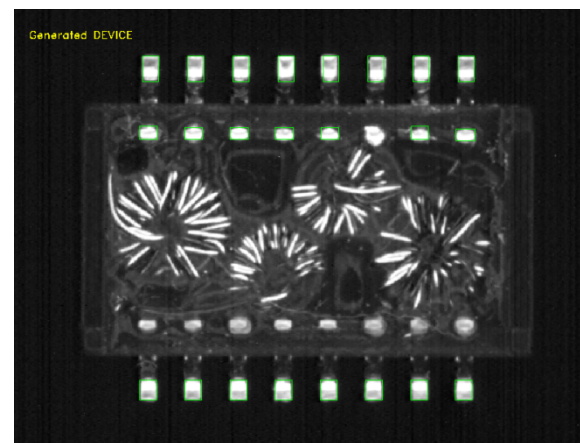
(a) 原画像



(b) リード群領域



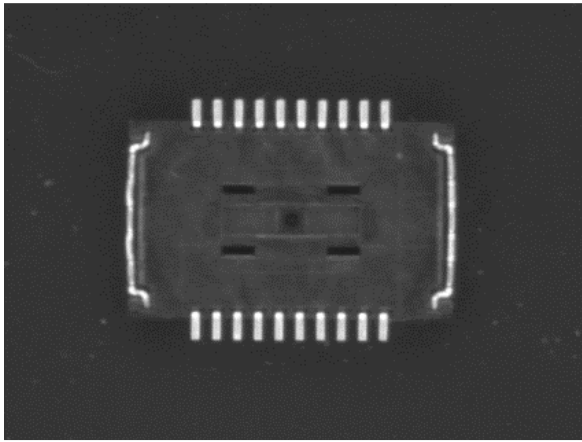
(c) 原画像からのリード抽出



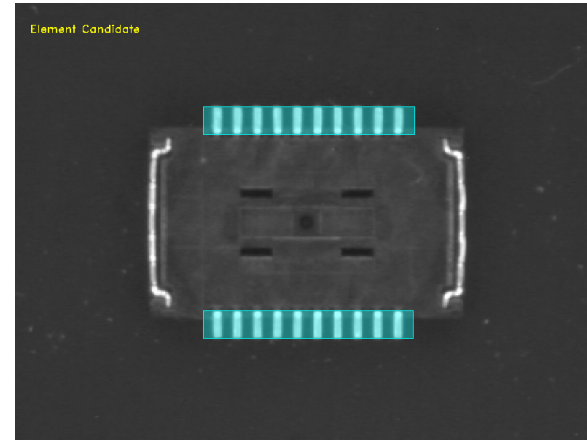
(d) リード群領域からのリード抽出

図 4-42 コネクタ部品のリード検出例 2

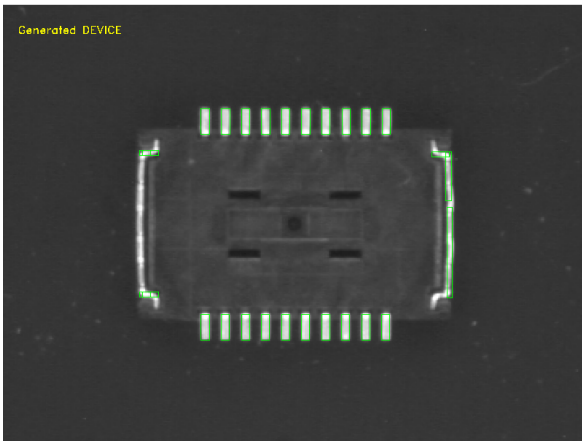




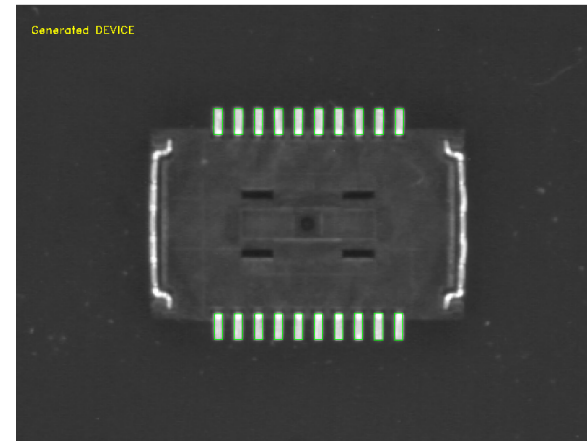
(a) 原画像



(b) リード群領域

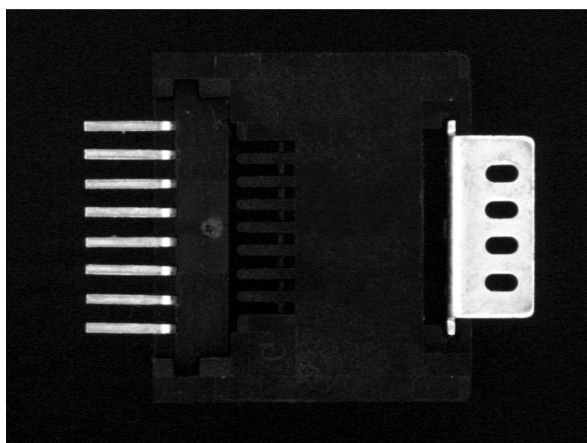


(c) 原画像からのリード抽出

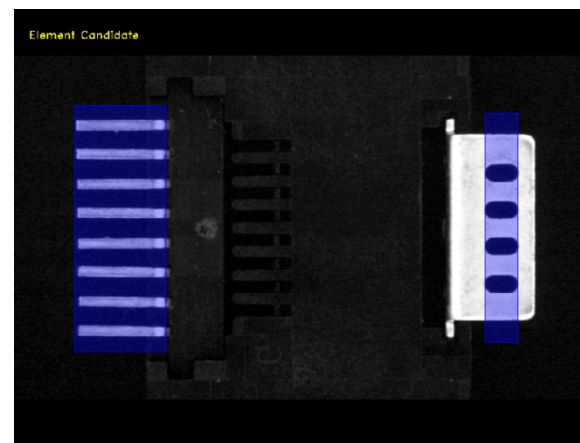


(d) リード群領域からのリード抽出

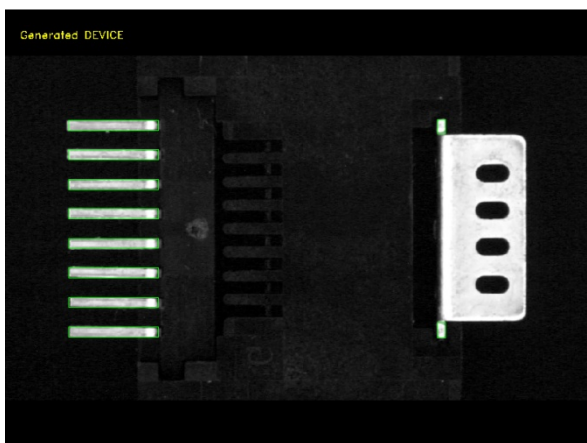
図 4-43 コネクタ部品のリード検出例 3



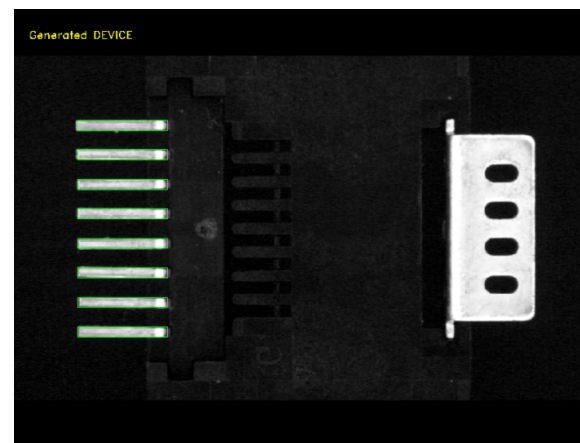
(a) 原画像



(b) リード群領域

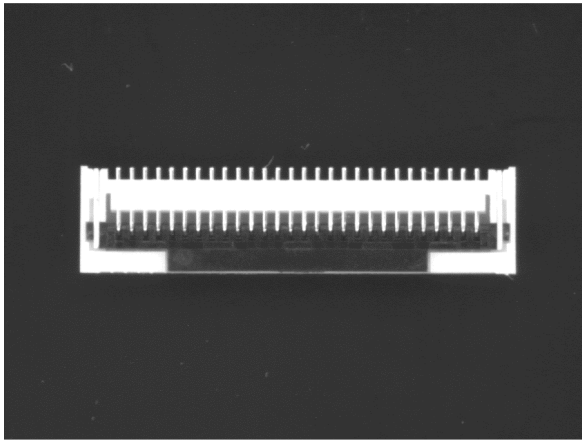


(c) 原画像からのリード抽出

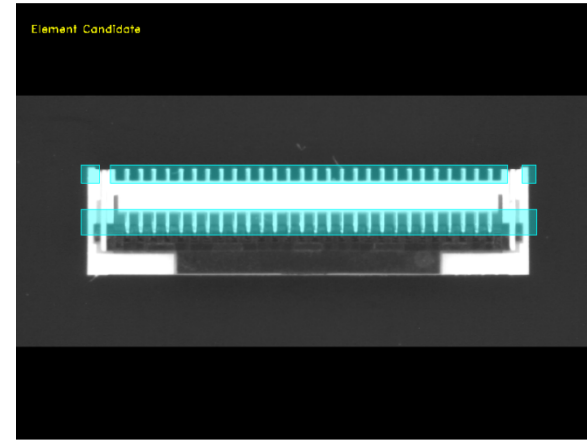


(d) リード群領域からのリード抽出

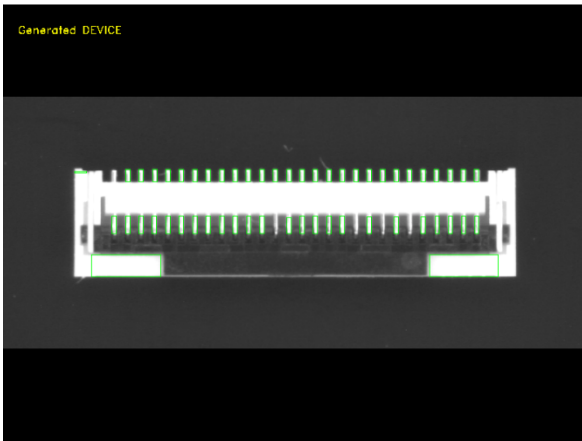
図 4-44 コネクタ部品のリード検出例 4



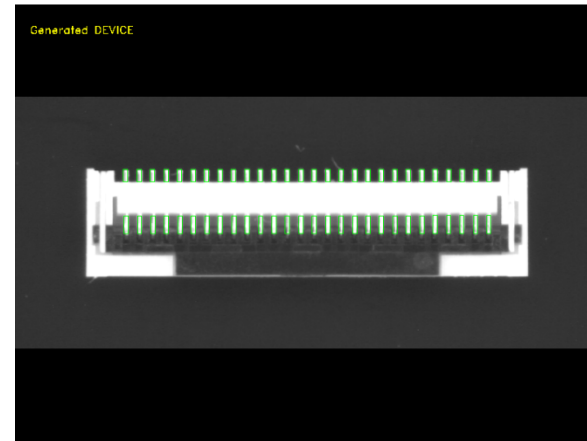
(a) 原画像



