

機械学習を用いた作業時間推定システムの開発

渡辺 博己 曾賀野 健一 松原 早苗 棚橋 英樹

Development of Work Time Estimation System using Machine Learning

Hiroki WATANABE Kenichi SOGANO Sanae MATSUBARA Hideki TANAHASHI

あらまし カイゼン活動における作業時間計測を支援するために、機械学習を用いた作業時間推定システムの構築を目指している。本研究では、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処理の精度を高めることを目的として、画像特徴表現手法の一つであるBag-of-Featuresにおける局所特徴量抽出手法、及びパターン認識モデルの一つであるSupport Vector Machineによる多クラス識別器構成手法について、幾つかの手法を検討し、画像分類実験を行った。また、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、時系列順に代表作業を並べることにより、作業チャートを生成するとともに、作業時間を算出し、本システムの有効性を確認した。

キーワード カイゼン、機械学習、作業時間推定、Bag-of-Features、Support Vector Machine

1. はじめに

製造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) ^[1]と呼ばれる工学的手法があるが、IEを活用した作業カイゼンにおいて、最初に取り組まなければならないのが現状分析である。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、作業者の作業状況を観察し、各作業にどの程度の時間を要しているのかを調査する。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]。

そこで本研究では、作業カイゼンにおける作業時間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、各作業の所要時間を推定するシステムの構築を目指している^[3]。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 (BoF) 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ている。

これまでに、アルミダイカスト部品の検査工程における作業画像について、SURF^[5]により局所特徴量を抽出し、一対他法による多クラスSVMを用いて識別することにより、目視確認作業、バリ取り作業、その他の作業の3つのクラスに分類するシステムを試作した^[6]。本年度は、試作システ

ムの認識精度を高めるために、BoFにおける局所特徴量の抽出手法、及びSVMによる多クラス識別器の構成手法について検討した。実験では、幾つかの局所特徴量抽出手法を比較するとともに、多クラス識別器の構成手法について比較した。また、分類結果に基づいた作業チャート生成処理を実装し、作業時間を推定した。以下、これらの内容について報告する。

2. 作業画像分類システムの試作

2.1 対象作業の概要

本研究では、アルミダイカスト部品の検査工程を対象として作業画像进行分类する。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、約5×2×1[cm]の部品のバリの有無等を検査する(図1)。作業手順は、以下のとおりである。

- ①部品置き場から左手で部品を1個掴む。



図1 検査工程における作業の様子

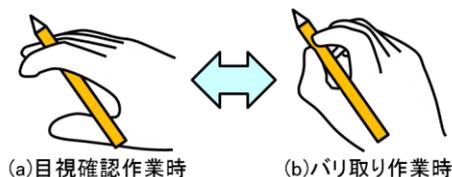


図2 バリ取り工具の把持状態

表1 作業映像の記録時間とフレーム数

| ファイル名 | 記録時間 | フレーム数 |
|--------|-------------|---------|
| Movie1 | 35:09.073 | 63,210 |
| Movie2 | 35:09.073 | 63,210 |
| Movie3 | 5:54.821 | 10,635 |
| Movie4 | 35:09.073 | 63,210 |
| Movie5 | 35:09.073 | 63,210 |
| Movie6 | 22:44.830 | 40,905 |
| Total | 2:49:15.943 | 304,380 |

- ②作業台上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する。
 - ③バリがなければ、既作業部品箱に右手で部品を放出し、①へ戻る。
 - ④バリがあれば、図2 (a) から (b) の状態に工具を持ち替え、バリを削り取り、②または③に戻る。
- これらを踏まえ、本研究で分類するクラスは、目視確認作業、バリ取り作業、その他の作業の3つとし、それぞれの作業を以下のとおり定義した。

目視確認作業：両手で部品を把持している状態（作業手順②）。

バリ取り作業：右手で把持しているバリ取り工具を持ち替え始めてから、バリを削り、再度両手で部品を把持するまでの状態（作業手順④）。

その他の作業：左手で部品を掴み、両手で把持するまでの状態（作業手順①）、左手から離れた部品が右手で放出され、作業台上に右手を戻すまでの状態（作業手順③）、及び部品の補充時や作業者の離席時など、目視確認作業、バリ取り作業以外の状態。

