

画像認識技術を用いた検査工程の作業時間推定

渡辺 博己 曾賀野 健一 棚橋 英樹

Estimation of Inspection Working Time by Using Image Recognition

Hiroki WATANABE Kenichi SOGANO Hideki TANAHASHI

あらまし 本研究では、カイゼン活動における作業時間計測を支援するために、ビデオカメラで撮影した動画ファイルについて、各フレームを作業内容に応じて分類することにより、作業時間を推定するシステムを構築した。作業画像の分類では、画像特徴表現手法の一つであるBag-of-featuresにより画像特徴量を抽出し、Support Vector Machineにより各フレームを分類した。また、作業時間の推定では、分類結果に基づいて単位時間当たりの代表となる作業を選定し、作業順序を示す作業チャートを作成することにより作業時間を算出した。実験では、検査工程の作業シーンを対象に、目視確認作業、バリ取り作業、その他の作業の三つのカテゴリに分類することにより作業チャートを作成し、各作業の作業時間を推定した。

キーワード カイゼン、作業時間計測、画像認識、Bag-of-features、Support Vector Machine

1. はじめに

若年労働者の減少、女性労働者の増加、外国人労働者の受け入れなど、企業の雇用形態が大きく変化している。また、少子高齢化、GDPの減退、消費者行動の変化など、消費者ニーズの多様化により、製品ライフサイクルが短縮化し、多くの企業で多品種少量生産への対応に迫られている。そのため、作業の標準化、作業者の能力向上や能率改善に向けた努力など、生産性向上のための生産管理支援に対する要望が高まっている。

こうした中、企業の作業現場では生産性向上のために、積極的に作業カイゼンに取り組み、生産工程の効率化を図っている。作業カイゼンの手法には、IE (Industrial Engineering) と呼ばれる工学的手法があるが、IEでは、作業時間を計測・分析することで、標準的な作業能力の算出、人件費等の原価管理、作業負担の軽減や作業の効率化等を図ることができる^[1]。

IEを活用した作業カイゼンにおいて、最初に取り組みなければならないのが現状分析である。これは、作業にどの程度の時間を要しているのかを調査することであり、カイゼン担当者は、ストップウォッチやビデオカメラを持って作業現場に張り付き、作業者が行う作業の様子を観察し、作業時間を計測する必要がある。しかし、この調査には、膨大な時間と手間を要し、計測ノウハウを有する人材の育成や人件費等の問題から、現状分析に至る前にカイゼン活動が停滞することも少なくない^[2]。また、人が時間を計測する場合には、長時間（長期間）にわたる作業時間の計測が困難であるため、時間帯や人による作業のムラ等を評価するまでに至っていないのが現状で

ある^[3]。

そこで、本研究では、カイゼン活動における作業時間の計測を支援するために、作業の様子を撮影した動画ファイルの各フレームを作業内容に応じて分類することにより、作業時間を推定するシステムを構築した。具体的には、各フレームの対象領域に対して、一般物体認識のアプローチとして数多く用いられているBag-of-featuresにより画像特徴量を抽出し、Support Vector Machine (SVM)により識別処理を行う方式^[4]を実装することで、各フレームを作業カテゴリに分類するとともに、分類結果に基づいて単位時間毎の作業順序を示す作業チャートを生成する技術を開発した。本報告では、検査工程における作業の様子を撮影した動画ファイルについて、各フレームを作業カテゴリに分類し、作業チャートを作成する実験を行ったので、その結果についても報告する。

2. 対象作業

本研究で対象とする作業内容については、アルミダイカスト部品の検査工程において、作業時間の把握を要望する県内企業の作業現場を対象に、部品の目視確認作業とバリ取り作業の作業時間を推定することにした。

