An Algorithmic Approach for Modern Robotics Education & Practice

Surya P. N. Singh[1] and Hanna Kurniawati[2]

Abstract— **From self-driving cars to interactive warehouse arms, robotics is extensive field and a growing focus of study. This article discusses the application of a concise, systematic framework for teaching modern robotics. Here topics are presented as facets of overarching principles as compared to a collection (or toolbox) of methods. For example, we extend the traditional kinematic narrative of introductory robotics classes to one based on (affine) computational geometry and frames in space – be it for robot dynamics, perception, multi-view geometry, motion planning, controls, etc. A systems approach informs subsequent experimental practice allowing for analysis, synthesis, and integration of learned methods. We present a project design approach based on standardized, widely-available, flexible, (but not necessarily complicated or expensive) kit and objects. Project levels may be tiered on algorithmic complexity.**

The flexibility allows students to scale the learning to their ability and concentrate on facets of interest. From a learning perspective, this approach facilitates interdisciplinary practice with an algorithmic approach that is central to modern robotics. Evidence supporting this approach is gleamed from overall student outcomes, projects and evaluations from three years of mezzanine-level Robotics courses. We illustrate this via a coin-sorting and a cup-arranging problem. Beyond practice alone, by presenting a challenging (but manageable) research problem, we find that such tasks teach robotics in a principled and engaging way that lets students focus on learning generalizable methods over tacit technical details. While our approach is driven by the need for a compact framework for modern robotics education, it may also promote research foundations supporting later research opportunities.

I. Introduction

Robotics rightly captures the imagination. Modern robotics spans the gamut from manipulators to self-driving cars to semi-autonomous rescue robots. This fascinating subject captivates the imagination. From online courses to new graduate programs in robotics, interest in the subject has increased both popularly and academically. This, in turn, has renewed interest in introductory robotics courses, particularly at the mezzanine level.

From a learning perspective such courses offer an opportunity to introduce systems engineering concepts and to integrate knowledge across multiple disciplines and topics. From a teaching perspective, these courses attract highly motived and engaged students due to the general enthusiasm for the subject. While such excitement is helpful, the applied nature and general expectation of robotics often implies interest in new material and "modern" results. Compared to the significant attention paid to curriculum and learning development within particular subdisciplines, the design and emphasis for (introductory) robotics courses as a gestalt has received less attention [2], [3]. In part, this is attributable to the interdisciplinary and expanding nature of robotics, which has grown from articulated serial kinematics chains to mobile systems with integrated sensing and control. Both the breadth of material and the (relatively) short course periods suggest the need for a careful structuring of such courses [4].

Even experimental robotics spans the gamut from courses that use robotics to support course topics to the entire art of robotics. Thus, many robotics kits are focused on teaching closely related subjects such as programming [5], dynamics and controls [6], [7], and mechatronics [8] more so than to the principles of robotics. Hence there is a need for laboratory designs with a focus on algorithmic principles that enables students to navigate robotics research results and to apply these methods. We delineated the algorithmic areas in robotics using the Robotics and Automation Society's program structure as a guide. From this we identified algorithmic areas as motion planning, perception, kinematics, mapping, machine learning, control, and systems.

How to design such classes? On the surface, modern robotics is vast, spanning multiple disciplines. Often motivated by manipulator arms, the traditional emphasis in many of these courses is on kinematics modeling and state-space control (often for motion regulation). From a deep learning [9] and critical thinking [10] perspective, we establish a foundation that supports one being able to "research" a solution and go beyond rote learning. This may be seen as a lens for robotics coursework design.

We term an *Algorithmic Approach for Modern Robotics* (or AAMR). By "algorithmic" we mean a process to guide robotics analysis, design, and problem-solving operations. The process, detailed in Sec. II-A is that principle is introduced and that this to motivate related theory. The theory is reinforced via latter laboratory exercises (i.e., problem-based learning) with the smaller laboratories brought together, systematically, in the form of a challenge task; such as autonomously inserting a straw in an arbitrarily placed soda bottle.

