4K1-2

英語センター試験を自動で解くための棒グラフの自動読み取り

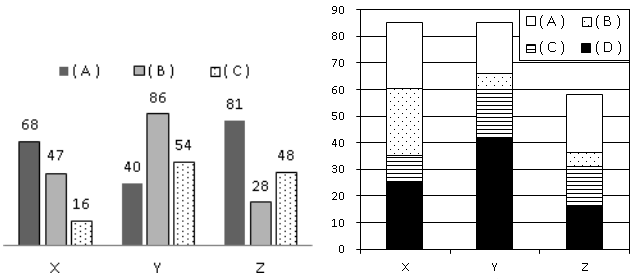
Automatic Reading of Bar-Graphs to Answer English Exams of National Center Test

中野　仁登 *1　　磯崎　秀樹 *2
Masato Nakano　　Hideki Isozaki

*1岡山県立大学大学院システム工学専攻
Okayama Prefectural University Graduate School of Computer Science and Systems Engineering

*2岡山県立大学情報工学部情報システム工学科
Okayama Prefectural University Faculty of Computer Science and System Engineering Department of Systems Engineering

We have been working on an AI project called "Can a Robot Get into the University of Tokyo?" Some English problems in the National Center Test for University Admission require recognition of graphs. Therefore, we are developing programs that can recognize different graphs such as (1) line graphs with marks, (2) line graphs without marks, (3) bar graphs, (4) pie charts. In this paper, we describe our program that can recognize complex bar graphs. Since the test is monochrome, bar graphs do not use colors but different filling patterns. We use OpenCV for processing images, and Tesseract-OCR for character recognition.



(a) 単純棒グラフ　　(b) 積み上げ棒グラフ

図 1: センター試験で用いられる縦棒グラフパターン

1. はじめに

「ロボットは東大に入れるか？」プロジェクトでは，センター試験問題を自動で解く人工知能の開発を行っている．センター試験の英語問題では例年，問 4 にグラフなどの図表を読み取り，答える問題が出題されている [東中 17]．リスニング問題でも図表を理解する必要のある問題が多く、リスニング問題の成績が上がらない主因となっている。

本稿ではそれらのグラフのうち，縦棒グラフの認識手法について述べる．表の読み取り [磯崎 15] と同様、今回も画像処理には OpenCV*1、文字認識には Tesseract-OCR*2 を用いた。

センター試験で用いられる棒グラフのパターンを図 1 に示す．使用されるグラフは (a) のような単純な棒グラフと（b) のような積み上げ型に分類できる．また，白黒であり，水玉や縞などのハッチングが用いられている．

棒グラフを認識する既存手法には，色を用いる手法 [Savva 11] やハフ変換を用いる手法 [Zhou 00] などがあるが，これらの手法はハッチングや積み上げ型グラフに対応していない．センター試験の棒グラフを認識するためには，ハッチングや積み上げグラフに対応した認識手法が必要である。

なお、グラフを読む処理は一般に chart recognition [Liu 13] や chart image recognition [Huang 04] と言われている。

連絡先: 磯崎　秀樹，isozaki@cse.oka-pu.ac.jp

∗1 http://opencv.org/

∗2 https://github.com/tesseract-ocr

2. 処理の概要

縦棒グラフ画像を入力として，構成要素を抽出し，問題解答に必要となる数値データを出力する．なお、入力画像はスキャン時に裏写りしていることがあるので，非常に薄い灰色を白に置き換えている．

抽出する構成要素は，凡例，棒，棒グループ名，数値文字列の 4 要素である．ここで「棒グループ」とは棒および複数の棒からなる組であり，「棒グループ名」は棒グループの下部に存在する文字列を示す．例えば，図 1(a) は X, Y, Z の 3 つの「棒グループ」から構成されている．「棒グループ」の下に書かれている「X」等が「棒グループ名」である．