正常な部品ばかりであれば、「全作業時間＝目視確認作業時間」となるはずであるが、実際にはバリ取り作業に時間を要し、その時間を無視することが困難な状況となっている。そのため、検査工程における作業時間の把握は生産性向上を図るうえで切実な課題となっている。検査工程の作業現場の様子を図1に示す。図には、守秘義務により画像の一部にフィルタ処理を加えている。



図1 作業現場の様子

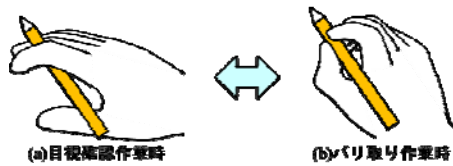


図2 バリ取り工具の把持状態

検査工程では、作業者は椅子に座り、机上に設置された拡大鏡を通して、5cm×2cm×1cm程度のアルミダイカスト部品を検査する。検査の手順は以下のとおりである。

- (ア) 左手外側にある未作業部品置き場から部品を1個掴む。
- (イ) 拡大鏡の下で両手を使って把持し、部品を回転させながら全体を目視し、良品かどうかを確認する。
- (ウ) バリがない場合は、右手外側にある既作業部品箱に部品を放出し、(ア)へ移行する。
- (エ) バリがある場合は、作業開始時から図2(a)の状態 で把持していたバリ取り工具を、(b)の状態に把持し直してバリを削り取り、(イ)または(ウ)に移行する。

本研究では、(イ)を目視確認作業、(エ)をバリ取り作業、(ア)、(ウ)、それ以外の状況をその他の作業として、三つのカテゴリに動画ファイルの各フレームを分類することとした。

3. 画像認識による識別処理

本研究では、画像を局所特徴量の集合とみなすことにより画像特徴量を抽出するBag-of-featuresと、教師あり学習を用いるパターン認識モデルの一つであるSVMにより、動画ファイルの各フレームをカテゴリに分類した。識別処理の流れを図3に示す。以下、詳細について述べる。

3.1 Bag-of-featuresによる画像分類

Bag-of-featuresでは、最初に、画像から局所特徴量を抽出し、ベクトル量子化する。ベクトル量子化においては、コードブックと呼ばれる辞書を用いてベクトル量子化ヒストグラムを生成する。このヒストグラムを特徴ベクトルとして識別器に入力することにより、識別器の構築や識別器による分類が可能となる。なお、本研究では、局所特徴量としてSURF (Speeded Up Robust Features) ^[5]を

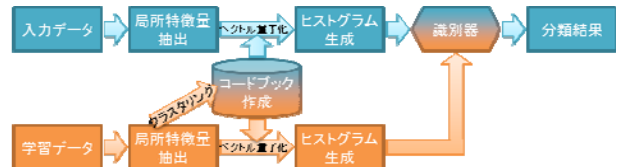


図3 識別処理の流れ

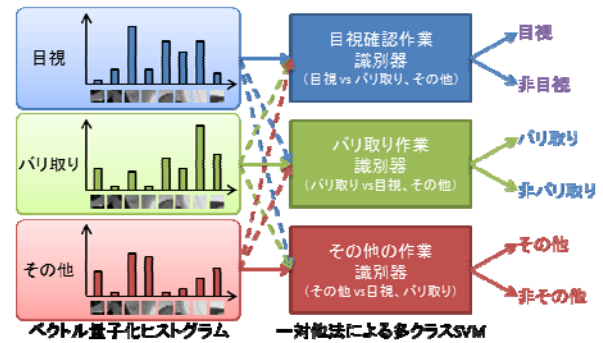


図4 多クラスSVMの概要

使用した。

3.2 コードブックの生成

コードブックとは、学習する全ての画像の局所特徴量を元に生成され、画像の特徴を一つの辞書として表したものである。コードブックの生成にあたっては、学習する全画像から抽出した局所特徴量をK-means法^[6]でクラスタリングし、各クラスターのセントロイド(重心)を要素として構成する。コードブックを構成する各要素はVisual Wordと呼ばれ、Visual Word数が多いほど認識性能が高くなることが知られている。しかし、計算コストも高くなるため、一般的には数千から数万にすることが望ましいとされている。