2. 2 作業画像の取得

作業画像は、1人の熟練作業員の手作業の様子を、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、動画ファイルとして取得した。カメラは、SONY製Action Cam (HDR-AS200V) を使用し、フレームサイズは1,920×1,080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。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表1に示す。なお、動画ファイルは、カメラの制約により、63,210フレーム目まで記録したところで自動的に分割され、新たなファイルに記録される。

2. 3 学習データの作成

学習データは、Movie1の全フレームについて、目視確認作業、バリ取り作業、その他の作業の3クラスに手作業で分



(a)目視確認作業

(b)バリ取り作業



(c)その他の作業

図3 各クラスの画像例

類し、順に28,011枚、21,601枚、13,598枚の計63,210枚の画像を学習データとして用意した。各クラスの画像例を図3に示す。

2. 4 試作システムの概要

試作システムは、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BoF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該当クラスに分類する。

BoFは、画像から局所特徴量を抽出し、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とにより特徴ベクトルを生成する画像特徴表現手法である。コードブックは、学習する全画像から抽出した局所特徴量をクラスタリングし、各クラスタの重心を要素（Visual Word）として生成する。ベクトル量子化ヒストグラムは、画像中の各局所特徴量について、最も近いVisual Wordを求め、対応するVisual Wordへの投票を繰り返すことにより生成する。特徴ベクトルは、ベクトル量子化ヒストグラムを局所特徴量の数で正規化することにより生成する。

識別器は、生成した特徴ベクトルを入力することにより構築する。識別器として用いるSVMは、様々なパターン認識問題に適用され、その有効性が確認されているが、SVMは、基本的には2クラス識別器である。そのため、多クラスに対して識別するには、複数のSVMを組み合わせた識別器を構成する必要がある。

試作システムにおける処理の流れを図4に示す。BoFおよびSVMの試作システムへの実装は、OpenCVをC#言語から使用するためのラッパーライブラリであるEmgu CV^[7]を用いた。なお、試作システムについては、処理速度の高速化を図るために、1,920×1,080のサイズで入力される各フレームを640×360にサイズダウンするとともに、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（注目領域、Region Of Interest）を設定し、グレース

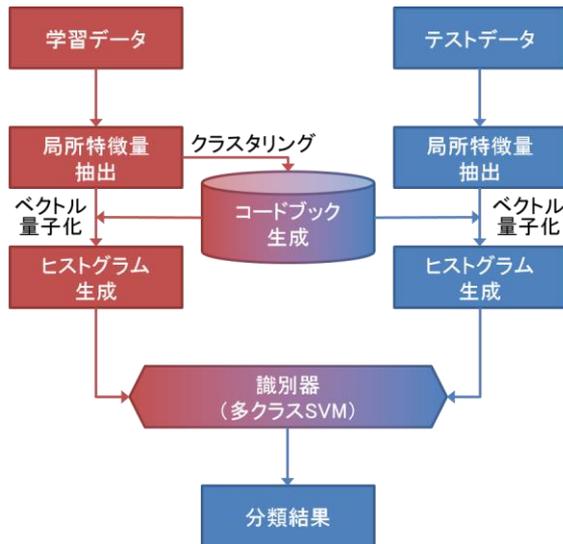


図4 画像分類処理の流れ

ケール画像に変換する前処理を実装している。

3. 局所特微量抽出手法の評価

3. 1 局所特微量抽出手法の概要

画像のスケール変化や回転に不変な特微量を抽出する SIFT^[8]や、SIFTの約10倍の高速化を実現したSURFは、BoFでよく用いられる局所特微量抽出手法である。

局所特微量抽出手法としては、SIFTやSURF以外にも様々な手法が提案されている。例えば、バイナリコードで特微量を生成することにより高速かつ省メモリ化を実現したBRIEF^[9]や、BRIEFにスケール不変性と回転不変性を導入したBRISK^[10]、ORB^[11]が提案されている。また、SIFTやSURFで使われているスケールスペースの等方性に着目し、非線形・非等方性のスケールスペースを利用したKAZE^[12]や、KAZEを高速化したAKAZE^[13]が提案されている。

他にも提案されている手法はあるが、本研究では、Emgu CVで利用可能な手法の中から、SIFT、SURF、BRISK、ORB、AKAZEの5種類について、BoFによる特徴抽出器を構築し、試作システムにおける認識率を評価する。