2.1 グラフ領域の取得, 目盛線消去

はじめに，構成要素を抽出する上で邪魔になる目盛線を消去する．また，その過程で棒が描かれている範囲の特定を行う．棒が描かれている範囲をグラフ領域とよぶ．

まず，x 軸を抽出する．x 軸には全ての棒が接触している．よって，棒を縦直線ととらえると，x 軸は縦直線の端点が集中している横直線ととらえることができる．よって，縦直線を抽出し，その端点付近の横直線を調べることで x 軸を抽出できる．こうして得られた x 軸の長さを横幅，x 軸に接する縦直線を全て含む縦幅としてグラフ領域を決定する．

その後，目盛線を消去する．目盛線は，グラフ領域の端から端まで引かれた横直線である．よって，グラフ領域の左右端に接する横直線を見ることで，目盛線の y 座標が得られる。

目盛線は棒によって分断されるため，y 座標のみで位置を得ると，棒を構成する線が含まれてしまう．よって，棒を構成する線を取り除く処理を行う．目盛線は縦に一定間隔に配置されている．これは，「目盛線同士の間に一定の縦幅を持つ空白領域が存在する」と言い換えることができる．図 3 は空白領域を縦幅で分類した例である．図の塗りつぶし部が一定の縦幅，斜線部が異なる縦幅の空白領域である．この例のように，棒を構成する線は斜線部に接している．すなわち，接する空白領域の幅を調べることで，棒を構成する線を判別し，取り除ける．

残った部分を目盛線と見なして消去する．

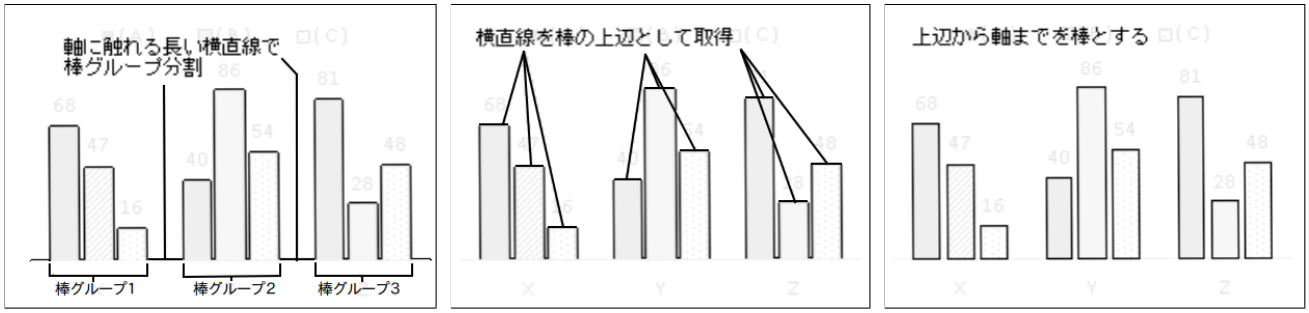


図 2: 棒および棒グループの抽出処理

2.2 凡例抽出

凡例領域では，矩形で表される凡例画像が縦横に整列され配置されている．よって，グラフ画像から矩形を抽出し，整列された矩形グループを検出することで凡例が抽出できる．

こうして得られた凡例画像を解析し，ハッチングの種類を分類する．ハッチングの多くは直線のみで構成される柄である．そこで、凡例画像内の直線をルールベースで調べることで、1) 縦縞，2) 横縞，3) 右下斜線，4) 左下斜線，5) ひし形，6) 格子，7) 塗りつぶし，8) 水玉の 8 種類に分類する。

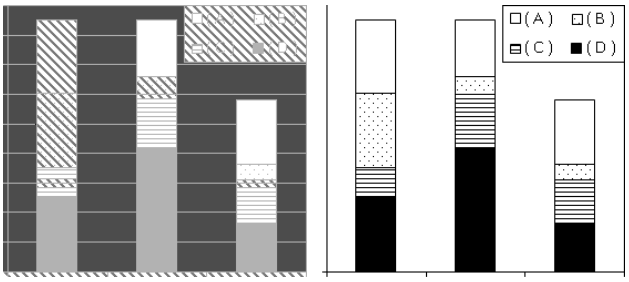
2.3 棒の抽出

グラフ領域内の構成要素の輪郭線を抽出し，棒および棒グループを抽出する．図 1(a) のグラフの輪郭線に対する検出処理の例を図 2 に示す．まず，棒グループを抽出する．抽出した輪郭線から，軸に接する長い横直線を検出し，その位置で輪郭線を分割する．こうして分割された一続きの輪郭線を 1 つの棒グループとする．次に，棒グループごとに棒を抽出する．輪郭線から十分な長さをもつ横直線を抽出し，これを棒の上辺とする．上辺から x 軸までの領域を棒とする．

