

プログラミングコンテスト 2012 におけるサイコロ検出問題の解法

新井 イスマイル* 柏木 まもる** 西村 友佑** 羽田 哲也**

A Dice Counting Solution for the 2012 Programming Contest

Ismail ARAI, Mamoru KASHIWAGI, Yusuke NISHIMURA, Tetsuya HADA.

ABSTRACT

At the competition sector of the 23th Programming Contest in 2012, the players competed in an accuracy measure for counting dice on a table. Despite the implementing of high technology features such as image processing, K-means clustering and an intuitive GUI interface, we lost at the quarterfinal due to an insufficient strategy. To lay the groundwork for future growth, we discuss the causes of victory and defeat of the players.

KEY WORDS: image processing, algorithm, programming contest

1. はじめに

プログラミングコンテスト（以下、プロコン）の競技部門は各高専から選抜された 1 チームが本選に出場するため、校内からエントリーがあれば毎年参加できる。プロコン公式サイトから確認できる過去 6 年を遡ってみると表 1 の通り、明石高専は一度も入賞していない。宇部高専が 3 回入賞しているので調査した結果、宇部高専にはコンピュータ部あり、毎年夏期休暇中に合宿を開催する等、組織立ってプロコンに挑んでいることが分かった。明石高専にも情報工学研究部があり、プロコンを意識して活動しているが、合宿をするほど活発ではなく、必ずしもこの部から本選出場チームが選抜されている訳ではない。校内で最も入賞の見込みがあるチームが参戦するのが望ましい一方で、長期的な PDCA サイクルを回せない構造が本校の戦績に表れていると言える。

本選に参加したチームは毎年「学校だより」（本校が発行する在校生の家庭向け広報誌）に参加報告を投稿するが、掲載されている 300 文字程度の文章には感想が述べられているだけで、今後のプロコンに

*電気情報工学科、**電気情報工学科学生

表 1 過去 6 年間のプロコン競技部門入賞校

回 (年)	1 位	2 位	3 位
23 (2012)	宇部	都立(荒川)	石川
22 (2011)	久留米	宇部	有明
21 (2010)	石川	松江	長野
20 (2009)	大阪府立	一関	釧路
19 (2008)	木更津	小山	舞鶴
18 (2007)	宇部	高知	広島商船

役立つノウハウは記載されていない。

過去の資料を調査した結果、宮本らによる第 16 回プロコン (2005 年) 競技部門への取り組みが詳述されている報告 [1] が確認された。このような文書を通してノウハウを校内に蓄積し、毎年度の反省点を次年度のチームに託すようなシステムを構築したい。その手始めとして、2012 年に開催された第 23 回プロコン (以下、プロコン 2012) 競技部門への取り組み結果を本論文にまとめる。

以降、2 章にプロコン 2012 競技部門の要項を、3 章に本校チームの選抜過程を、4 章に提案した解法を述べる。5 章に大会結果を述べ、6 章で本論文をまとめる。

2. 競技ルール

プロコン 2012 競技部門の題目は「数えなサイ ～ Here are Dice!～」であった。テーブル上に無造作に置かれたサイコロの山からコンピュータとデジカメ等の調査用機材を活用してその山に埋もれているサイコロの個数をできるだけ正確にしかも早く数えたチームが勝者となる。用語が沢山あるため、以下に一通り説明する。