3. 2 局所特微量抽出手法の比較

実験では、SIFT、SURF、BRISK、ORB、AKAZEの5つの手法を用いて局所特微量を抽出し、Visual Word数を4,000として各コードブックを生成した。また、学習データは、目視確認作業、バリ取り作業、その他の作業の各クラスについて、1,000枚を2. 3節の学習データからランダムに選択し、各手法における多クラス識別器を構成した。評価にあたっては、Movie1からMovie6の全フレーム304,380枚をテストデータとして、学習、及びテストを10回繰り返し、試作システムにおける平均認識率を求めた。各手法について、局所特微量の抽出例を図5に、各クラスに対する平均認識

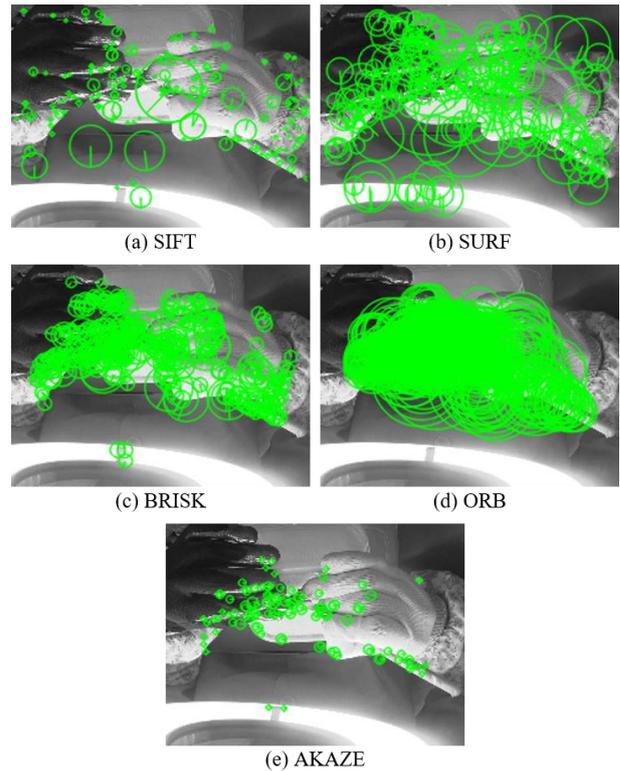


図5 各局所特微量の抽出例

表2 各局所特微量抽出手法における平均認識率[%]

| クラス | SIFT | SURF | BRISK | ORB | AKAZE |
|------|------|------|-------|------|-------|
| 目視確認 | 83.7 | 87.6 | 88.7 | 88.6 | 92.3 |
| バリ取り | 89.4 | 87.9 | 87.1 | 84.8 | 89.9 |
| その他 | 95.8 | 94.8 | 92.6 | 91.9 | 94.9 |
| 全作業 | 88.7 | 89.6 | 89.3 | 88.3 | 92.3 |

率を表2に示す。なお、多クラス識別器については、後述する一対他法を用いて構成した。

表より、全体の平均認識率は、ORB、SIFT、BRISK、SURF、AKAZEの順に高くなった。また、AKAZEを利用した場合は、昨年度利用したSURFと比べて、平均認識率が2.7ポイント向上した。

4. 多クラス識別器構成手法の評価

2クラス識別器から多クラス識別器を構成する手法としてよく用いられているのが一対一法や一対他法^[14]である。しかしながら、両手法ともに、識別結果が不定となる場合があり、不定となる割合が試作システムの認識性能に少なからず影響すると考えられる。そこで、不定となる割合を評価する予備実験を行い、不定となる割合を極力小さくする多クラス識別器の構成手法について検討した。

4. 1 一対一法

一対一法は、 k 個のクラスから2個のクラスを選択して kC_2 個の全ての組合せについて2クラス識別器を構築し、各識

表3 比較実験の結果

| 識別器 | 項目 | 確定 | 不定 | 全体 |
|------------|----------|------|------|------|
| 一対一 識別器 | 平均出現率[%] | 98.9 | 1.1 | 89.6 |
| | 平均認識率[%] | 90.1 | 40.5 | |
| 一対他 識別器 | 平均出現率[%] | 81.1 | 18.9 | 89.6 |
| | 平均認識率[%] | 94.6 | 68.3 | |

別器の出力結果を該当クラスへ投票することにより、最も投票数の多いクラスを識別結果とする方法である。最多投票数となるクラスが複数ある場合、識別結果は不定となるが、どのクラスに識別するかをランダムに選択したり、クラス番号が若いクラスに決定したりする等の方法が行われている。