その後，グラフの種類の判別する．全ての棒グループが棒 1 本で構成されている場合，積み上げ型，それ以外ならば単純棒グラフに分類する．

積み上げ型の場合，1 本の棒に複数の要素を持つため，棒を分割する必要がある．分割は隣接画素を比較することにより行う．棒内部の画素を一行抜き出すと，図 4 のように画素が一定のパターンを示す行と塗りつぶし行の 2 種がみられる．これら 2 種類に棒の各行を分類する．この分類を用いて棒を分割する．分割位置はパターン行と塗りつぶし行の境界および画素値の変化が大きい塗りつぶし行同士の境界とする．

この方法のみでは，横縞など横直線を有するハッチングにおいて，図 5 のように本来分割するべきでない位置で分割されてしまう．これらのハッチングを分割した場合，等間隔かつ狭い領域が連続して続いている状態になっている．このような領域を検出して，統合することで分割位置を修正できる．



(a) 空白領域の分類　(b) 目盛線消去結果

図 3: 目盛線消去処理の例

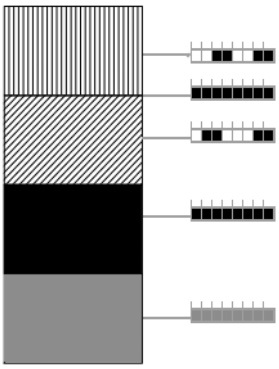


図 4: 棒の各行の画素パターン

表 1: 模試に含まれる棒グラフ読み取りの実験結果

棒グラフの種類	積み上げ棒	単純棒	合計
画像数	4	11	15
抽出すべきデータの数	94	140	234
プログラムが出力したデータの数	145	132	277
正解した数	59	117	176
再現率	0.628	0.836	0.752
適合率	0.407	0.886	0.635
F 値	0.494	0.860	0.689

次に検出した棒および分割領域を解析して，凡例と同様にハッチングの種類を分類をする．得られた分類の情報を凡例と比較することで凡例を棒に割り当てる．

2.4 数値，棒グループ名の抽出

Tesseract-OCR を用いて文字列を読み取り，縮尺と棒グループ名を得る．縮尺はグラフ領域の左側にある文字列を数値として読み取ることで得る．図 1(a) のような数値軸が存在しないグラフの場合，棒の上に存在する文字列を読み取ることにより縮尺を得る．棒グループ名はグラフ領域の下側にある文字列を読み，位置が近い棒に割り当てる．

