2023 Annual Conference & Exposition
Baltimore Convention Center, MD | June 25 - 28, 2023
The Harbor of Engineering Education for 130 Years
ASEE
Paper ID #40331

# Challenges in Designing Complex Engineering Problems to Meet ABET Outcome 1

**Dr. Bijan G Mobasseri, Villanova Univeristy**
**Ms. Liesl Klein, Purdue University at West Lafayette (PPI)**

Liesl Krause-Klein is a assistant teaching professor at Villanova University in their electrical and computer engineering department. She graduated from Purdue University's Polytechnic institute in 2022. Her research focused on student well-being. She is currently in charge of curriculum for capstone projects within her department.

**Mr. Edward Stephen Char Jr., Villanova University**

BS EE Villanova University 1996 MS EE Villanova University 1998

©American Society for Engineering Education, 2023

# Challenges in Designing Complex Engineering Problems to Meet ABET Outcome 1


## Abstract

ABET requires seven student outcomes to be met throughout a four-year curriculum for full accreditation. The first of these outcomes is related to identifying and solving complex engineering problems. As complex problem-solving requires higher orders of thinking along both knowledge and cognitive process dimensions, it is difficult to design adequate assessments for student outcome one in more beginning courses of a curriculum. Here, the authors discuss the definition and requirements for an assessment to fully evaluate ABET student outcome 1. Also discussed is the pedagogical background required for designing realistic engineering problems. Finally, an example project for sophomore-level electrical and computer engineers is explained in detail, with the author's own experiences in assigning this project explored. The project is an open-ended problem with multiple solution options. Students have scaffold-ed experiences within the course to guide them towards several possible techniques. Students follow a full problem-solving structure through defining their problem, exploring options, planning a method, implementing said method, and then reflecting upon the success of their design.


## Introduction

The first of the seven ABET outcomes is stated as "an ability to identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics."[1] It goes on to say that a "complex" engineering problem must meet one or more criteria such as having multiple solutions, no obvious or unique solution, include many sub problems, involve multiple disciplines etc. When designing assessments that accurately portray a student's ability to identify, formulate, and solve complex engineering problems, it is critical that instructors keep in mind the definition of complex engineering problems. As students are still gaining knowledge skills and cognition skills in earlier courses, it can be difficult to assess true complex engineering problems in lower-level engineering courses[2]. Within one private university's electrical engineering department, there has been discussion of how faculty are assessing ABET Student Outcome 1 and if their current assessments meet the definition set-forth by ABET. It was noted that many faculty may deem homework "problem sets" an assessment of Outcome 1, as students are being asked to solve problems. However, these problem sets are functionally a test for understanding of content and minor application of content to contained problems. Although there is abundant literature on how to assess Outcome 1, there is a noted lack of example assignments to build upon[3,4,5]. In this paper, we will share an example assignment that has been used in a sophomore-level MATLAB coding course for several semesters. This assignment can be adapted such that it is modality-agnostic, i.e. could be done in a different coding language or program. Another highlight of this assignment is it's allowance for growth in complexity. This assignment represents, as Perkins would say, a "small game" of image processing[6]. Depending on the rigor or level of the course, this assignment could be made more complex with different end goals for the processing.

This assignment is nicknamed "the skittles project" as, in the authors' iteration, the primary image used is an image of scattered skittles. The goal of the project is automated counting of the number of red, blue, green, and yellow skittles are in a bag. It did not say use image processing as the sole means to achieve it. As such, the approach did not forestall alternate approaches. However, it turns out that color is inherently a visual perception. There are no other cheaply available sensors that can measure color other than a camera. The output of the camera being a color image, naturally lends itself to using image processing tools. This choice does not negate the complexity of the problem. Complexity as defined in Section 1, still surfaces in abundance even within the selected approach.

Section 1 lists five requirements for a problem to qualify as "complex." Meeting any one of them is sufficient. It runs out, the approaches taken by students, meet at least four out of five requirements. Further, students had to answer five key questions listed in Appendix A that all in one form or another pertain to the complexity of the problem. specifically Appendix A.2.2. Students face conflicting technical issues and grapple with the lack of no obvious solution. As the skittles are randomly scattered, there is not a uniform size to any skittle, as there can be overlap, angles, and cutoff skittles. This means that no student group will count exactly the same number of any color skittle. Student counts will vary based on their approach, how they handle partial shape and orientation. This characteristic is the hallmark of complex problems where no unique solution exists and the "right" answer is elusive.