4. 2 一対他法

一対他法は、 k 個の各クラスに対して、対象クラスかそれ以外かを出力する2クラス識別器を構築し、1個の識別器で正の値を、 $k-1$ 個の識別器で負の値を出力する時に、正の値を出力する識別器のクラスを識別結果とする方法である。複数のクラスで正の値となる場合や、全てのクラスで負の値となる場合、識別結果は不定となるが、超平面からの距離が最も大きくなるクラスを識別結果とする方法が一般的となっている。

4. 3 一対一法と一対他法との比較

比較実験では、BoFの局所特徴量にはSURFを利用し、Visual Word数を4,000としてコードブックを生成した。また、学習データは、3. 2節の実験と同様に、各クラスについて1,000枚をランダムに選択し、一対一法、及び一対他法による多クラス識別器（一対一識別器、及び一対他識別器）をそれぞれ構築した。テストデータは、Movie1からMovie6の全フレーム304,380枚を対象とした。なお、実験では、学習、及びテストを10回繰り返し、テストデータに対して全クラスの識別結果が確定/不定となるデータ数の割合（平均出現率）と、正しく識別されたデータ数の割合（平均認識率）を求めた。また、不定時の識別結果は、各識別器における超平面からの距離を求め、距離が最大となるクラスとした。

実験結果を表3に示す。一対一識別器については、不定となる割合が低いため、不定時の識別結果が全体の認識率に及ぼす影響が小さい。しかしながら、一対他識別器については、不定となる割合が高く、不定時の識別結果が全体の認識率に大きく影響していることが分かる。

4. 4 一対他法と一対一法の組合せ

予備実験の結果を踏まえ、本研究では、識別器の認識性能を向上させるために、確定時の認識率が高い一対他識別器に、不定の割合が低い一対一識別器を組み合わせた多クラス識別器（組合せ識別器）を構成する。本研究で対象とする3クラス分類における組合せ識別器のクラス判定基準（図6）を以下に述べる。

(ア) 一対他識別器において、1つのクラスで正、2つのクラスで負となる場合（1P&2N、図5中の領域(1)）に

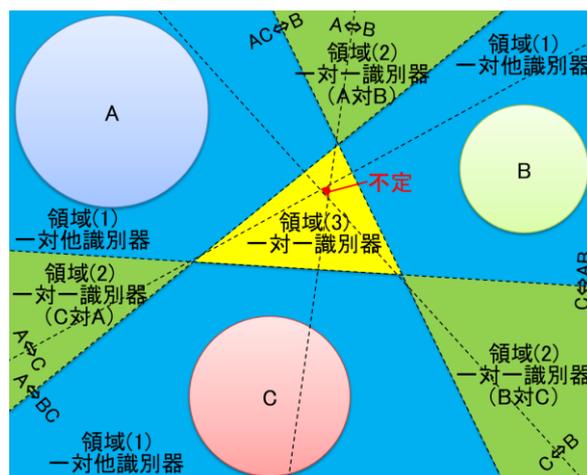


図6 組合せ識別器における判定基準

表4 組合せ識別器における各状態の平均出現率と平均認識率

| 項目 | 1P&2N | 2P&1N | 3P/3N | 不定 | 全体 |
|----------|-------|-------|-------|------|------|
| 平均出現率[%] | 81.1 | 4.3 | 14.0 | 0.6 | 90.7 |
| 平均認識率[%] | 94.6 | 76.6 | 74.6 | 48.4 | |

は、正となるクラスを分類結果とする。

- (イ) 一対他識別器において、2つのクラスで正、1つのクラスで負となる場合（2P&1N、図5中の領域(2)）には、正となる2クラス間の一対一識別器を用いてクラスを再判定し、正/負となるクラスを分類結果とする。
- (ウ) 一対他識別器において、3クラスともに正/負となる場合（3P/3N、図5中の領域(3)）には、全ての一対一識別器を用いてクラスを再判定し、最多投票数となるクラスを分類結果とする。
- (エ) (ウ)の一対一識別器において、最多投票数となるクラスが複数ある場合（不定）には、一対他識別器における個々の識別器の超平面からの距離を求め、距離が最大となるクラスを分類結果とする。