(b) リード群領域

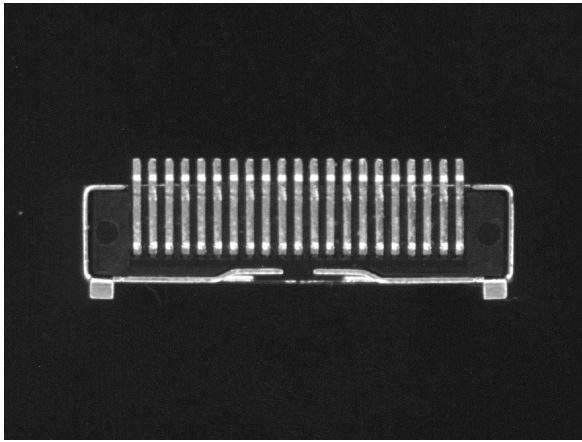


(c) 原画像からのリード抽出

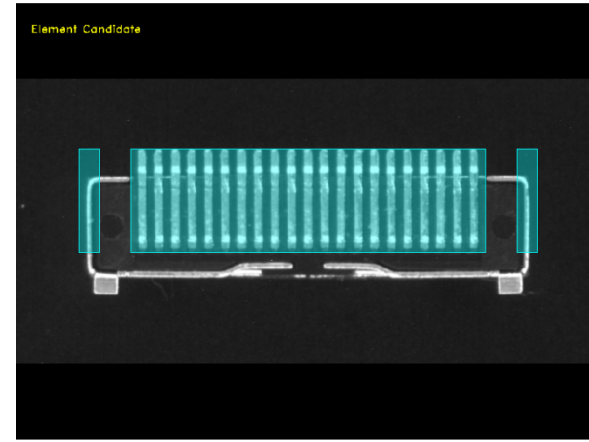


(d) リード群領域からのリード抽出

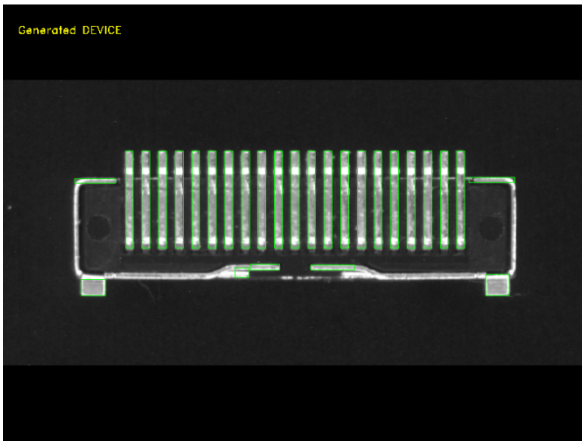
図 4-45 コネクタ部品のリード検出例 5



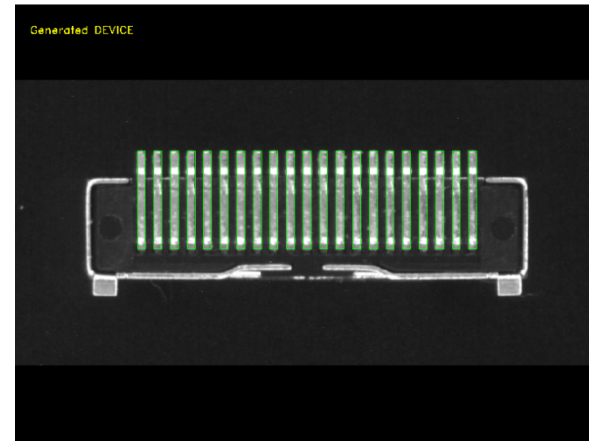
(a) 原画像



(b) リード群領域

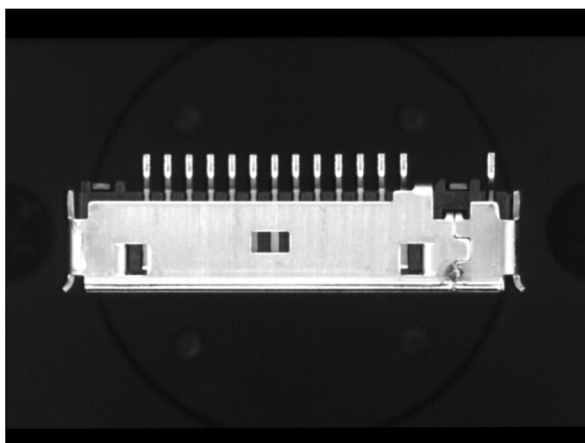


(c) 原画像からのリード抽出



(d) リード群領域からのリード抽出

図 4-46 コネクタ部品のリード検出例 6



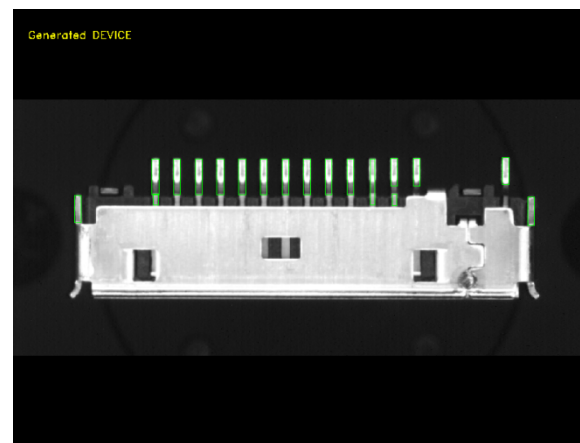
(a) 原画像



(b) リード群領域

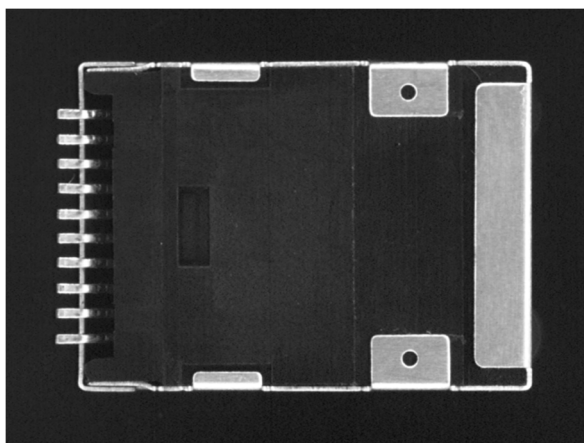


(c) 原画像からのリード抽出

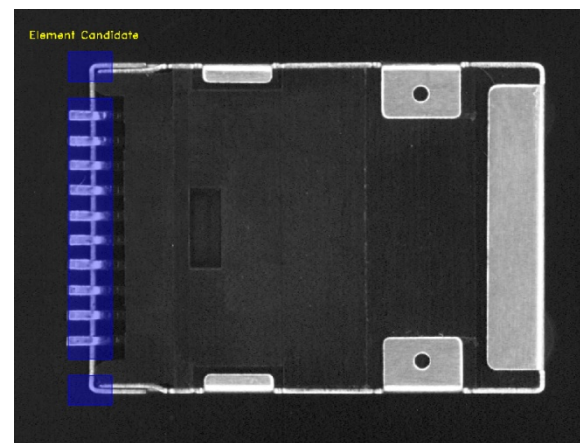


(d) リード群領域からのリード抽出

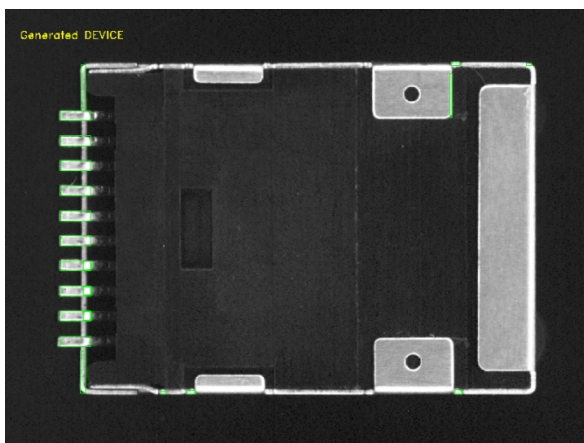
図 4-47 コネクタ部品のリード検出例 7



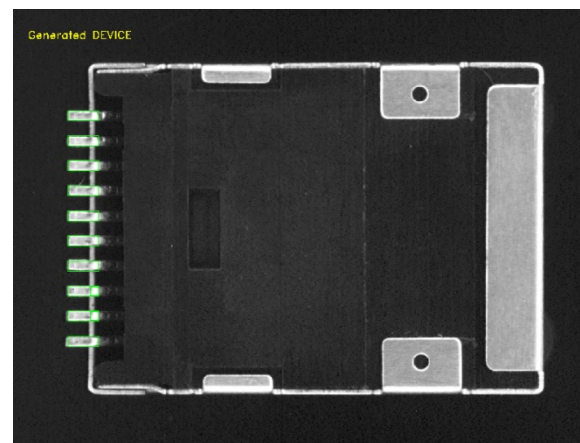
(a) 原画像



(b) リード群領域

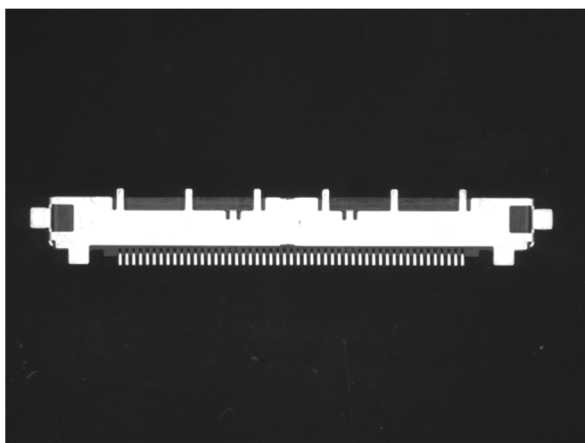


(c) 原画像からのリード抽出

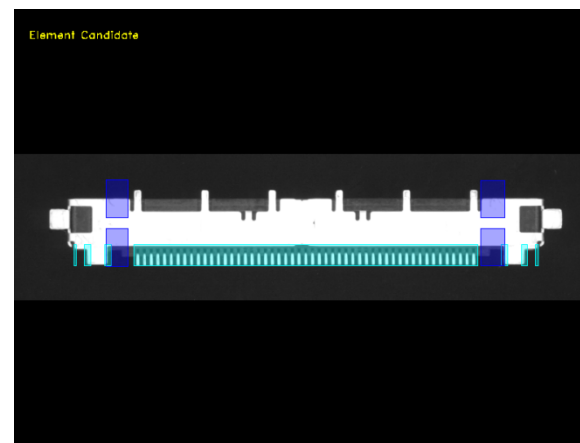


(d) リード群領域からのリード抽出