## 1 Assessing Complex Problems

ABET Student Outcome 1 relies on having a "complex problem" for students. ABET gives the following five qualifiers to define a "complex problem":[1]

1. involving wide-ranging or conflicting technical issues,
2. having no obvious solution,
3. addressing problems not encompassed by current standards and codes
4. involving diverse groups of stakeholders, including many component parts or sub-problems,
5. involving multiple disciplines, or having significant consequences in a range of contexts.

To fulfill ABET Student Outcome 1, students are required to identify, formulate, and solve complex engineering problems, but what makes a problem particularly "complex?" Classroom problems are often well-structure and well-defined, with a clear path to the solution. Complex problems, on the other hand, are often undefined with no clear answer, are iterative, and have multiple solutions. However, it is reported that engineering educators often have a poor grasp of complex engineering problems and subsequently fail to design such problems in their courses[7]. In

Table 1: Performance indicators and level of achievement

| Performance Indicator | Mastery(%) | Satisfactory(%) | Limited(%) |
|---|---|---|---|
| Preamble | 95.5 | 4.5 | 0 |
| Color Value | 81.8 | 13.6 | 4.5 |
| Skittle Size | 68.2 | 31.8 | 0 |
| Red Skittle | 95.5 | 4.5 | 0 |
| Blue Skittle | 95.5 | 4.5 | 0 |
| Yellow Skittle | 95.5 | 4.5 | 0 |
| Green Skittle | 95.5 | 4.5 | 0 |
| Problem Description | 92.9 | 7.1 | 0 |
| Solution Concepts | 63.6 | 31.8 | 9.1 |
| Concept Explanation | 68.2 | 31.8 | 0 |
| Solution Description | 64.3 | 35.7 | 0 |

a focus group of 12 engineering educators, only one understood complex engineering problems and their attributes. Additionally, the rest were not able to identify more than three attributes[7]. Senior engineering students do not fully understand or are unable to classify a problem as a complex engineering problem[8]. In a survey of seniors, while a majority believed they had experienced complex problem solving, only 21% could identify the required attributes of a complex problem[9].

Working with the revised Bloom's Taxonomy, ABET Student Outcome 1 refers to learning objectives that are higher order on both the cognitive and knowledge dimension[2]. Complex problems require metacognitive thinking, as well as analysis, evaluation, or creation from the student. Higher-order thinking is needed as there is a level of iteration that needs to occur within problem-based learning[10]. The McMaster Problem-Solving Structure states that students must define, explore, plan, implement, and reflect, with options for iteration in the case that the solution is unacceptable[10]. If an assignment does not require a student to go through these steps and use higher-order thinking, then it cannot be considered for evaluation of ABET Student Outcome 1. Within the example discussed in this paper, students are given clear guidelines of what is to be accomplished. They are then asked to define and explore the problem, as it relates to concepts learned in class. They must then plan their approach and implement it. In the example assignment, students are also asked to reflect on the accuracy of their project design and explain how their design works to a lay audience. Therefore, the example project follows the McMaster Problem-Solving Structure. Additionally, there is no one, correct solution to the project. Students can effectively solve the problem with multiple approaches and must decide on these approaches and the level of accuracy with which they can solve the problem. This requires higher level cognitive and knowledge skills on the revised Bloom's Taxonomy.

The details of the project appears in Appendix A. The following section is the assessment methodology and results.

## 2 Assessment Methodology and Results

This project was assigned in a sophomore coding course using Matlab. The course was offered in three sections totaling 72 students. The project was assigned in teams of three students. Team members were selected by random drawing. In previous iterations of the project, team membership was self-selected. Random assignment help foster diverse membership and generate new conversations among students.

A rubric was created to assess the completion of this project in context of the course goals and their alignment to aspects of ABET Student Outcome 1. Each category listed below was graded on a limited, satisfactory or mastery scale, worth 0%, 50% and 100% of the points respectively. The rubric was discussed with the students when the project was assigned, so that students were aware of the expectations they were held to. The rubric is shown below. The course-generated learning objectives for this assessment are:

1. Students will be able to distinguish between multiple colors in an image via matrix methods,
2. Students will be able to define sub-image sizes based on pixels within an image.
3. Students will be able to code a counting system based on said colors and sizes defined.
4. Students will be able to reflect on their experiences in such a way that an average seventh grader would understand.