4. 5 組合せ識別器の評価

実験では、まず、3クラス識別に対する一対他識別器、一対一識別器を構成し、次に、前節の基準で分類結果を出力する組合せ識別器を構成することにより、全クラスに対する不定の出現率と認識率を評価した。なお、4. 3節の実験と同様に、SURFを用いて局所特徴量を抽出し、Visual Word数を4,000としてコードブックを生成した。また、学習データ、及びテストデータについても同様とし、学習、及びテストを10回繰り返した。表4に、組合せ識別器における1P&2N, 2P&1N, 3P/3N, 不定の各状態の平均出現率と平均認識率を示す。

表より、組合せ識別器は、表3の一対他識別器と比較して、不定となる割合が0.6%に低下したため、不定時の認識率が48.4%に低下しても、全体の認識率に及ぼす影響が小さくなった。また、一対他識別器のみで構成した識別器と比べて、組合せ識別器は、認識率が1.1ポイント向上した。

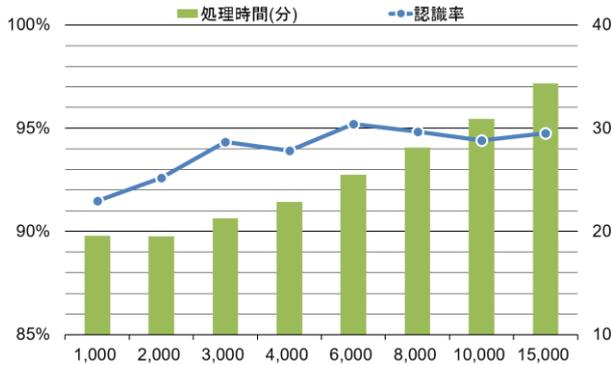


図7 Visual Word数の違い (横軸) による認識率と処理時間[分] (右側縦軸) の推移

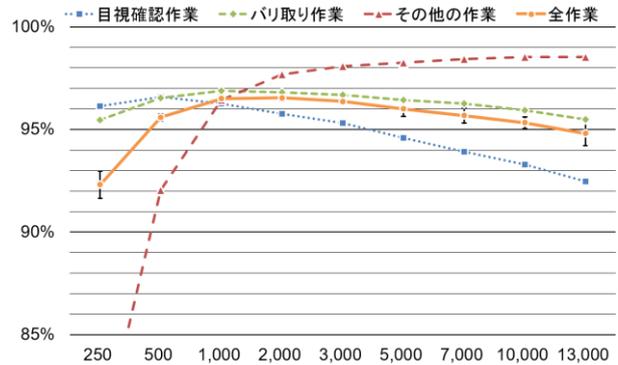


図8 学習サンプル数の違い (横軸) による各クラスの平均認識率の推移

5. 試作システムの評価

3, 4章の結果を踏まえ, 試作システムを改良するとともに, 昨年度¹⁶⁾と同様に, Visual Word数, 学習サンプル数を実験により選定し, 試作システムによる画像分類性能を評価した.

5. 1 Visual Word数の選定

BoFでは, Visual Word数が多いほど認識精度が高くなるのが知られている. 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. そこで, システムの試作にあたり, Visual Word数を選定するための実験を行った.

実験では, 2. 3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Movie1の全フレームに対する認識率と処理時間[分]の推移を調査した. 図7に結果を示す. なお, 局所特微量にはAKAZEを, 多クラスSVMには組合せ識別器を使用した.

図より, Visual Word数が6,000までは, Visual Word数が増加するにつれて, 認識率が高くなる傾向が見られたが, 6,000以降は, ほぼ横ばいとなった. 一方で, 処理時間については, Visual Word数の増加とともに, 長くなる傾向が見られたが, Movie1の再生時間よりは短くなった. そのため, 本システムにおけるVisual Word数は6,000とした.