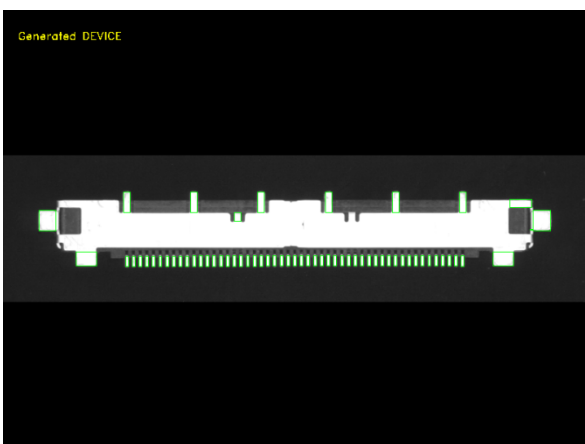
図 4-48 コネクタ部品のリード検出例 8



(a) 原画像



(b) リード群領域

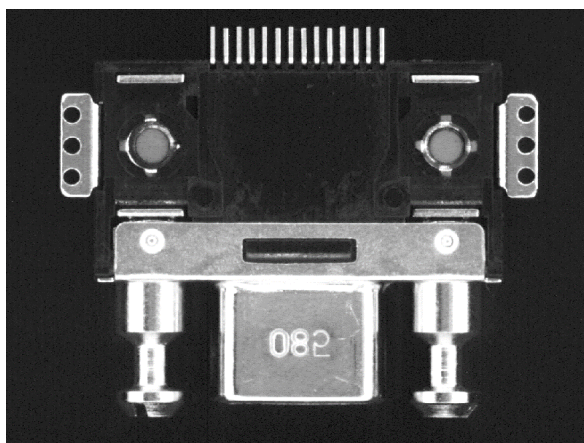


(c) 原画像からのリード抽出

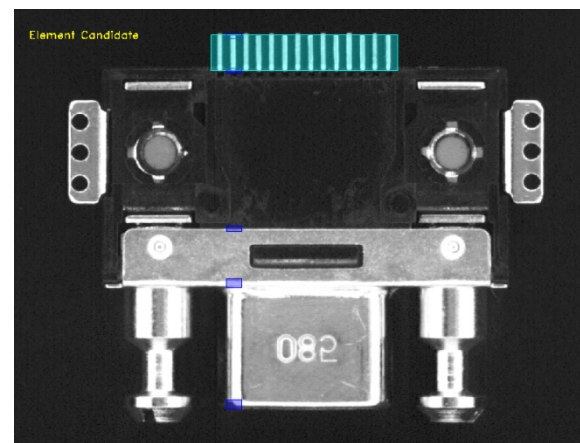


(d) リード群領域からのリード抽出

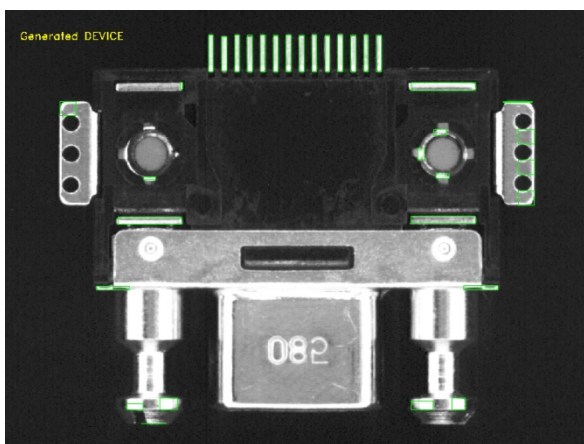
図 4-49 コネクタ部品のリード検出例9



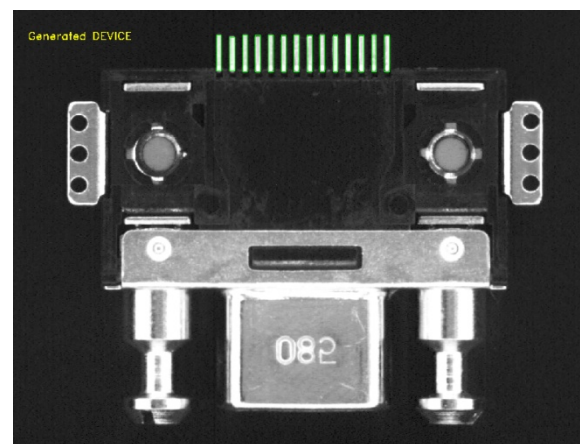
(a) 原画像



(b) リード群領域



(c) 原画像からのリード抽出



(d) リード群領域からのリード抽出

図 4-50 コネクタ部品のリード検出例 1 0



## 4.5 まとめ

本章では、第3章でコネクタ部品と分類された部品画像に対し、リードそのものを検出する手法として AdaBoost を用いた検出方法と誤検出の削減方法を提案した。

はじめに、Discrete AdaBoost を用いた手法により、Haar-Like 特徴と HOG 特徴を使用したリード先端検出を行い、Haar-Like 特徴によるリード先端の検出が HOG 特徴を使用した場合よりも有効であることを示した。

次に検出器の比較として Discrete AdaBoost と Real AdaBoost を用いてリードの検出実験を行い、リード先端の検出性能を比較した。僅かではあるが Real AdaBoost + Haar-Like 特徴を用いた場合が Discrete AdaBoost の場合よりも検出性能が高く、Real AdaBoost を採用した。しかし、いずれの手法を用いた場合でも部品中の局所的にリードと見える部分に対しての誤検出が多く、巨視的な観点からリードの誤検出を削減する手法としてリードの周期性を利用したリード群領域の検出法を提案した。