3.3 ベクトル量子化ヒストグラムの生成

ベクトル量子化ヒストグラムの生成にあたっては、まず、画像中の局所特徴量を抽出し、各局所特徴量について、最も近いVisual Wordをコードブックから求める。次に、対応するVisual Wordへの投票を繰り返し、コードブックに対するVisual Wordの出現頻度を表すヒストグラムを生成する。最後に、局所特徴量の数が画像により異なるため、ヒストグラムを局所特徴量の数で正規化し、特徴ベクトルとして生成する。

3.4 識別器の構成

識別器にはSVMを使用した。SVMは、基本的には2クラス識別器であるため、多クラスに対して識別するには、複数のSVMを組み合わせる必要がある^[7]。そこで、本研究では、一対他法で三つのSVMを構築し、識別器を構築した。各SVMからは、それぞれに結果が出力されるが、本研究では、超平面からの距離により、画像を三つの作業カテゴリの内の一つに分類した。図4に構築した多クラスSVMの概要を示す。なお、各SVMに用いるカーネルとしてガウシアンカーネルを使用した。

3.5 プログラムへの実装

プログラムは、短期間で開発が可能なMicrosoft社の

表1 動画ファイルの記録時間とフレーム数

ファイル	記録時間	フレーム数
Movie-1	35:09.073	63,210
Movie-2	35:09.073	63,210
Movie-3	5:54.821	10,635
Movie-4	35:09.073	63,210
Movie-5	35:09.073	63,210
Movie-6	22:44.830	40,905
計	2:49:15.943	304,380

Visual Studio Expressを使い、C#言語で開発した。また、画像処理技術の実装については、OpenCVをC#言語から使用するため、ラッパライブラリであるEmgu CV^[8]を利用した。

4. 実験

4. 1 実験データの取得

実験に使用する動画ファイルは、図1の作業の様子を、拡大鏡のアームに固定したビデオカメラで撮影することにより取得した。カメラは、SONY製 Action Cam (HDR-AS200V)を使用し、フレームサイズは1920×1080ピクセル、フレームレートは29fps、ファイルフォーマットはMP4形式で動画ファイルに記録した。

作業は、9時から12時10分まで行われ、90分の作業後、10分の休憩をはさみ、再度90分の作業が行われた。動画ファイルは、休憩時間を除く約170分の作業について、一人の熟練した作業員の作業シーンを連続して撮影した。なお、動画ファイルは、1ファイルあたりの最大記録時間が35分9秒であるため、6つの動画ファイルとして取得した。

表1に、取得した動画ファイルの記録時間とフレーム数の内訳を示す。Movie-1からMovie-3が休憩前の作業を撮影したもの、Movie-4からMovie-6が休憩後の作業を撮影したものであり、動画ファイルの数値の順に時間が経過している。

4. 2 学習データの作成

学習データの作成にあたっては、Movie-1の全フレームについて、①目視確認作業、②バリ取り作業、③その他の作業の作業のカテゴリに手作業で分類し、順に①28,011個、②21,601個、③13,598個の計63,210個の画像を学習データとして用意した。各カテゴリの画像例を図5に示す。なお、図には、図1と同様に、画像の一部にフィルタ処理を加えている。

4. 3 前処理と後処理

実験では、処理速度の高速化及び認識性能の高精度化を図るために、3章の識別処理の前後に以下の処理を追加した。

動画ファイルの各フレームは、サイズが1920×1080であるため、そのままプログラムで利用すると計算コストが高くなる。そのため、実験では、前処理として各フレ



図5 作業カテゴリの画像例

ームの解像度を640×360にサイズダウンした。また、目視確認作業、バリ取り作業は拡大鏡を通して行われることから、それらの作業を行う位置はほぼ固定されているため、サイズダウンした画像にROI (Region of Interest)を設定し、更に計算コストを軽減した。実験では、サイズダウンした画像の左上から(180, 180)の位置に幅240、高さ180でROIを設定した。なお、4.2節の学習データは、ROIを設定して取得したカラー画像をビットマップ形式で保存したものである。