5. 2 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, 2. 3節の学習データから各クラスについて同数のサンプルをランダムに選択し, サンプル数を変更した時のMovie1の全フレームに対する認識率の推移を調査した. なお, 実験では, 学習, 及びテストを10回繰り返し, 平均認識率を求めた. 各サンプル数における平均認識率の

表5 改良前の試作システムにおける各クラスの平均認識率[%]

| クラス | Movie1 | Movie2 | Movie3 | Movie4 | Movie5 | Movie6 | Total |
|------|--------|--------|--------|--------|--------|--------|-------|
| 目視確認 | 93.9 | 91.2 | 89.4 | 85.2 | 85.2 | 77.9 | 87.6 |
| バリ取り | 95.1 | 90.5 | 89.2 | 82.1 | 84.7 | 83.9 | 87.9 |
| その他 | 96.1 | 95.2 | 94.4 | 94.7 | 94.3 | 93.8 | 94.8 |
| 全作業 | 94.8 | 91.9 | 90.6 | 87.1 | 88.5 | 83.8 | 89.6 |

表6 改良後の試作システムにおける各クラスの平均認識率[%]

| クラス | Movie1 | Movie2 | Movie3 | Movie4 | Movie5 | Movie6 | Total |
|------|--------|--------|--------|--------|--------|--------|-------|
| 目視確認 | 96.4 | 95.4 | 94.4 | 92.9 | 94.6 | 88.6 | 94.0 |
| バリ取り | 96.9 | 93.3 | 91.1 | 89.3 | 90.2 | 87.6 | 91.9 |
| その他 | 96.2 | 95.3 | 94.4 | 93.6 | 94.3 | 93.0 | 94.5 |
| 全作業 | 96.5 | 94.7 | 93.2 | 92.1 | 93.4 | 89.3 | 93.5 |

推移を図8に示す.

図より, サンプル数が1,000の時, 全作業の平均認識率は96.5%となり, クラス間の平均認識率の差が最も小さくなった. また, その他の作業については, サンプル数が増加するに伴い, 認識率が高くなる傾向があるが, 目視確認作業, 及びバリ取り作業については, 低くなる傾向が見られた. これらの原因としては, 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率を低下させていると考えることができる. さらに, 全作業の平均認識率については, 標準偏差を示しているが, 値が小さいことから, ランダムにサンプルを選択しても, 認識性能に大きな影響は与えないと考えられる. これらのことから, 本システムにおける各クラスの学習サンプル数は1,000とした.

5. 3 画像分類手法の評価

これまでの実験結果より, 局所特微量をSURFからAKAZEに, 多クラスSVMを一対他識別器から組合せ識別器に変更し, Visual Word数を6,000, 学習サンプル数を1,000として試作システムを改良し, 画像分類実験を行った.

実験では, これまでの実験と同様に, 学習, 及びテストを10回繰り返し, 平均認識率を求めた. 改良前のシステムによる各クラスの平均認識率を表5に, 改良後の平均認識率を表6に示す.

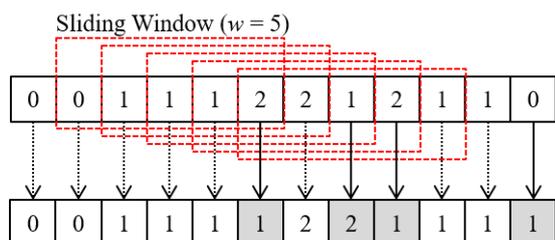


図9 フィルタリング処理例 (ウィンドウ幅 $w=5$ の場合)

両表より, その他の作業の平均認識率は若干低くなったものの, 目視確認作業, バリ取り作業については高くなり, 全作業としては認識率が3.9ポイント向上した.

6. 作業チャートの生成

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, 作業の時間的分布を表す. フレーム単位で処理すれば, 29分の1秒単位で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, 高い認識精度が必要となる.

本研究では, フレーム単位での作業チャートを生成するために, 画像分類処理後にフィルタリング処理を加えることにより, 認識性能の向上を試みた. なお, 作業チャートの生成にあたっては, フィルタリング処理後の分類結果について, 単位時間 T 内で最も多く分類されたクラスの作業を, 単位時間あたりの代表作業として選定した. 代表作業が複数となる場合は, 単位時間内の中央値とした.

6. 1 フィルタリング処理

フィルタリング処理では, フレーム F_i を中心に幅 w のウィンドウを設定し, ウィンドウを1フレームずつスライドさせながら, 各フレームにおける分類結果を選定するスライディングウィンドウ法を用いた. 選定にあたっては, ウィンドウ内の各フレームの分類結果に基づいて各クラスに投票し, 投票数が最大となるクラスをフレーム F_i の分類結果として選定した. 最大投票数となるクラスが複数ある場合は, 投票結果に対象フレームの元の分類結果が含まれていれば, 元の結果を選定し, 対象フレームの元の分類結果が含まれていなければ, 対象フレームの1フレーム前の分類結果を選定した. 図9にフィルタリング処理における分類結果の選定例を示す.