3. 評価実験

センター試験模試の縦棒グラフ画像 15 枚を入力とし，その読み取り精度の評価を行った．実験結果を表 1 に示す．

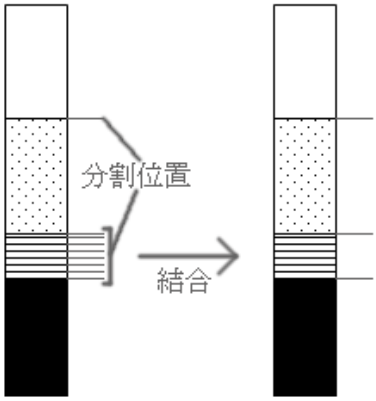


図 5: 横縞ハッチングを有する棒の分割

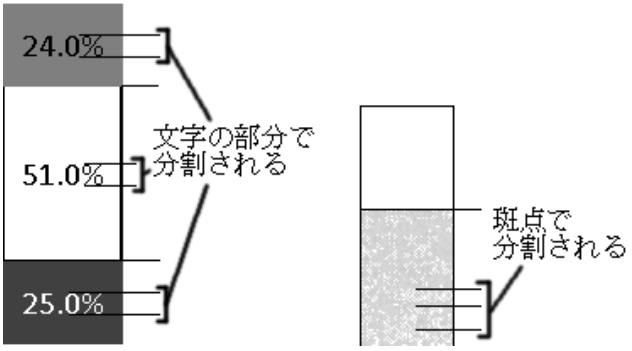


図 6: 積み上げ型グラフの分割失敗例

単純棒グラフでは，F 値が 0.860 と高い精度で読み取りを行うことができた．失敗例としては，棒の値が小さい場合，ハッチングが十分に描かれておらず，ハッチングの分類を正しく行うことができなかったことがあげられる．分類方法を見直すか，凡例割り当ての方法に工夫が必要である．

積み上げ棒グラフでは F 値が 0.494 と十分な精度で読み取りを行えていない．積み上げ棒グラフの失敗例は図 6 のような棒の中に数値が存在するグラフ，塗りつぶし領域に印刷むらが存在するグラフにおいて棒の分割が正しく行われなかったことである．内部の数値や印刷むらによって生じた不規則な白い斑点をパターン行として認識し，棒が分割されてしまっている．対策としては，分割処理時や領域の統合時に取得した凡例の情報を利用するといった方法が考えられる．

4. おわりに

センター試験に用いられる縦棒グラフの認識手法を提案し，その精度評価を行った．単純棒グラフは F 値 0.860 と高い精度で認識できたが，積み上げ棒グラフは F 値 0.494 と十分な精度で読み取りができなかった．

今後の課題としては，積み上げ棒グラフの分割方法の見直しがあげられる．また，今回使用した模試データに含まれず，センター本試験で使用されているグラフとして図 7 のような数値軸に省略があるグラフや折れ線グラフが描かれたグラフなどがあげられる．こうした特殊なグラフへの対応も課題である．

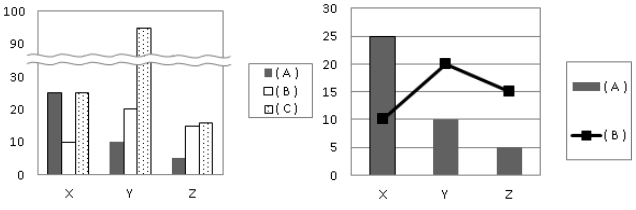


図 7: センター試験で用いられる特殊なグラフ

謝辞

本研究は NTT との共同研究である．本研究を推進するにあたって，大学入試センター試験問題のデータをご提供くださった独立行政法人大学入試センター及び株式会社ジェイシー教育研究所に感謝します。実験データをご提供くださいました学校法人高宮学園，株式会社ベネッセコーポレーションに感謝いたします．

参考文献

[Huang 04] Huang, W., Tan, C. L., and Leow, W. K.: Model-based Chart Image Recognition, in *GREC 2003, LNCS 3088*, pp. 87–99 (2004)

[Liu 13] Liu, Y., Lu, X., Qin, Y., Tang, Z., and Xu, J.: Review of Chart Recognition in Document Images, in *Visualization and Data Analysis* (2013)

[Savva 11] Savva, M., Kong, N., Chhajta, A., Fei-Fei, L., Agrawala, M., and Heer, J.: ReVision: Automated Classification, Analysis and Redesign of Chart Images, in *UIST'11* (2011)

[Zhou 00] Zhou, Y. P. and Tan, C. L.: Hough Technique for Bar Charts Detection and Recognition in Document Images, in *International Conference on Image Processing* (2000)

[磯崎 15] 磯崎 秀樹, 伊藤 圭汰, 荒木 良元：論文 QA のための画像処理〜表を読む〜, 言語処理学会年次大会 (2015)

[東中 17] 東中 竜一郎, 杉山 弘晃, 成松 宏美, 磯崎 秀樹, 菊井 玄一郎, 堂坂 浩二, 平 博順, 南 泰浩, 大和 淳司：「ロボットは東大に入れるか」プロジェクトにおける英語科目の到達点と今後の課題, 人工知能学会全国大会 (2017)