また、全ての分類結果を時系列情報に基づき並び替え、誤分類をノイズとみなすことにより除去する平滑化処理を後処理として加えた。平滑化処理では、対象フレームの前後の分類結果が同一で、対象フレームの分類結果が前後と異なる時、対象フレームの分類結果を前後と同一にすることにより、誤分類を除去した。

4. 4 Visual Word数の選定

3.2節で述べたとおり、Visual Word数を多くすると認識性能は高まるが、計算コストも高くなる。そこで、動画ファイルの各フレームを作業カテゴリに分類するにあたり、Visual Word数を選定するための実験を行った。実験では、4.2節の全ての学習データを使用し、Visual Word数を変化させた時のMovie-1の全てのフレームに対する認識率と処理時間の推移を調査した。図6に結果を示す。実線は平滑化処理後の認識率であり、平滑化処理前(識別処理後)の認識率を点線で示す。

Visual Word数が増加するにつれて、上下に変動はあるものの、徐々に認識性能が高くなり、平滑化処理前の全作業のグラフでは、Visual Word数が3,000までは認識率が90%未満であったが、4,000からは90%以上となった。一方で、Visual Word数が増加するにつれて、処理時間が徐々に長くなった。そのため、本節以降の実験で用いるVisual Word数は4,000とした。