6. 2 生成実験

作業チャートの生成実験では, フィルタリング処理におけるウィンドウ幅を $w=1\sim 31$ の間で変更し, 5. 3節の実験結果について正しく分類された平均分類率を求めた. また, 単位時間については, $T=1/29$ (フレームレート), 0.1, 0.5, 1.0[秒]の4種類について変更し, 平均分類率を比較した. 単位時間毎にウィンドウ幅を変更した場合の全作業に対する平均分類率を図10に示す. なお, $w=1$ の時は, フィルタリング処理を行わずに, 代表作業を選定した時の平均

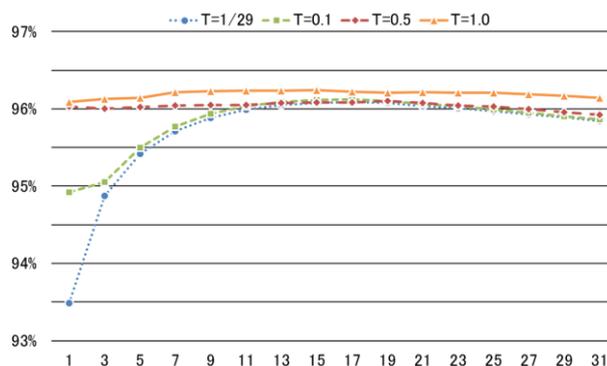


図10 各単位時間 T のウィンドウ幅 w の違い (横軸) による全作業に対する平均分類率の推移

表7 $T=1.0, w=15$ の場合の作業時間[秒]の推定結果 (括弧内の

数値は誤差)

| クラス | Movie1 | Movie2 | Movie3 | Movie4 | Movie5 | Movie6 | Total |
|------|------------------|------------------|-----------------|------------------|------------------|-----------------|--------------------|
| 目視確認 | 923.3 (17.3) | 949.4 (25.4) | 145.1 (11.1) | 937.9 (50.9) | 834.4 (43.4) | 564.8 (9.8) | 4354.9 (157.9) |
| バリ取り | 736.7 (-0.3) | 691.9 (-6.1) | 124.2 (-5.8) | 587.9 (-19.1) | 515.6 (-13.4) | 463.1 (-1.9) | 3119.4 (-46.6) |
| その他 | 450.0 (-17.0) | 468.7 (-19.3) | 85.7 (-5.3) | 584.2 (-31.8) | 760.0 (-30.0) | 337.1 (-7.9) | 2685.7 (-111.3) |
| 全作業 | 2110.0 | 2110.0 | 355.0 | 2110.0 | 2110.0 | 1365.0 | 10160.0 |

分類率を表し, $T=1/29$ の時は, フィルタリング処理のみの平均分類率を表す.

図より, $T=1/29, 0.1$ の時は, フィルタリング処理により平均分類率が向上していることが分かるが, 0.5, 1.0の時は, フィルタリング処理の効果はあまりないことが分かる. これらのことから, 代表作業の選定処理は, 単位時間が大きい場合, フィルタリング処理の効果を持っていると判断できる. そのため, 作業チャートの用途に応じて, フィルタリング処理の有無を切り替える等の運用方法が考えられる.

6. 3 作業時間の推定

前節で最も分類率が高かった $T=1.0, w=15$ の時の作業チャートから, 各作業の作業時間[秒]を推定した. 表7に推定結果を示す. なお, 値は平均値であり, 括弧内の数値は, 認識率を求める際の正解結果から算出した作業時間との誤差を表す. 誤差の絶対値の合計を作業時間の合計で除した平均絶対誤差率は3.1%であり, 概ね作業時間の推定が可能であることが分かる.

6. 4 作業チャートの可視化

作業チャートを可視化するために, 単位時間を1ピクセルとして時系列順で水平方向に描画し, タイムチャート形式で画像ファイルを生成した. 生成した作業チャートの一部を図11に示す.