Table 1 tabulates the performance indicators and the level of achievement as measured by student work, and is directly tied to the rubric in Table 2. As previously discussed, there are no “correct” solutions to this problem. Students are graded on their ability to develop a methodology to count the skittles and explain their decisions. A critical component of the decision process is reflection. In this assignment, students were asked to explain their process and the technical work of the Skittles project as if they were explaining it to a seventh grader. Students who fully understand the assignment and their methods can easily explain their work to a lay audience. Clear, concise explanations of their code are prioritized over the accuracy of Skittle counts.

Table 2: Assessment Rubric for the Report: Iteration

| **Criteria** | **Mastery** | **Satisfactory** | **Limited** | **Learning Objective Mapping** |
|---|---|---|---|---|
| Preamble | Students have included a "clear all; close all; clc;" statement in their header, along with their names, the datetime, and project description | Students are missing two or more parts of the header | No header | NA |
| Color Values | Students have determined the RGB values for the red, green, blue, and yellow skittles, and these values are approximately correct | Students have not correctly/accurately identified 50% of the skittle colors | Students could not identify more than 50% of the skittle colors | 1,3 |
| Skittle Size | Students have developed a way to determine the average size of a skittle in the image. Their method is clearly explained. | Students have identified the average skittle size, but the method in how they did so is not clear. | Students did not identify or set average skittle size. | 2,3 |
| Red Skittle | Students have generated a relationship between the R,G, and B values that is used to identify the number of red skittles. Their number of red skittles is similar to the true value. | Students have generated a relationship between the R,G, and B values that is used to identify the number of red skittles, but their number is not similar to the true value. | Students did not attempt to use color relationships to determine the number of red skittles. | 1,2,3 |

| | | | | |
|---|---|---|---|---|
| Blue Skittle | Students have generated a relationship between the R,G, and B values that is used to identify the number of blue skittles. Their number of blue skittles is similar to the true value. | Students have generated a relationship between the R,G, and B values that is used to identify the number of blue skittles, but their number is not similar to the true value. | Students did not attempt to use color relationships to determine the number of blue skittles. | 1,2,3 |
| Yellow Skittle | Students have generated a relationship between the R,G, and B values that is used to identify the number of yellow skittles. Their number of yellow skittles is similar to the true value. | Students have generated a relationship between the R,G, and B values that is used to identify the number of yellow skittles, but their number is not similar to the true value. | Students did not attempt to use color relationships to determine the number of yellow skittles. | 1,2,3 |
| Green Skittle | Students have generated a relationship between the R,G, and B values that is used to identify the number of green skittles. Their number of green skittles is similar to the true value. | Students have generated a relationship between the R,G, and B values that is used to identify the number of green skittles, but their number is not similar to the true value. | Students did not attempt to use color relationships to determine the number of green skittles. | 1,2,3 |
| Problem Description | Problem is clearly and concisely explained in simple terms. | Explanations could be more concise. | Students do not explain the problem concisely or in simple terms | 4 |
| Solution Concepts | Clear explanation of reasoning behind the proposed solution. | Explanation misses key features and properties of the solution | Explanations are difficult to understand. | 4 |

| | | | | |
|---|---|---|---|---|
| Concept Explanation | Students can clearly articulate why they are using their concepts in their solution. Their explanation is free of jargon. | Students can articulate why they are using their concepts in their solution, but their explanation is not clear. | Students are unable to explain why they are using these concepts in their solution. | 4 |
| Solution Description | Students are able to explain how they solved the problem and reference specific comments within their code. Their explanation is easy to understand based on the prior concept explanations and problem description. | Students are able to explain how they solved their problem, but do not reference specific code comments or prior definitions. | Students are unable to explain how they solved the problem. | 4 |

## 2.1 What does this mean for learning?

During the creation of this project, several tenants of teaching complex problems were utilized, including the Revised Bloom's Taxonomy and Problem-Based Learning techniques[2,4]. The rubric and assessment evidence was written to assess the higher levels of learning along the cognitive process and knowledge dimension of the Revised Bloom's Taxonomy. As an example, students needed to determine the size of a skittle for their project. Possible methods mentioned included counting pixels or using the image processing toolbox in Matlab to draw circles. It was up to the students to determine which method they wanted to use, and evaluate how effective said method was for identifying all skittles in the image. Effectiveness was challenged by Skittles that were not oriented flat or were cut-off on the sides of the image. It was observed that some student teams preferred to debate effectiveness before attempting either method, while others preferred to test both methods before determining which one was more effective. Even after selecting a method, students needed to iterate their code in order to test different size definitions within their method.

## 2.2 Mapping our rubric to Student Outcome 1

In this section, we will discuss the direct mapping of our rubric to that of the ABET Student Outcome 1 assessment criteria. The following sections show the mastery of the students based on the questions from section A.1.