リード群領域の検出法は、音声関係のピッチ検出法として利用されている AMDF および NSDF 利用した方法を検討した結果、NSDF を使用した方法でより効果的にリード群領域を特定することが確認できた。最終的にリード群領域を検出したうえで、Real AdaBoost + Haar-Like 特徴を使用したリード検出を行うことで、領域検出をしない場合に比べ誤検出が 24.5[%]から 2.4[%]へと削減できることを確認した。

提案手法を用いた手法では、まだ 2.4[%]の誤検出と 4.3[%]の未検出リードが存在する。誤検出に関しては同じ部品画像内から検出された他のリードとの類似度や配置などから削減が可能か検討する余地がある。また本手法ではリードの配置に周期性がないコネクタ部品には有効ではない。未検出の多くは、周期性に並ばないリードが検出されていないためである。多くのコネクタ部品はリードが周期的に並んでいるものの、周期的にリードが並ばない部品に関する改良が必要となる。例えば、部品全体からリードを検出し、その上で周期性があるものや他の検出リードの類似性などにより全検出リードのスコアを算出し、その上でリードと非リードを分類する手法などが考えられる。

本章では部品画像だけを用い、リードデータを自動生成する手法を提案し

た。通常、プリント基板を生産する場合には対象部品の装着座標は決定されているものの、試作時には対象部品の画像処理アルゴリズムに応じた座標の微調整が必要になる。この作業は試作中に作業員が目視で装着座標の確認を取りながら行っているが、プリント基板上の配線パターン等を撮像し、部品画像とのマッチングを行うことで、さらなるリードデータ生成の性能向上や装着座標の微調整までも自動化できる可能性がある。