- サイコロ
競技で使用するサイコロは以下の通り 3 種類あり、使用するサイコロの総数は公表されない。
 - 大サイズ (一辺 16mm 程度、重さ 5.6g 程度)
 - 中サイズ (一辺 10mm 程度、重さ 1.3g 程度)
 - 小サイズ (一辺 6mm 程度、重さ 0.3g 程度)
- ブース
プレイヤーが調査で得た情報を元に、コンピュータを用いて分析する場所。回答入力用 PC が設置されている。
- フィールド
競技ステージ上において、サイコロの調査を行う場所。山積みされたサイコロが置かれているテーブルが 2 か所 (フィールド A, フィールド O) ある。それぞれ、テーブルを囲むように調査エリアと進入禁止エリアがある。フィールドのイメージを図 1 に示す。
- 調査エリア
調査用機材を持ち込んで調査できる場所。図 1 に示すように、テーブルの中心から半径 1.5m～2m 程度の円内は進入禁止エリアで、テーブルを囲むように 120 度に区切られた調査エリアが 3 つある。進入禁止エリアは設置だけでなくライン上空を越えることも禁止されている。調査エリアには 2m の高さ制限があり、また一度に入れる人数は最大 2 名である。
- テーブル
天板上にサイコロが置かれている。高さ 70cm、直径 90cm 程度で縁にはサイコロ落下防止のために 3cm 程度の囲いがある。競技中のテーブルの様子を図 2 に示す。
- オブジェクト
数える対象ではないもの。サイコロに混ざってテーブル上に置いてある。図 2 は一例だが人型ロボットが置かれている。サイコロがこのオブジェクトの下敷きになって、どの方向から見ても見えないサイコロは無いものとする。サイコロ同士が重なっていたり、オブジェクト上にサイコロが乗

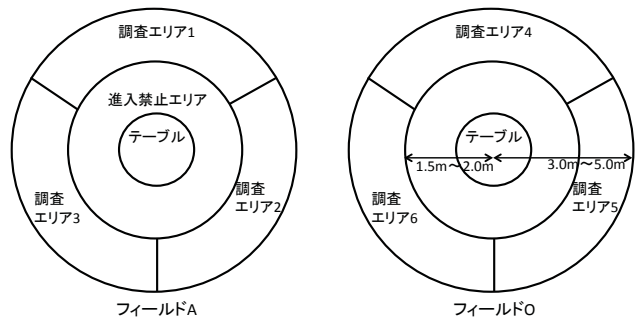


図 1 フィールドのイメージ (真上から)



図 2 競技中のテーブル

っていたりする場合がある。また、オブジェクトの表面は一様ではない模様や鏡面もある。

- 調査
フィールドの調査エリア内で行う行為。ライトや赤外線等の他チームの調査に影響を与えるような行為は禁止されている。
- 調査用機材
調査エリア内に持ち込んで調査に使用できる機器。調査エリア内に収まり携帯可能なものに限定され、ケーブルの引き回しはできない。また、競技に持ち込めるプログラマブルな調査用機材は 2 台までとなっている。

競技では 2 つのフィールド内のテーブルに山積みされたサイコロの個数を計算する。またテーブル上のサイコロの総重量が試合開始前に公開される。

進行については 1 調査エリアの調査制限時間は 1 分間となっている。1 競技に同時に参加するチームは 6 チームのため、1 分毎に各チームは調査エリアを移動し、合計 6 分の調査を行う。また、問題毎に分析制限時間が 1 分以上 3 分以下設けられている。したがって、1 競技の時間は 7 分以上 9 分以下となる。

順位はサイコロの種類それぞれの個数の正解誤差の合計が少ないチームが勝利する。これで勝敗がつかない場合は、フィールド毎のサイコロ総数の正解誤差の合計、2 つのフィールドのサイコロ総数の正解誤差の合計、回答提出時刻の早さ、じゃんけんの優先順位で勝敗が決まる。

尚、プロコン公式サイトに掲載されている第18回～第22回プロコン競技部門の課題を確認したところ、ここ数年は、与えられた問題をコンピュータ内で閉じて解くものが続いていた。人の動きが少なかったため、観客からは退屈な競技となりがちだったが、今回は人がサイコロの近くまで移動して調査を行い、またコンピュータに戻って結果を分析する等、人の動きがあったため、観客も楽しめ会場が盛り上がる競技となった。今後しばらくこの傾向が続くと予想される。