4. 5 学習サンプル数の選定

本システムを実際に作業現場で運用する場合、学習デ

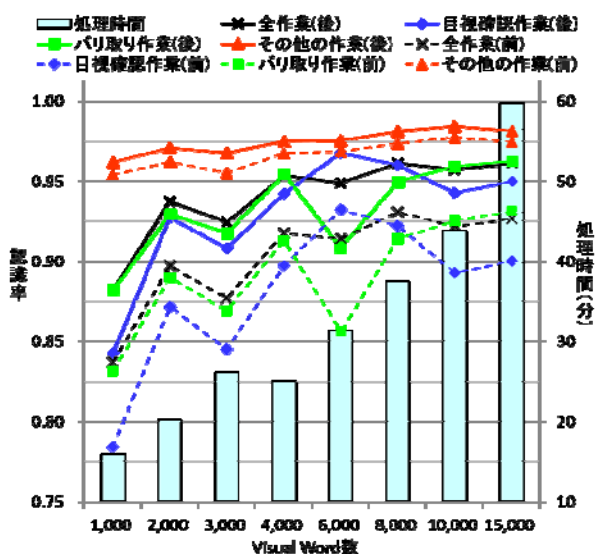


図6 Visual Word数の違いによる認識率と処理時間の推移 ※(後)は平滑化処理後、(前)は平滑化処理前の認識率

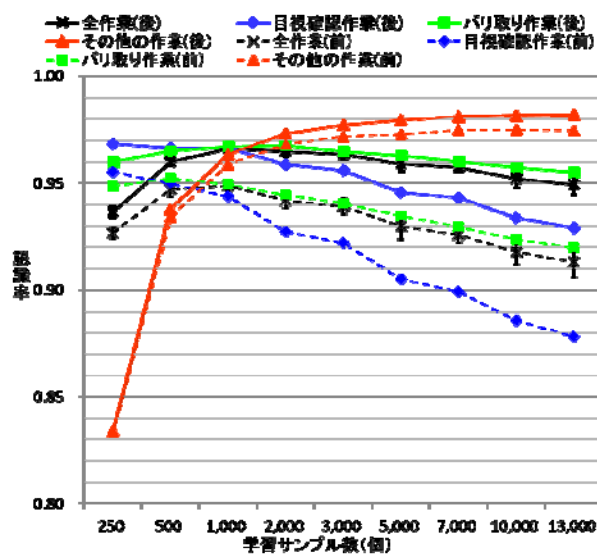


図7 学習サンプル数の違いによる平均認識率の推移 ※(平)は平滑化処理後の平均認識率

ータのサンプル数が多く、事前に学習データを準備（学習データとして使用する画像を手作業で作業毎に分類）する作業の負担が大きいたことが課題となる。そこで、学習データのサンプル数の違いによる認識率への影響について実験することにより、少ないサンプル数でも認識性能を維持することが可能であるかを検証した。

実験では、4.2節の学習データから各カテゴリについて同数のサンプルをランダムに抽出することとし、サンプル数を変更した時のMovie-1の全てのフレームに対する認識率の推移を調査した。なお、ランダムサンプリングによる偏りを評価するために、各サンプル数について10回ずつ実験を繰り返した。また、各実験において、ランダムに抽出したサンプルは全て異なり、同一のサンプルは含まれていない。各サンプル数における平均認識率の推移を図7に示す。実線は平滑化処理後の平均認識率であり、平滑化処理前の平均認識率を点線で示す。

検証の結果、サンプル数が1,000個の時、全作業の平均認識率が、平滑化処理の前後共に、最も高くなった。また、カテゴリ間の平均認識率の差も、平滑化処理の前後共に、最も小さくなった。グラフを概観すると、その他の作業については、サンプル数が増加するに伴い、認識性能が高くなる傾向があるが、目視確認作業、バリ取り作業、及び全作業の認識性能については、低くなる傾向が見られた。この原因としては、サンプル数が増加することにより、その他の作業に関するVisual Wordが増加し、その結果、目視確認作業、バリ取り作業の認識性能を低下させていると考えることができる。これらのことから、撮影した動画ファイルから各作業について1,000個の学習データを収集し、Visual Word数を4,000として学習すれば、十分な認識性能を得ることができると言える。なお、標準偏差の値が小さいことから、ランダムにサンプルを抽出しても、認識性能に大きな影響は与えないと考えら

れる。

4. 6 作業チャートの作成

作業チャートは、時間と作業内容から構成されるテーブルで、作業手順を表す。フレーム単位で出力すれば、時間分解能の高い作業チャートを作成することが可能であるが、識別処理の認識性能がそのまま作業チャートに反映されるため、高い認識性能が必要となる。作業時間の把握を依頼された企業からは、秒単位の作業時間が把握できれば十分であるとの回答が得られているため、本実験では、1秒間隔で分類結果の代表となる作業（代表作業）を選定し、その結果により作業チャートを作成することにした。なお、代表作業の選定にあたっては、1秒間の中で最も多く分類された作業カテゴリを選定した。

実験では、Movie-1から6までの全てのフレームを対象に、識別処理、平滑化処理を行うことにより各フレームを作業カテゴリに分類し、認識率を求めた。また、分類結果から作業チャートを作成することにより正誤を判定し、分類率を求めた。学習データは、4.2節で作成した学習データからカテゴリ毎に1,000個のサンプルをランダムに抽出した。正誤の判定においては、手作業で各カテゴリに分類した結果から作成した作業チャート（正解チャート）と本システムによる分類結果から作成した作業チャート（推定チャート）とを比較した。これらの実験を10回繰り返すことにより認識率、及び分類率の平均値を求めた。なお、各実験において、ランダムに抽出したサンプルは全て異なり、同一のサンプルは含まれていない。平滑化処理前後の平均認識率を表2に、平滑化処理の有無により作業チャートを作成した場合の分類率を表3に、本システムで作成した作業チャートから算出した各作業時間の総計を表4に示す。