## 第5章 むすび

本論文では、電子部品の表面実装技術における作業の自動化を目的とし、画像処理技術の応用による部品形状の認識と部品形状データの自動生成システムの改良に関する研究を行った。

本論文の第2章では表面実装技術について述べた。表面実装技術において、マウンタを用いてプリント基板を生産するためには、実装データが必要になる。実装データには、部品のプリント基板への装着座標を表す「部品実装座標データ」、部品の位置決めに使用するための「部品形状データ」、部品の位置決めアルゴリズムを示す「画像処理アルゴリズム」など様々なデータの入力が必要となる。近年、市場の変化や顧客のニーズの多様化に柔軟に対応する生産形態として少量多品種生産があるが、少量多品種生産では実装データの切り替えが多く発生し、実生産時間における段取りにかかる時間が相対的に増加する。この問題に対応するため、実装データ生成の自動化が必要となることを述べた。

第3章では、はじめに部品種類自動分類システムの改良について検討した。従来手法における問題点を整理し、特に従来手法では分類性能の低いコネクタ部品に注目し、コネクタ部品の共通の特徴であるリード群の存在確率の高い領域を特定した。リード群の持つ周期性の特徴を得るため、リード群の存在確率の高い領域に対してフーリエ変換をかけることにより、パワースペクトル画像を作成した。得られたパワースペクトル画像から HLAC 特徴量を使用して 35 次元の特徴量を求め、サポートベクターマシンを使用することでコネクタ部品と非コネクタ部品とを分類できることが確認できた。提案手法を従来手法に組み込むことにより、コネクタ部品の分類性能が 40.0[%]から 91.3[%]に、部品全体では 89.2[%]から 96.0[%]へと改善できることを示した。