3. 本校チームの本戦出場過程と提案した解法

競技部門に参加できるのは1校につき1チームのため、本校では複数の応募があった場合には、本選応募切りの約1ヶ月前に校内予選を開催する。ほとんどの場合はプログラミング未着手なので、作戦をプレゼンテーションする。教員数名（情報工学を専門とする教員全員および学生委員）が聴講し、聴講した教員の多数決で応募チームを決定する。今回は2チームの応募があったため、校内予選を開催した。

第1チームは、互いの位置が固定された2台のカメラで写真を撮影して、エビポーラ幾何によって2つの写真内の対応点の3次元座標を再現し、テーブル上の3次元形状を点群で再現した後に、サイコロの大きさに合った個体を検出し、各サイコロの数をカウントするアルゴリズムを提案した。

第2チームは、1台のカメラでテーブルの周りを移動しながら動画を撮影し、各フレームでサイコロをパターンマッチングにより検出し、前後のフレームで認識したサイコロの位置関係を考慮して検出の重複を削除しつつ、画像内の奥行きの違いによるサイコロの大きさの変化が少ないことに着目して、検出したサイコロを分類する画像処理と、各サイコロの個数に上限を設定した上で、考え得る全てのサイコロの組み合わせに対する総重量のデータベース（以下、重さDB）を併用するアルゴリズムを提案した。

第1チームの手法は画像処理を用いたアプローチとしては王道ではあるが、精度向上のために沢山の写真を撮影して解析すると、点群の算出に時間がかかるため、競技時間内に計算結果が出ない恐れがあることと、サイコロの形状を認識するための十分な量の点群が得られるかどうか懸念された。第2チームの手法はサイコロの検出法について具体的な提案ができていない点に問題があったが、エビポーラ幾何に比べると計算量が少ないことが予想できた。また、事前に画像を観察して現実的な手法を提案す

るといった準備の姿勢が評価され、第2チームが選抜された。

4. 本校チームが提案した解法

2章で述べたルールは募集要項に記載されていた内容だが、熟読すると競技のルールで不明な点が見えてくる。そのような場合は、事務局に問い合わせると回答が得られ、特に共有すべきものはプロコン公式サイトのFAQ欄に公開される。特に重要だったFAQは以下のものだった。いずれも明らかになった時期は校内予選後である。

- サイコロの仕様
メーカー・製品名が公開された
- テーブル上に敷かれたカーペットの仕様
メーカー・製品名が公開された
- 人力による調査の可否
可能と回答された

本校では公表通りのサイコロとカーペットを購入した。サイコロを確認したところ、小サイズのサイコロは6mm角と小さく、目の部分は筆で点を描いたような、いびつな形だった。カーペットは図2の通り彩度の低いピンク色だった。最後のFAQは盲点だったが、大会名がプログラミングコンテストである以上、本番では目視でサイコロを数え上げるのが困難な量のサイコロがテーブル上にちりばめられるだろうと判断し、なるべくコンピュータ処理（主に画像処理）によってサイコロをカウントすることにした。

画像処理による物体検出は対象物の特徴を熟考し、その特徴を抽出するアルゴリズムを見出すのが一般的な流れである。使用するサイコロには以下の特徴がある。

- 目がある
 - 1の目は赤い
 - 1の目は他の目より大きい
 - 1以外の目は黒い
- 艶がある（反射光が生じる）
- 目以外の部分はクリーム色
- サイコロの角には丸みがある

本選で実践した手法は、以下の流れになった。画像処理専門用語が続出するが、オープンソースのライブラリ²⁾が存在するほど、当分野では一般的な処理である。本チームでもOpenCV2のライブラリ²⁾を多用した。

① 写真撮影

調査時間内になるべく多くの位置から写真を

撮影する。高所から撮影した方が全体を一度に捉えられるため、調査エリアの制限のほぼ限界の高さ(約2m)から撮影した。大会での本校チームの調査の様子を図3に示す。カメラが三脚の高い位置にありシャッターボタンを押すだけでもブレが生じるので、リモコン操作でシャッターを切る。