表2に示すとおり、学習データのサンプル数を各カテゴリについて1,000個としても、高い認識性能を得ることが

表2 平滑化処理後の全フレームに対する平均認識率（試行回数：10回）※括弧内は平滑化処理前の結果

ファイル	Movie-1	Movie-2	Movie-3	前半	Movie-4	Movie-5	Movie-6	後半	全体
目視確認作業	0.963 (0.940)	0.952 (0.920)	0.936 (0.898)	0.956 (0.928)	0.891 (0.842)	0.905 (0.855)	0.825 (0.770)	0.879 (0.829)	0.915 (0.875)
バリ取り作業	0.969 (0.952)	0.934 (0.908)	0.927 (0.898)	0.950 (0.928)	0.881 (0.844)	0.897 (0.862)	0.902 (0.860)	0.892 (0.855)	0.921 (0.891)
その他の作業	0.965 (0.961)	0.956 (0.951)	0.950 (0.946)	0.960 (0.955)	0.953 (0.946)	0.959 (0.949)	0.948 (0.938)	0.955 (0.946)	0.957 (0.949)
全作業	0.965 (0.949)	0.947 (0.923)	0.936 (0.910)	0.955 (0.934)	0.906 (0.872)	0.923 (0.892)	0.881 (0.842)	0.906 (0.872)	0.928 (0.900)
標準偏差	0.001 (0.002)	0.004 (0.005)	0.005 (0.006)	0.003 (0.003)	0.008 (0.010)	0.008 (0.010)	0.012 (0.012)	0.008 (0.010)	0.005 (0.007)

表3 平滑化処理有りの作業チャートにおける平均分類率（試行回数：10回）※括弧内は平滑化処理無しの結果

ファイル	Movie-1	Movie-2	Movie-3	前半	Movie-4	Movie-5	Movie-6	後半	全体
目視確認作業	0.973 (0.969)	0.967 (0.963)	0.964 (0.963)	0.970 (0.966)	0.942 (0.937)	0.954 (0.946)	0.897 (0.891)	0.935 (0.929)	0.952 (0.946)
バリ取り作業	0.981 (0.979)	0.954 (0.952)	0.951 (0.954)	0.967 (0.965)	0.912 (0.911)	0.934 (0.931)	0.956 (0.957)	0.932 (0.931)	0.949 (0.948)
その他の作業	0.966 (0.970)	0.960 (0.965)	0.942 (0.948)	0.961 (0.966)	0.961 (0.968)	0.971 (0.974)	0.972 (0.975)	0.968 (0.972)	0.965 (0.970)
全作業	0.975 (0.973)	0.961 (0.960)	0.954 (0.956)	0.967 (0.966)	0.939 (0.938)	0.955 (0.953)	0.936 (0.934)	0.944 (0.943)	0.955 (0.953)
標準偏差	0.002 (0.003)	0.006 (0.006)	0.008 (0.006)	0.004 (0.004)	0.004 (0.004)	0.005 (0.006)	0.010 (0.010)	0.004 (0.005)	0.003 (0.003)

表4 作業チャートから算出した平均作業時間（試行回数：10回，単位：秒）※括弧内は正解との差

ファイル	Movie-1	Movie-2	Movie-3	前半	Movie-4	Movie-5	Movie-6	後半	全体
目視確認作業	930 (+4)	956 (+18)	143 (+6)	2,030 (+29)	921 (+22)	822 (+15)	531 (-30)	2,274 (+7)	4,303 (+35)
バリ取り作業	735 (+5)	685 (-8)	126 (-1)	1,546 (-4)	576 (-27)	508 (-15)	477 (+13)	1,561 (-29)	3,107 (-33)
その他の作業	444 (-10)	469 (-10)	86 (-5)	1,000 (-24)	613 (+5)	780 (0)	357 (+17)	1,750 (+22)	2,750 (-2)
計	2,110	2,110	355	4,575	2,110	2,110	1,365	5,585	10,160