第4章ではコネクタ部品からのリード形状データの自動生成手法について検討した。部品画像中からリードを検出しデータ化するために、顔検出法で知られる Haar-Like 特徴と AdaBoost を利用した手法を応用した。更にリードの誤検出を削減するために、リード群のもつ周期性の特徴を、音声関係のピッチ検出手法として知られる NSDF を応用した手法で検出し、誤検出を削減することに成功した。これらのシステムを使用することで、これまでに部品形状データの自動作成ができなかったコネクタ部品に対し、データ作成における工数削減

が期待できる。

本論文で提案した手法は、電子部品のリードという人工建造物の周期性に着目した手法である。本手法を応用することで、画像中から周期的な物体を抽出、あるいは除外することが可能になると考えられる。特にビルの窓や駐車場の航空写真などの人工建造物では周期的な構造をもつ物体が多く見られ、本手法を応用することで、これらの画像認識において効果的な認識が期待できる。

近年、コンピュータの画像認識能力はCPUやGPUの進化により高性能化している。本論文では部品の形状データの自動認識について述べたが、部品形状だけでなく、プリント基板の形状や配線パターンを認識することで、対象部品の装着座標の調整も自動化できる可能性がある。今後、これらの技術について更に検討を行い、より使いやすく高精度なシステムの構築を実現していきたい。

# 謝辞

本研究を行うにあたり、終始懇切丁寧なるご指導を頂きました名古屋工業大学 梅崎太造教授、伊藤孝行教授、本谷秀堅教授に謹んで感謝いたします。

また、本研究を行うにあたりご協力いただいた、田口亮助教をはじめ、中部大学 保黒政大准教授、服部公央亮助手、梅崎・田口研究室関係者各氏に深く感謝します。

最後に、私が名古屋工業大学大学院博士後期課程に進むことを快諾してくださいました、富士機械製造株式会社ならびに、富士機械製造株式会社 須原信介専務取締役、田代雅幸部長、中村伸隆部長、星川和美氏、大池博史氏、他各氏に深く感謝いたします。

## 参考文献

- [1] 株式会社矢野経済研究所, *携帯電話の世界市場に関する調査結果2015*. 2015.
- [2] 一般社団法人電子情報技術産業協会, *2015年度版 実装技術ロードマップ*. 2015.
- [3] 村田製作所, “0201サイズ 積層セラミックコンデンサ.” [Online]. Available: <http://www.murata.com/ja-jp/about/newsroom/news/product/capacitor/2014/0507>.
- [4] ローム株式会社, “世界最小チップ抵抗器.” [Online]. Available: [http://www.rohm.co.jp/documents/11546/3002227/Micro\\_2-D.pdf](http://www.rohm.co.jp/documents/11546/3002227/Micro_2-D.pdf).
- [5] N. Dalal and B. Triggs, “Histograms of Oriented Gradients for Human Detection,” *CVPR '05 Proc. 2005 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. - Vol. 1*, pp. 886–893, 2005.
- [6] 藤吉弘亘, 山内悠嗣, and 山下隆義, “Boostingに基づく特徴量の共起表現による人検出,” *電子情報通信学会論文誌 D*, vol. 92, no. 8, pp. 1125–1134, 2009.
- [7] 藤吉弘亘, “Gradientベースの特徴抽出-SIFTとHOG-,” *電子情報通信学会技術研究報告*, vol. 107, no. 206, pp. 211–224, 2007.
- [8] 松島千佳, 山内悠嗣, 山下隆義, and 藤吉弘亘, “人検出のための Real AdaBoost に基づく HOG 特徴量の効率的な削減法,” *情報処理学会研究報告. Cvim, [コンピュータビジョンとイメージメディア]*, vol. 2009, pp. 1–8, 2009.
- [9] B. Wu and R. Nevatia, “Detection of multiple, partially occluded humans in a single image by bayesian combination of edgelet part detectors,” *Proc. IEEE Int. Conf. Comput. Vis.*, vol. I, pp. 90–97, 2005.
- [10] P. Viola and M. Jones, “Rapid object detection using a boosted cascade of simple features,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 1, pp. 511–518, 2001.
- [11] 柳井啓司, “一般物体認識の現状と今後 The Current State and Future Directions on,” *情報処理学会論文誌, コンピュータビジョンとイメージメディア*, vol. 48, no. SIG 16(Cvim 19), pp. 1–24, 2007.
- [12] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray, “Visual categorization with bags of keypoints,” *Proc. ECCV Int. Work. Stat. Learn. Comput. Vis.*, vol. 1, no. 1–22, pp. 59–74, 2004.
- [13] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2323, 1998.

- [14] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks,” *Adv. Neural Inf. Process. Syst.*, pp. 1–9, 2012.
- [15] L. Deng, “The MNIST Database of Handwritten Digit Images for Machine Learning Research,” *IEEE Signal Process. Mag.*, no. November, pp. 141–142, 2012.
- [16] 江寄弘健, 鍵井清幸, 梅崎太造, and 堀越哲美, “電子部品の階層型自動分類システムの開発,” *電気学会論文誌 C*, vol. 126, no. 12, pp. 1447–1453, 2000.
- [17] 江寄弘健, “電子部品実装における画像認識技術の応用に関する研究,” *博士論文(第2章)*.
- [18] J. W. Cooley and J. W. Tukey, “An Algorithm for the Machine Calculation of Complex Fourier Series,” *Math. Comput.*, vol. 19, 1965.
- [19] Fftw@fftw.org, “FFTW.” [Online]. Available: <http://www.fftw.org/>.
- [20] “FFTPACK.” [Online]. Available: <http://www.netlib.org/fftpack/index.html>.
- [21] T. S. Project, “FFTSS: 高速フーリエ変換ライブラリ.” [Online]. Available: <http://www.ssisc.org/fftss/>.
- [22] T. OOURA, “General Purpose FFT,” *Research Institute for Mathematical Sciences*. [Online]. Available: <http://www.kurims.kyoto-u.ac.jp/~ooura/index-j.html>.
- [23] 尾崎貴洋, 山内悠嗣, and 藤吉弘亘, “Joint HOG特徴を用いた2段階 AdaBoostによる車両検出,” *動的画像処理実利用化ワークショップ*, no. 11–13, pp. 101–106, 2008.
- [24] 三井相和, 山内悠嗣, and 藤吉弘亘, “Joint 特徴量を用いた 2 段階 Boosting による物体検出,” *電子情報通信学会論文誌 D*, vol. 92, no. 9, pp. 1591–1601, 2009.
- [25] T. Mita, T. Kaneko, and O. Hori, “Joint Haar-like features for face detection,” *Proc. IEEE Int. Conf. Comput. Vis.*, vol. II, pp. 1619–1626, 2005.
- [26] 大津展之, “パターン認識における特徴抽出に関する数理的研究,” *電子技術総合研究所研究報告*, 1981.
- [27] 栗田多喜夫, 堀田一弘, and 三島健稔, “Log-Polar画像の高次局所自己相関特徴を用いた大きさ・回転に不変な2次元図形及び顔画像の認識,” *電子情報通信学会技術研究報告(PRMU, パターン認識・メディア理解)*, vol. 96(599), pp. 151–158, 1997.
- [28] T. Kurita, N. Otsu, and T. Sato, “A face recognition method using higher order local autocorrelation and multivariate analysis,” *Int. Conf. Pattern Recognit.*, vol. 2, pp. 213–216, 1992.