② 目の検出

②-① 反射光の除去

サイコロには艶があるため、反射光が目の誤検出要因になる。目の検出精度向上のために、目の中に映る反射光を除去する。ガウシアンフィルタによって平滑化後、グレースケールに変換し、オープニング処理する。尚、元画像の時点であるべく反射光を除去するために偏向フィルタをカメラのレンズに取付けた。

②-② 輪郭の検出

キャニーフィルタによるエッジ検出で得られた2値画像から輪郭の集合を得る。あくまで輪郭検出のため、目以外の輪郭も検出される。

②-③ 楕円の検出

サイコロの目自体は円であるが、写真中のサイコロは向きがそれぞれ異なるため楕円を検出して目を検出するのが賢明である。各輪郭に対して楕円フィッティングする。楕円の大きさ、他の楕円との距離、入力画像に対する楕円内の画素の情報を元に、目の集合を得る。

③ 面の検出

③-① 目のクラスタリング

2.3 から得られた目の集合からサイコロの面を検出するための下処理として、同一面上の目の部分集合に分割する。具体的には目の角度と目の大きさの情報を元に K-means 法でクラスタリングする。

③-② 面の検出

3.1 で生成した部分集合内の目が全てサイコロの目としてあり得る位置関係にある時、その部分集合を同一面内のサイコロの目の集合と見なす。具体的には、部分集合内の目のうち1つを選び、そこからの距離が閾値以下の目を全て選ぶ。選ばれた目を内包する最小矩形の重心と、選ばれた目の重心が同じであれば、選ばれた目はサイコロの面内のものとする。選ばれた目の数が1~6の場合にサイコロの面とみなす。1の目だけは赤いため、閾値以上の赤さの点があれば1の面とみなす。



図3 競技の様子

④ サイコロの検出

面の向きが目の向きから計算できるので、周りの面の向きと距離を元にサイコロを構成する面の組み合わせを求めてサイコロを検出する。また、面内の目の大きさを元にサイコロの種類を判定する。

⑤ 各サイコロの比率の導出

1~4の手法で誤検出あるいは検出漏れするものがあるので、人力で修正して各サイコロの比率を導出する。競技中には複数枚撮影するが、分析時間内に全ての写真に対して人力で修正することは困難なため、最も有力な写真を選定する。各画像で得られたサイコロ数の平均を求め、その平均値に最も近い写真を選定し、その写真内で検出したサイコロ上にマーカを重畳表示する。その後、マウス操作で誤検出や検出漏れを人力で修正する。

⑥ サイコロ数の推定

5で求めた各サイコロの比率および総重量に最も近い組み合わせを重さデータベースから取得し、各サイコロの数を推定する。

開発当初は学科既所有の2,000万画素の一眼レフカメラを利用していたが、目の検出率が低かったため、4,000万画素の一眼レフを購入した。その結果、検出率が向上した。とはいえ、検出漏れはあった。プログラミングコンテストの主旨を考慮すると人力は望ましくないと考えていたが、作成したプログラムでは完全なサイコロ検出が困難(特に小サイコロ)だったため開発終盤に上記5の処理を加えた。

また、大会前日までサイコロ検出アルゴリズムを調整していたため、他手法との比較評価はほとんどできなかった。また、フィールドを完全に再現した模擬試合を行うには至らず、数回の練習に留まった。

5. 大会の結果

競技部門の登録チームは61チームだった。全国の高専に加えて、ハノイ国家大学、成都東軟学院、モンゴル科学技術大学からもエントリーがあった。最大6チームで1試合が行われるため、第1回戦は11戦あった。試合毎に上位2チームが勝ち上がり、第2回戦が準々決勝となる。また、第1回戦後に敗者復活戦があり、その中の上位2チームも準々決勝にエントリーする。