Question 1 involves wide-ranging or conflicting technical issues. This is directly measured by the sections of the rubric that evaluated detecting the colors of the skittles and the size of an individual skittle in order to determine the number of each color. This is lines 2-7 in the rubric. The results from the three classes were very positive. The students were rated at 2.91 out of 3 in detecting the colors of the skittles, and 2.95 in determining the number of each skittle color.

Question 2 focuses on having no obvious solution: This is also determined by the giving a range for each color since depending on the lighting, the yellow and red skittles could blend. It is up to the student to decide how to handle skittles that are on the border since many are cut off, how they are to be counted. Also, no specific size for the skittle was mentioned since due to manufacturing, some skittles are larger or smaller than standard. As seen in Part 1, the students did very well in explaining how they determined a skittle size and counted them, with a 2.91 score from the rubric.

Question 3 focuses on addressing problems not encompassed by current standards or codes: Once again, no definition of a specific color was given nor was the size of the skittle determined ahead of time.

Question 4 dealt with involving diverse groups of stakeholders, including many component parts or sub-problems: Students not only have to solve the problem in a group, but they then have to explain their solutions to the faculty and to a pretend audience of 7th grade students. This is seen in the last 4 parts of the rubric. The students did well on these 4 parts, with scores of 2.93 for Problem Description, 2.54 for Solution Concepts, 2.68 for Concept Explanation and 2.64 for Solution Description.

Question 5 relates to involving multiple disciplines, or having significant consequences in a range of context. Clearly, the students have to show programming knowledge, but they also must show their mathematics competency in creating their algorithms. Additionally, in adding a second picture not directly related to the original, they have to be able to adapt their processes. The authors measure this with the last four entries in the rubric as well.

## 3 Conclusions

In conclusion, it is possible for assessments to be created at entry-level courses that explore students' ability to solve complex engineering problems per ABET Student Outcome 1. Instructors should be aware of what factors define a complex engineering problem as opposed to a simple problem set, and utilize knowledge of cognitive processes for their students in the assessment design. This paper provides an example problem that is accessible to students with some coding background and utilizes skill-sets taught in an entry-level programming course. While the examples shared here are not the only possible solutions to the Skittles project, they provide inspiration and guidance for instructors to use this in their own courses.

# A Appendix

## A.1 The Skittles Project

Within the context of the course, students were divided into teams of three and provided with the scattered skittles image. They were tasked with using Matlab to determine how to represent the four colors of Skittles in the picture. They, additionally, had to define the size of an individual Skittle. The ultimate goal was to use the two sub-tasks of color representation and skittle size to count the number of skittles of each color within the image. Students ended the project by running an un-related image through their code to evaluate how well their code could identify colors within a different context. Students reflected on their experience by writing a report about the problem-solving process, keeping in mind an average seventh grader as their audience. The following steps were specifically asked for:

1. What is the problem you are trying to solve?
2. What concepts you used to solve the problem?
3. Why you chose those concepts to solve the problem?
4. How did you iterate or refine your code?
5. How effectively did you solve the problem?

## A.2 The Skittles Project as a complex problem

This section will discuss various aspects of the skittle's problem with reference to aspects of complex problems, while also showing examples of solutions. The first two components of ABET's definition of a complex problem are explored in the context of this problem.

### A.2.1 Involving wide-ranging or conflicting technical issues

There are five colors in a bag of original Skittles; red, orange, yellow, green, and purple. Fig. 1 contains all, but the purple. Machine-assisted color separation of Skittles requires separation of colors in the RGB (red, green, blue) color space. The RGB color space is a 3-dimensional matrix that indicates a level of red, green, or blue on three separate pages. When mapping the Skittle colors to the RGB space, there are clear overlaps in what numbers within the matrix would represent the five colors within a Skittles bag. One tool to assess these overlaps, or adjacencies, is a color histogram. Each color is described by its own histogram but even pieces of the same Skittle do not have the same color histogram. This is caused by highlights, lighting and orientation of the Skittles. Fig. 1 shows the histograms for three different green Skittles. The distribution of red, green, and blue bands are similar but not the same. It is unlikely that any two students will be working with the same exact histogram of any one color. Decisions must be made to either use just one Skittle or use a histogram averaged over a sample of same colors. The latter choice lengthens training to produce a more representative color histogram, but it is not immediately clear if the impact on final accuracy is worth the trade-off. Also, notice the relationship among the three red, green, and blue histograms. Although there are minor shifts in the mean, the relationships among the bands of the same color persist. Conflict in the color