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

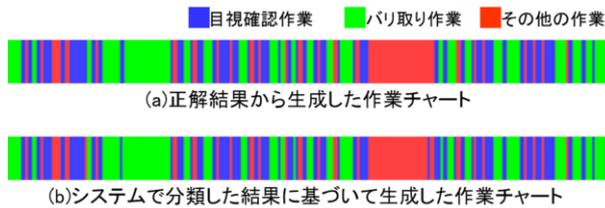


図11 作業チャートの可視化例（一部抜粋）

ることが可能である。

また、図11 (a) は正解結果から生成した作業チャート、(b) はシステムで分類した結果から生成した作業チャートの一部を示している。図より、誤分類はあるものの、概ね正確に再現できていることが分かる。

7. まとめ

本研究では、作業時間計測の支援を目指して構築している作業時間推定システムの認識精度を高めるために、BoFにおける局所特徴量の抽出手法、及びSVMによる多クラス識別器の構成手法について、他手法との比較実験を行い、最適なフレームワークについて検討した。その結果、局所特徴量抽出手法にはAKAZEを、多クラスSVM構成手法には一対他識別器と一対一識別器を組合せた識別器を使用する場合に、認識精度が最も高くなることが分かった。また、これらの手法をシステムに実装し、Visual Word数、学習サンプル数について最適なパラメータを得るための分類実験を行った結果、Visual Word数を6,000にした場合、各クラスについて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。これらの結果から、最適なフレームワークによるシステムを構築し、全動画ファイルに対する評価実験を行った結果、93.5%の認識率が得られた。

また、分類結果に基づいて作業チャートを生成する方法を検討し、フィルタリング処理におけるスライディングウィンドウ法のウィンドウ幅と、代表作業の選定処理における単位時間について、パラメータの変更による分類率への影響を調査した。その結果、単位時間を0.1秒以下とする場合には、フィルタリング処理を加えることにより、作業チャートの分類精度が高くなり、0.5秒以上では、フィルタリング処理の効果は低いことが分かった。推定した作業時間については、平均絶対誤差率が3.1%となり、本システムによる作業時間推定が有効であることが示唆された。また、作業チャートを可視化することにより、作業手順の確認が容易に行えることを確認した。

今後は、他の作業について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物体認識手法による作業分類の可能性を検証し、汎用性を高めていくことを検討する。

文献

- [1] 鈴木準, “IEを活用した現場改善のススメ”, ロジステイクス・ビジネス, Vol.9, pp.20-23, 2005.
- [2] 藤田彰久, “IEの基礎”, 建帛社, 1997.
- [3] 渡辺博己, 曾賀野健一, 棚橋英樹, “Bag-of-featuresによる検査工程の作業画像分類”, 動的画像処理実用化ワークショップ2016, OS1-3, 2016.
- [4] G. Csurka, C. Dance, L. Fan, J. Willamowski and C. Bray, “Visual Categorization with Bags of Keypoints”, 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”, Proc. of European Conference on Computer Vision, pp.404-417, 2006.
- [6] 渡辺博己, 曾賀野健一, 棚橋英樹, “画像認識技術を用いた検査工程の作業時間推定”, 岐阜県情報技術研究所研究報告, No.17, pp.24-29, 2016.
- [7] Emgu CV, <http://www.emgu.com/wiki/index.php>
- [8] D.G. Lowe, “Object Recognition from Local Scale-Invariant Features”, Proc. of International Conference on Computer Vision, pp.1150-1157, 1999.
- [9] M. Calonder, V. Lepetit, C. Strecha and P. Fua, “BRIEF: Binary Robust Independent Elementary Features”, Proc. of European Conference on Computer Vision, pp.778-792, 2010.
- [10] S. Leutenegger, M. Chli and R.Y. Siegwart, “BRISK: Binary Robust Invariant Scalable Keypoints”, Proc. of International Conference on Computer Vision, pp.2548-2555, 2011.
- [11] E. Rublee, V. Rabaud, K. Konolige and G. Bradski, “ORB: An Efficient Alternative to SIFT or SURF”, Proc. of International Conference on Computer Vision, pp.2564-2571, 2011.
- [12] P.F. Alcantarilla, A. Bartoli and A.J. Davison, “KAZE Features”, Proc. of European Conference on Computer Vision, pp.214-227, 2012.
- [13] P.F. Alcantarilla, J. Nuevo and A. Bartoli, “Fast Explicit Diffusion for Accelerated Features in Nonlinear Scale Spaces”, Proc. of British Machine Vision Conference, pp.13.1-13.11, 2013.
- [14] 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”, 信学技報, PRMU2009-121, pp.195-200, 2009.