できた。括弧内に平滑化処理前の平均認識率を示したが、平滑化処理前後で全作業に対する平均認識率が2.8%高くなったことから、平滑化処理を後処理として用いることは効果が高いと言える。しかし、休憩前後で認識率に差があること、時間の経過に従って認識率が低下するとともに標準偏差が大きくなる傾向があることから、均一な認識性能やより高い認識性能を必要とする場合には、これらのことを考慮した学習データの選定方法を検討しなければならない。

表3の作業チャートの平均分類率では、表2の平均認識率よりも高い結果となっていることから、1秒間隔で代表作業を選定することにより、識別処理における作業カテゴリへの誤分類を効果的に解消していると考えられる。特に、表2のMovie-4とMovie-6では、平滑化処理後の全作業の平均認識率の差が2.5%であったのに対し、表3の平滑化処理有りの平均分類率の差では0.3%に縮まっている。このことから、1秒間隔で代表作業を選定する作業チャートの分類率においては、僅かな認識率の差の影響は少ないと考えられる。一方、表2では平滑化処理の前後で全作業の平均認識率の差が2.8%であるのに対し、表3では平滑化処理の有無による平均分類率の差が0.2%であり、標準偏差よりも小さい結果となった。このことから、1秒間隔で代表作業を選定する場合、平滑化処理の効果は

無いものと判断できる。

表4では、平滑化処理後の作業チャートの作成結果に基づいて各作業の作業時間を集計したが、平均値ではあるものの35秒以下の誤差となり、作業間の時間割合を把握する上では、十分な結果が得られている。

4. 7 作業チャートの可視化

作業チャートを可視化する方法として、タイムチャート形式で図示した例を図8に示す。タイムチャートとして作業チャートを可視化することにより、作業の進捗状況の視認性が高まるだけでなく、バラツキ度合いも把握することができるようになり、生産管理に有用な情報を提供することが可能になると考えられる。

図9に正解チャートと推定チャートとの正誤の例をタイムチャートにより示す。推定チャートにおける誤分類の殆どは、アのような作業前後の時間のズレであるが、イのような作業の抜けやウのような誤認も見受けられるため、認識性能の向上や代表作業の選定方法を検討する必要がある。