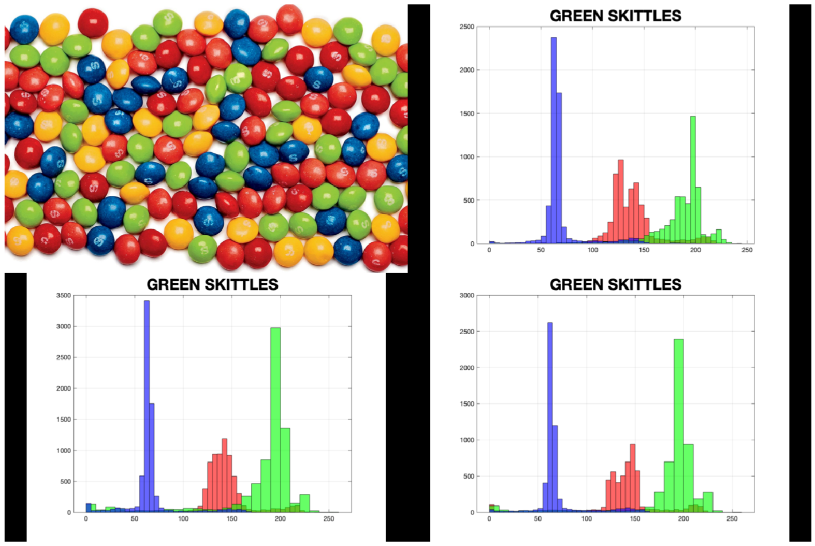


Figure 1: Color histograms of three different green pieces. Although all three histograms share structural similarities, they are nevertheless different. However, the RGB bands maintain similar relationships to each other, if not actual values.

definition through histograms adds complexity to the problem space that students must recognize and work through.

### A.2.2 Having no obvious solution

Choosing an optimal definition of each skittle color in the RGB space is crucial to the problem of counting the different colored skittls. Fig. 2 shows color histograms for four separate Skittle colors. As discussed in section A.2.1, there can be multiple "correct" color definitions based on where the student samples from. There are intuitive approaches, but there is no obvious solution. In all four colors, there is considerable overlap among the RGB layers. This is a clear indication of an open-ended problem with no obvious solution.

The use of different RGB color definitions will results in different outputs. All of these outputs are valid, despite the differences. Using the relationship between the R,G, & B values for each color, it is possible to create a "mask" to visually remove that color from the image. For example, by inspection of histograms, red skittles stand out from other colors by the following relationship among its three primary colors, `maskRed=[R>B&R>G]`. The application of this mask to the entire original image results in Fig. 3.

Using this rule, has resulted in lumping of the red, yellow and green skittles together. Students should be able to identify the lumping issue and recognize a need to refine their mask. Inspection of the red histogram shows that the separation of blue and green bands are minute and the red values are much greater. As a result, two more constraints are added to the mask.

```
maskRed=[R>B&R>G&abs(B-G)<2&R/G>2]
```

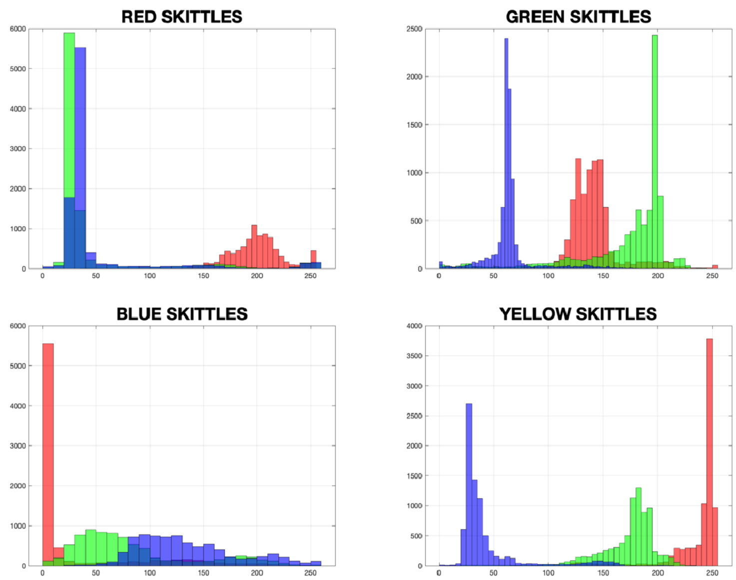


Figure 2: Color distribution in the RGB space is used to identify color pieces. What is important is not so much the actual intensities, but the relationship among color bands. For example, green and yellow Skittles have roughly the same blue values but the red and green are swapped.