- [29] 白木孝義 and 大津展之, “HLAC特徴を用いた照明変化に頑健な物体認識,” 電子情報通信学会技術研究報告, vol. 105, no. 302, pp. 59–63, 2005.
- [30] 依田育士 and 坂上勝彦, “3次元動き情報を利用した複数対象物の抽出とその実時間認識,” 電子情報通信学会論文誌 D(画像・パターン認識, コンピュータビジョン), vol. J81-D-2, no. 8, pp. 2043–2051, 1998.
- [31] 依田育士 and 坂上勝彦, “高次局所自己相関特徴を使った多眼多重解像度画像からの物体認識の試み,” 電子情報通信学会技術研究報告(PRMU, パターン認識・メディア理解), vol. 98(275), pp. 1–7, 1998.
- [32] 宮田繁春 and 中田博之, “高次局所自己相関特徴を用いたカラー画像からの顔検出,” 日本機械学会講演論文集, no. 015–1, pp. 243–244, 2001.
- [33] F. Goudail, E. Lange, T. Iwamoto, K. Kyuma, and N. Otsu, “Face Recognition System Using Local Autocorrelations and Multiscale Integration,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 10, pp. 1024–1028, 1996.
- [34] 石原辰也 and 大津展之, “高次局所自己相関特徴の自己回帰係数によるジェスチャー認識,” 電子情報通信学会技術研究報告. PRMU, パターン認識・メディア理解, vol. 103, no. 737, pp. 61–66, 2004.
- [35] 井手徳隆 and 石川眞澄, “視覚情報に基づく異種物体の実時間検出,” 電子情報通信学会技術研究報告. NC, ニューロコンピューティング, vol. 104, no. 758, pp. 161–166, 2005.
- [36] 豊田崇弘 and 長谷川修, “テクスチャ識別のためのマスクパターンによる特徴抽出法 (テーマ関連セッション (1)),” 情報処理学会研究報告. CVIM,[コンピュータビジョンとイメージメディア], vol. 91, pp. 77–84, 2004.
- [37] 豊田崇弘 and 長谷川修, “高次局所自己相関特徴の拡張,” 画像電子学会誌, vol. 34, no. 1, pp. 1–8, 2005.
- [38] 堀田一弘, 栗田多喜夫, and 三島健稔, “Log-Polar画像の高次局所自己相関特徴による大きさに不変な顔画像の認識,” 電子情報通信学会論文誌 D, vol. 180-D-2, no. 8, pp. 2209–2217, 1997.
- [39] T. Ojala, M. Pietikäinen, and D. Harwood, “A comparative study of texture analysis algorithms in textile inspection applications,” *Pattern Recognit.*, vol. 29, no. 1, pp. 51–59, 1996.
- [40] T. Ojala, M. Pietikäinen, and T. Mäenpää, “Multiresolution gray-scale and rotation invariant texture classification with local binary patterns,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, 2002.
- [41] W. Zhang, S. Shan, W. Gao, X. Chen, and H. Zhang, “Local Gabor Binary Pattern Histogram Sequence (LGBPHS): A novel non-statistical model for face representation and recognition,” *Proc. IEEE Int. Conf. Comput. Vis.*, vol. I, pp. 786–791, 2005.



- [42] G. Zhao and M. Pietikäinen, “Dynamic texture recognition using local binary patterns with an application to facial expressions.,” *Pattern Anal. Mach. Intell.*, vol. 29, no. 6, pp. 915–928, 2007.
- [43] Z. Lei, S. Liao, R. He, M. Pietikainen, and S. Z. Li, “Gabor volume based local binary pattern for face representation and recognition,” *Autom. Face Gesture Recognition, 2008. FG '08. 8th IEEE Int. Conf.*, pp. 1–6, 2008.
- [44] B. Zhang, Y. Gao, S. Zhao, and J. Liu, “Local derivative pattern versus local binary pattern: face recognition with high-order local pattern descriptor.,” *IEEE Trans. Image Process.*, vol. 19, no. 2, pp. 533–544, 2010.
- [45] X. Tan and B. Triggs, “Enhanced local texture feature sets for face recognition under difficult lighting conditions,” *IEEE Trans. image Process.*, vol. 19, no. 6, pp. 1635–1650, 2010.
- [46] J. Ren, X. Jiang, and J. Yuan, “Relaxed Local Ternary Pattern for Face Recognition,” *Proc. IEEE Int. Conf. Image Process.*, pp. 2–6, 2013.
- [47] 村松千左子, 張敏, 原武史, 遠藤登喜子, and 藤田広志, “中心方向性Local Ternary Patternを用いたマンモグラムにおける腫瘍の良悪性鑑別法の検討,” *電子情報通信学会技術研究報告. MI, 医用画像*, vol. 113(410), pp. 327–330, 2014.
- [48] F. Han, Y. Shan, and R. Cekander, “A two-stage approach to people and vehicle detection with hog-based svm,” *Perform. Metrics Intell. Syst. 2006 Work.*, pp. 133–140, 2006.
- [49] G. Guo, S. Z. Li, and K. K.-L. Chan, “Face recognition by support vector machines,” *Autom. Face Gesture Recognition, 2000. Proceedings. Fourth IEEE Int. Conf.*, pp. 196–201, 2000.
- [50] B. Heisele, P. Ho, and T. Poggio, “Face Recognition with Support Vector Machines: Global versus Component-based Approach,” *Comput. Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE Int. Conf. on.*, vol. 2, pp. 688–694, 2001.
- [51] 瀧川えりな, 細井聖, and 川出雅人, “顔画像による人種推定技術 (顔とジェスチャの認識),” *電子情報通信学会技術研究報告. HIP, ヒューマン情報処理*, vol. 103, no. 454, pp. 19–24, 2003.
- [52] M. Yoshikawa, M. Mikawa, and K. Tanaka, “Real-time hand motion estimation using EMG signals with support vector machines,” *2006 SICE-ICASE Int. Jt. Conf.*, no. 1, pp. 593–598, 2006.
- [53] 岡部孝弘 and 佐藤洋一, “照明変化をともしなう物体認識へのサポートベクターマシンの適用,” *情報処理学会論文誌*, vol. 44, no. Cvim 6, pp. 22–29, 2003.
- [54] 三浦純, 森田英夫, ヒルドミヒヤエル, and 白井良明, “SVM による物体と位置の視覚学習に基づく屋外移動ロボットの位置推定,” *日本ロボット学会誌*, vol. 25, no. 5, pp. 792–798, 2007.