5. まとめ

本研究では、検査工程における作業時間を把握するために、画像認識技術による作業分類を行い、分類結果から作業手順を示す作業チャートを作成するシステムを構

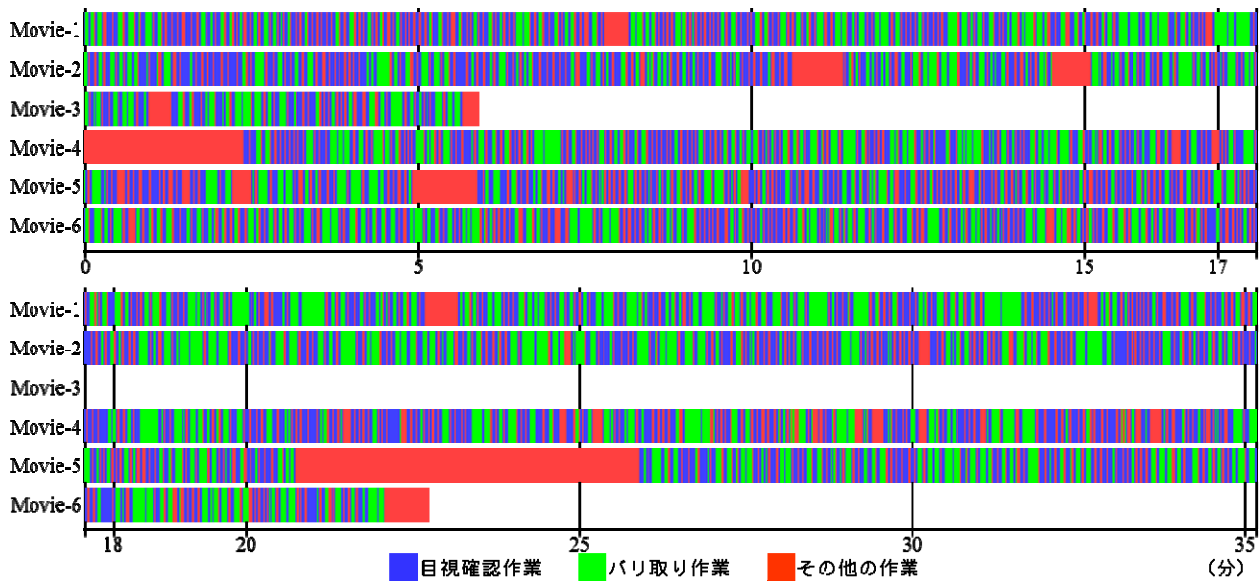


図8 作業チャートの可視化例



図9 作業チャート間の正誤例

築した。画像認識技術としては、Bag-of-featuresにより特徴ベクトルを抽出し、SVMを用いた一対他分類器により入力画像を分類する手法を用いた。実験では、特徴ベクトルの次元数（Visual Word数）や学習データのサンプル数の違いによる認識率への影響を評価し、作業シーンを撮影した動画ファイルの各フレームについて、作業カテゴリに分類するとともに、作成した作業チャートから各カテゴリにおける分類率や作業時間を推定した。

その結果、検査工程における作業内容を目視確認作業、バリ取り作業、その他の作業の三つのカテゴリに分類する識別処理については、Visual Word数が4,000の場合、各カテゴリについて1,000個の学習データを用意すれば、十分な認識性能が得られることが分かった。また、後処理として平滑化処理を行うことにより、認識性能が高められることが分かった。さらに、推定した作業時間の集計結果には誤差が含まれるものの、作業時間の割合を把握する上では、十分な結果が得られることも分かった。これらのことから、本システムは有用性が高いと判断できる。なお、本研究では、動画ファイルを入力データとしたが、約35分の動画ファイルの各フレームの分類にかかる処理時間は約27分（CPU：Core i7-4770K 3.50GHz）であり、動画ファイルの記録時間に比べて短時間で処理することが可能であることから、リアルタイムに取得するカメラ映像についても適用可能である。

今後は、他の作業者についても同様の動画ファイルを取得し、本システムの有効性を検証するとともに、他の作業についても作業分類の可能性を検証し、汎用性を高

めることを検討する。

文 献

- [1] 鈴木準, “IEを活用した現場改善のススメ”, ロジスティクス・ビジネス, Vol.9, pp.20-23, 2005.
- [2] 藤田彰久, “IE の基礎”, 建帛社, 1997.
- [3] 曾賀野健一, 渡辺博己, 棚橋英樹, “IEによる作業工程分析及び可視化手法の研究”, 岐阜県情報技術研究所研究報告, Vol.13, pp.47-50, 2011.
- [4] G. Csurka, C. Dance, L. Fan, J. Willamowski and C. Bray, “Visual Categorization with Bags of Keypoints”, In Proc. of ECCV Workshop on Statistical Learning in Computer Vision, pp.59- 74, 2004.
- [5] H. Bay, T. Tuytelaars and L. V. Gool, “SURF : Speeded Up Robust Features”, In Proc. of ECCV, pp.404-417, 2006.
- [6] J. B. MacQueen, “Some Methods of Classification and Analysis of Multivariate Observations”, In Proc. of 5th Berkeley Symposium on Math. Stat. and Prob., pp.281-297, 1967.
- [7] 村山善透, 松川徹, 栗田多喜夫, “多クラス識別問題における2クラス識別器の選択”, 電子情報通信学会パターン認識・メディア理解研究会, PRMU2009-121, pp.195-200, 2009.
- [8] Emgu CV, <http://www.emgu.com/wiki/index.php>.