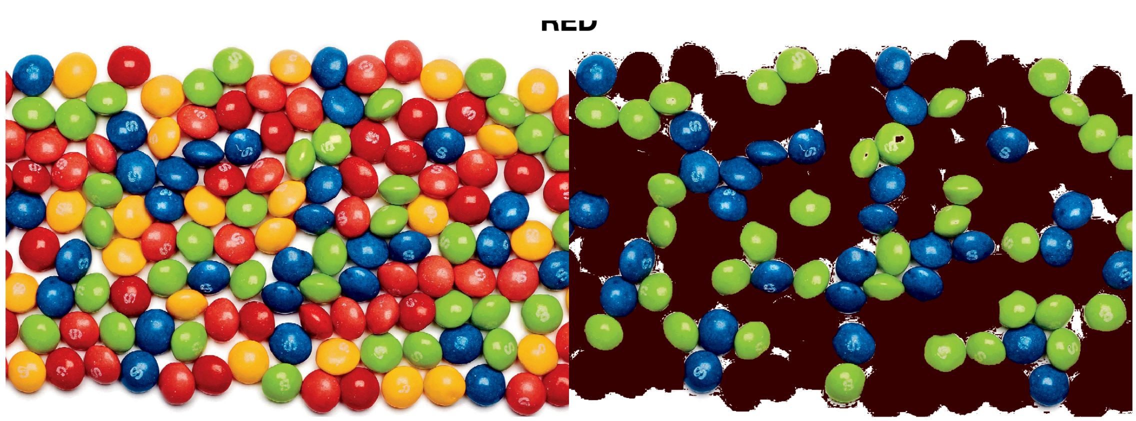

Figure 3: Poor separation of red Skittles using a simple rule. Red and yellow are merged. A different color mask is required.

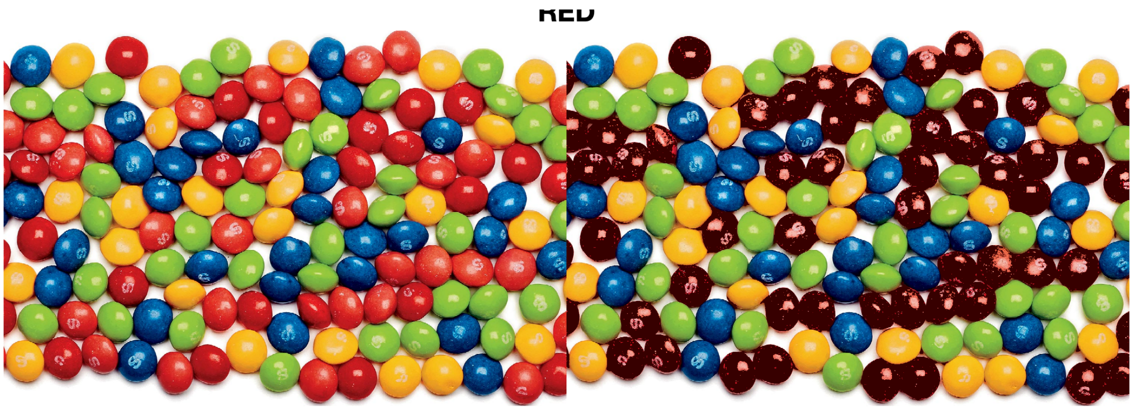

Figure 4: Separation of red Skittles from the rest is improved by adding one additional constraint to the mask.

In this example, an additional constraint is added to the original mask rule where the ratio of the red to green values is greater than 2:1. Comparison of Fig. 3 and Fig. 4 shows a clear improvement in separating reds from other colors. While this is not the only possible RGB color mask, one example mask that separates the skittle colors properly is:

$$MASK = \begin{cases} \texttt{maskR=[R>B\&R>G\&abs(B-G)<2\&R/G>2]} & \text{Red filter} \\ \texttt{maskY=[R>G\&R>B\&G>B\&abs(G-B)>50]} & \text{Yellow filter} \\ \texttt{maskG=[R<G\&G>B\&R>B]} & \text{Green filter} \\ \texttt{maskB=[R<B\&R<G\&R+G+B>95]} & \text{Blue filter} \end{cases} \tag{1}$$

Application of this mask to the whole image is shown in Fig. 5. Clockwise from the top left, the original image is subject to red, green, yellow and blue filters and the detected Skittles are recolored by computer code. Students may develop other masks that achieve similar, or better, results. There are also ways to write RGB rules without using histograms.