- [55] 武藤亮介, 嶋田和孝, and 遠藤勉, “SVM と逐次学習を併用した HOG 特徴による手形状認識手法,” *電子情報通信学会技術研究報告. PRUM パターン認識・メディア理解*, vol. 109, no. 470, p. 459, 2010.
- [56] ビショップ C. M., *パターン認識と機械学習 下(ベイズ理論による統計的予測)*. 丸善出版, 2012.
- [57] N. Cristianini, S.-T. John, and 大北剛, *サポートベクターマシン入門*. 共立出版, 2005.
- [58] 江寄弘健, 鍵井清幸, 梅崎太造, and 堀越哲美, “IC 部品形状データの自動生成,” *電気学会論文誌 D(産業応用部門誌)*, vol. 127, no. 2, pp. 152–157, 2007.
- [59] P. McLeod and G. Wyvill, “A Smarter Way to Find Pitch,” *ICMC Proc.*, pp. 138–141, 2005.
- [60] 松田徹也, 広瀬啓吉, and 峯松信明, “経験的モード分解による主構造抽出を介した雑音環境下における音声信号の基本周波数推定,” *電子情報通信学会技術研究報告. EA, 応用音響*, vol. 109, no. 55, pp. 49–54, 2009.
- [61] R. E. Schapire and Y. Singer, “A Decision-Theoretic Generalization of on-Line Learning and an Application to Boosting,” *Comput. Learn. theory. Springer Berlin Heidelb.*, 1995.
- [62] 林伸治 and 長谷川修, “低解像度画像からの顔検出,” *画像電子学会誌*, vol. 34, no. 6, pp. 726–737, 2005.
- [63] R. E. Schapire and Y. Singer, “Improved boosting algorithms using confidence-rated predictions,” *Mach. Learn.*, vol. 37, no. 3, pp. 297–336, 1999.
- [64] P. Viola and M. J. Jones, “Robust Real-time Object Detection,” *Int. J. Comput. Vis.*, no. February, pp. 1–30, 2001.
- [65] R. Lienhart and J. Maydt, “An extended set of Haar-like features for rapid object detection,” *Int. Conf. Image Process.*, vol. 1, pp. 0–3, 2002.
- [66] F. Porikli, “Integral histogram: A fast way to extract histograms in Cartesian spaces,” *Proc. - 2005 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognition, CVPR 2005*, vol. I, pp. 829–837, 2005.
- [67] G. Muhammad, “Extended average magnitude difference function based pitch detection,” *Int. Arab J. Inf. Technol.*, vol. 8, no. 2, pp. 197–203, 2011.

# 研究業績一覧

## 学術論文（査読あり） 主：3，副：0

- [1] 鬼頭秀一郎，服部公央亮，田口亮，保黒政大，梅崎太造，“表面実装部品のリード形状データの自動生成”，電気学会論文誌C, Vol. 135, No.4, pp.365-371, 2014.
- [2] 鬼頭秀一郎，服部公央亮，田口亮，梅崎太造，“リードの周期性に着目した電子部品自動分類システム”，電気学会論文誌C, Vol.135, No.10, pp.1229-1235, 2015.
- [3] Shuichiro Kito, Koosuke Hattori, Ryo Taguchi, Taizo Umezaki, “The Feature Extraction Method in Power Spectrum Images for Connector Parts”, Journal of the Institute of Industrial Applications Engineers, Vol.3, No. 4, pp.154-160, 2015.

## 国際会議（査読あり） 主：1，副：0

- [4] Shuichiro Kito, Koosuke Hattori, Ryo Taguchi, Taizo Umezaki, “Automatic Classification System of Electric Parts Focused on Periodicity of Leads”, The 3rd International Conference on Industrial Application Engineering, pp.575-580, 2015.

## 研究会・口頭発表（査読あり） 主：1，副：0

- [5] 鬼頭秀一郎，梅崎太造，“電子部品のリード形状データの自動生成”，SSII2014(Symposium on Sensing via Image Information), IS1-27, 2014.

## 特許出願

- [6] PCT/JP2014/065043 リード画像認識方法及びリード画像認識装置並びに画像処理用部品データ作成方法及び画像処理用部品データ作成装置

- [7] PCT/JP2015/057220 部品種類自動判別方法及び部品種類自動判別システム並びに画像処理用部品データ作成方法及び画像処理用部品データ作成システム
- [8] PCT/JP2015/067241 挿入部品位置決め検査方法及び挿入部品実装方法並びに挿入部品位置決め検査装置及び挿入部品実装装置
- [9] 特願2015-120766 挿入部品実装方法及び挿入部品実装装置
- [10] 特願2015-173822 多重通信システム及び画像伝送方法

## 表彰

- [11] 平成27年度 名古屋工業大学基金 学生研究奨励賞受賞