本校チームは第1回戦で敗退して、敗者復活戦で勝利、その後、準々決勝で敗退した。準々決勝開始前に本校チームの調査担当がカメラを落とし、偏向フィルタにヒビが入った。代替機の解像度が低かったため十分なサイコロ検出ができず敗退に終わった。とはいえ、入賞チームの成績と比較すると、例え偏向フィルタが破損しなかったとしても、入賞は難しかったといえる。

全チームのシステム詳細は会場で配布されたパンフレットにA4半ページ程度で説明されている。これに加えて競技中の調査の様子を確認するとサイコロのカウント手法は概ね以下の3パターンに分類できた。

- 3次元解析

本校校内予選の第1チームと同様のアプローチ。同形式の一眼レフカメラを2台固定設置して撮影している様子が確認された。比較的早い段階で敗退していた。

- 2次元画像解析

本校チームと同様のアプローチである。複数の写真を取り終えた後にブースに戻り解析を行っている様子が確認された。離れた距離から撮影するため多くのチームがズームレンズを取付けた一眼レフカメラを用いていた。

- 人力

画像処理に頼らず調査時間内に可能な限り人力でサイコロの数を数え上げていた。あるチームはiPadで撮影後に調査エリア内に座り込んでタッチ操作でカウントをしていた。あるチームは、大中小のサイコロをカウントする際にノートPCのキーを押し分けて各サイコロをカウントしており、もはやプログラミング不要な手法だった。

サイコロのカウント手法は、上記の通り何種類かあるが、公開されるサイコロの総重量とサイコロ数の比率を元にサイコロ数を推定する手法は、パンフレットに書かれているシステム詳細説明と実際の試合を観察する限り、どのチームも同様だった。

優勝チームは表1の通り宇部高専だった。このチームと3位の石川高専のチームは人力でサイコロをカウントしていた。プログラミングコンテストの主旨として若干疑問が残るところではあったが、優勝

チームへのインタビューによると、画像処理によるカウントも試したところ、人力の性能を超えられなかったと回答していた。人力によるサイコロカウントでも、特に小サイコロは小さすぎて隠れたり見落とししたりする量も多くなるため、調査時間内に正解通りのサイコロの割合を出すことは難しい。彼らはその問題に対して何度も練習をして、正解と照らし合わせて各サイコロのカウント数に、経験的に得た係数を乗算していた。複数の手法を確認して十分な準備ができていたことが勝因だったと言える。

それと比べて本校チームは、1つの手法の精度を向上するまでギリギリまでプログラミングをしていて、他の手法を比較する余裕がなかったのが敗因と言える。人力によるサイコロカウントがルール上認められている以上、試すべきだった。せめて、模擬試合を沢山実施して、経験的にエラーの傾向を知り、精度の高い補正をするための係数を見出せば決勝あるいは準決勝までは勝ち上がったかもしれない。システムの高度さを競うものであれば優秀な戦績をあげられたかもしれないが、競技ルールはあくまでカウントしたサイコロの数の正確さ・速さといった結果を求めるものなので、カメラにトラブルがあったとはいえ、入賞は難しかったと思われる。

6. おわりに

2012年のプログラミングコンテスト競技部門はコンピュータ内に閉じない内容となり、高度なプログラムを作成したチームが上位に勝ち上がらない波乱の結果となった。複数の手法を比較評価できなかったことと、十分な回数の検証ができなかったことが本校チームの敗因であった。両者のいずれも工学研究では基本的なことである。最終学年で卒業研究に取り組む頃には気づけるだろうが、それよりも前の学年では一度でも失敗した経験がないと、気づくのは難しい。今後の競技者がこの論文を読んで失敗を疑似体験して、その次の状態から取り組むことができれば幸いである。

参考文献

- 1) 宮本行庸、渡邊俊介、橋本卓也、森洸介：“プログラミングコンテストにおける画像検出問題の解法”、明石高専紀要、第50号、9-13頁(2007)。
- 2) OpenCV 2 プログラミングブック製作チーム：“OpenCV 2 プログラミングブック”、マイナビ(2011)。