Just as students developed rules for identifying the different colors, they also needed rules to define the size of a skittle. One method used was the manually measure the size of several skittles in the image in pixels and average them. Given the average size in pixels, students could then identify the number of pixels of any color within the image. An estimated amount of skittles could then be determined by dividing skittle pixel size into the total pixel count of one color. There is a problem with this approach in that skittles appear in all shapes, size and orientations in the image. There are pieces that are imaged in perfect circles but they are a small proportion. To account for partial shape, a “fudge factor” need to be used to account for partial shapes. There is no right fudge factor. Knowing the actual count, some student groups used trial and error but most used one piece that they considered as average.

Another method is a shape-based approach that relies on fitting circles to the Skittles and counting the number of circle centers. There are many circle-finding algorithms, but they require some preprocessing of the data. For the Skittles project, a color separation step is still required. First,

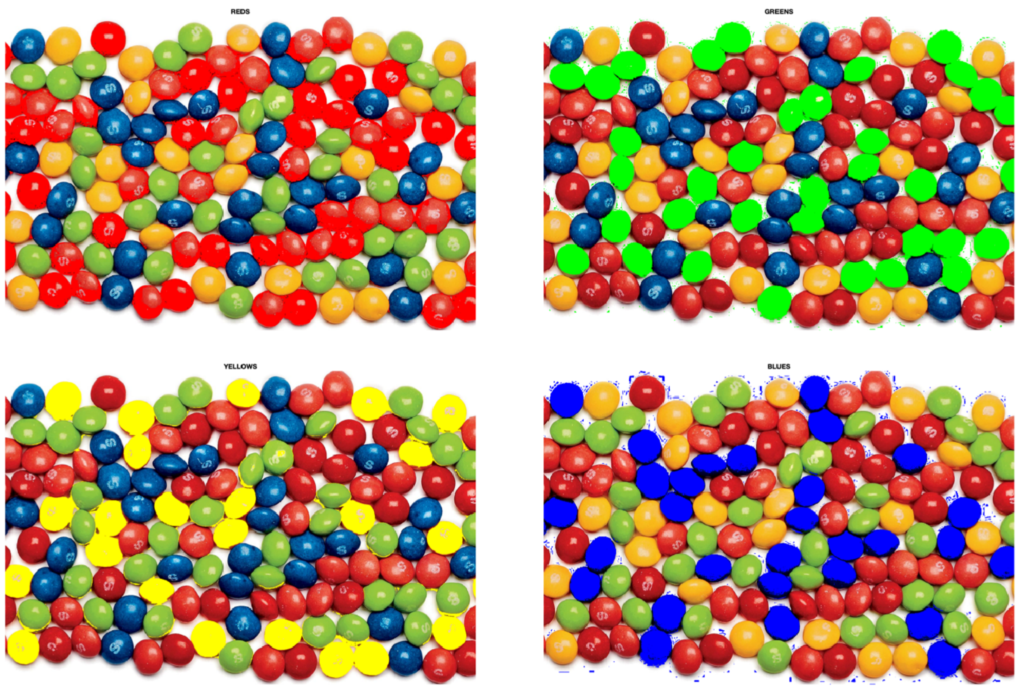


Figure 5: Application of four filters to the original Skittles figure using the rules in (1). Clockwise from top left, red, green, yellow and blue filters are repeatedly applied to the original image. Minor bleeding of colors through each filter can be seen.

| Color-based count | | | | Shape-based count | | | |
|---|---|---|---|---|---|---|---|
| Red | Green | Yellow | Blue | Red | Green | Yellow | Blue |
| 43/29 | 39/35 | 31/23 | 33/29 | 49/29 | 34/35 | 28/23 | 26/29 |

Table 3: Estimated count/true count pairs using two methods. The first method is based on color separation followed by pixel counts. The second method is shape-based by first fitting circles to specific colors. The output of the circle finder provides an estimate of the number of pieces of various colors.

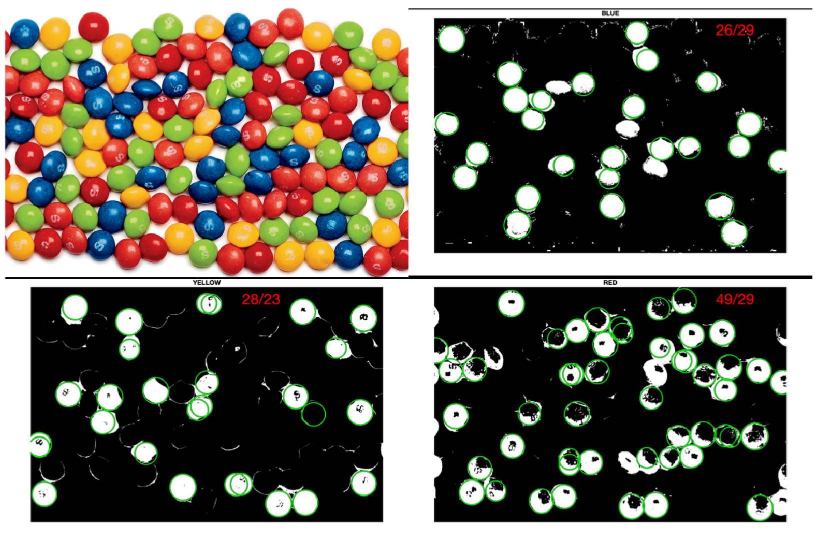


Figure 6: Counting color pieces by circle fitting. For each color the image is filtered then binarized to a black and white image. A circle fitting step identifies and counts the number of best fit circles. Due to fragmentation in the binary image, there are false as well as missed circles resulting in missed counts. Examples of both are evident in the images.

the image is run through each filter then binarized to a black and white image. A circle-finding algorithm is run over the image to generate Fig. 6. The algorithm outputs two vectors; circle radii and circle centers. The length of either vector is the total number of circles, or Skittles, from that image.

Table 3 compares the counts using both approaches, with the second number being the actual number when counting by hand. Overall, the two approaches show that a complex problem does not have an obvious, or a single, solution. Neither approach is fully accurate and can be continuously refined with color and size definitions. These approaches are also only two potential options. Additional solution paths may be employed.

## References


[1] ABET Engineering Accreditation Commission, "Criteria for Accrediting Engineering Programs, 2019-2020," ABET, Baltimore, MD, Tech. Rep., 2020. [Online]. Available: https://www.abet.org/accreditation/accreditation-criteria/

[2] D. R. Krathwohl, "Theory Into Practice A Revision of Bloom's Taxonomy: An Overview," 2010. [Online]. Available: https://www.tandfonline.com/action/journalInformation?journalCode=htip20

[3] B. Garry, "Examples of Rubrics Used to Assess ABET Student Outcomes in a Capstone Course," Tech. Rep.

[4] R. M. Felder and R. Brent, "Designing and Teaching Courses to Satisfy the ABET Engineering Criteria," *Journal of Engineering Education*, vol. 92, no. 1, pp. 7–25, 1 2003. [Online]. Available: https://onlinelibrary.wiley.com/doi/full/10.1002/j.2168-9830.2003.tb00734.x https://onlinelibrary.wiley.com/doi/abs/10.1002/j.2168-9830.2003.tb00734.x https://onlinelibrary.wiley.com/doi/10.1002/j.2168-9830.2003.tb00734.x

[5] J. G. Coello and K. Huggins, "The students outcomes ABET (1-7) and SOLO's taxonomy: An approach," in *ACM International Conference Proceeding Series*. Association for Computing Machinery, 6 2019, pp. 110–117.

[6] D. Perkins, *Making Learning Whole*. San Francisco: Jossey-Bass, 2009.

[7] C. P. Liew, S. H. Hamzah, M. Puteh, S. Mohammad, and W. H. W. Badaruzzaman, "A Systematic Approach to Implementing Complex Problem Solving in Engineering Curriculum," in *The Impact of the 4th Industrial Revolution on Engineering Education*, M. E. Auer, H. Hortsch, and P. Sethakul, Eds. Cham: Springer International Publishing, 2020, pp. 880–891.

[8] A. R. Fernando, J. G. U. Vergara, and C. A. D. Canlapan, "Work in Progress: Perception of complex engineering problem among capstone design students," in *2022 IEEE Global Engineering Education Conference (EDUCON)*, 2022, pp. 14–16.

[9] F. A. Phang, A. N. Anuar, A. A. Aziz, K. Mohd Yusof, S. A. H. Syed Hassan, and Y. Ahmad, "Perception of Complex Engineering Problem Solving Among Engineerng Educators," in *Engineering Education for a Smart Society*, M. E. Auer and K.-S. Kim, Eds. Cham: Springer International Publishing, 2018, pp. 215–224.

[10] R. M. Felder and R. Brent, *Teaching and Learning STEM: A Practical Guide*, 1st ed. San Francisco: Jossey-Bass, 2016.