This article dovetails previous work by the authors [1] published at IROS 2014 (Chicago). Where as the previous work focused on the coin-sorting problem as an exemplar, this article generalizes and considers the larger mezzanine-level introductory robotics course.

Systems shown in this article were submitted as part of an associated robotics course at The University of Queensland, METR 4202. http://robotics.itee.uq.edu.au/ metr4202

This work was supported by the Mechatronics Program and the Robotics Design Lab (RDL) at The University of Queensland (UQ), Brisbane, Queensland 4072, Australia

[1]Surya P. N. Singh (corresponding author) is RDL and Mechatronic Program director and coordinates METR 4202 (Advanced Controls and Robotics), `spns@uq.edu.au`

[2]Hanna Kurniawati is with the RDL and coordinates COMP 3702 (Artificial Intelligence), `hannakur@uq.edu.au`

Fig. 1. The Delta Cup Robot was designed as part of an autonomous cup arranging and filling project where cup and condiment locations are not known in advance. This involves several principles including object recognition, kinematics/control, and motion planning and compliance

We illustrate the approach via two candidate challenges, which have also been trailed in an introductory robotics course: (1) Arranging cups and filling them with (dry) objects, and (2) Sorting coins on a moving turntable. We show candidate solutions to these challenge problems: (1) the Delta Cup robot (pictured in Fig. 1) and (2) the CHARM (or the Coin Handling Arm for Robotics Mastery) robot [1].

In Sec. II-A, we detail the algorithmic course design strategy as a hybrid of both traditional didactic theory courses and problem-based learning for allowing coverage, depth, and interconnection between topics. While effective, we note that this requires careful design of the "challenges", a characteristic we term the "Goldilocks Problem" (Sec. II-B). The article concludes in Sec. V with some discussion of this as it relates to the broader perspective of mechatronics/robotics education along with some generalizations based on the outcomes of the platform.

II. Robotics Course Design

The multidisciplinary appeal of robotics is also a curse; it is a field whose topics originate from many disciplines (e.g., computer science, signal processing, mechatronics, etc.). These topics, when viewed as a collection of methods, this may seem overwhelming; however, subsets of them share foundations and core principles. For example, vision and motion planing of a serial manipulator are couched, in part, on computational geometry. This, in turn, provides a framework for structuring course concepts and material so that students may not only learn concepts more quickly (or conversely more concepts in similar time), but also so that they can integrate, instead of compartmentalize, theory from across the field and test their interplay via challenging laboratory exercises.

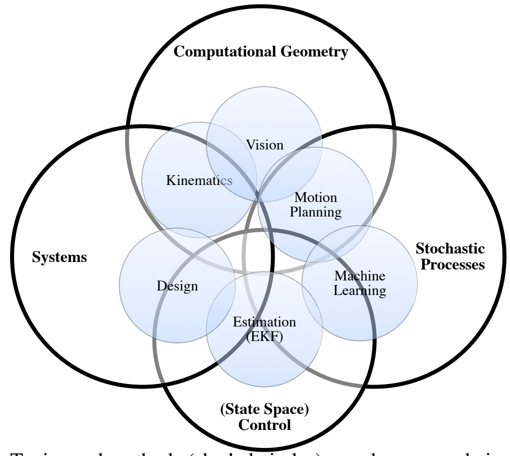

Fig. 2. Topics and methods (shaded circles) may be seen as being couched in overlaping principled areas (bold type). For example, parts of kinematics, vision, and motion planning may be seen as facets of computational geometry principles.

A. A Principles ⇒ Theory ⇒ Experiments Structure

Modern robotics is a rapidly developing field, which implies that field will be different in the not so future. Therefore, a robotics class must not only teach methods, but also fundamental concepts and necessary skills to understand and apply results from the state-of-the-art. Indeed, a metric for such a learning outcome would be students' ability to understand the digest of a major robotics conference (e.g., *IEEE's International Conference on Robotics and Automation (ICRA)*).

Bloom's taxonomy [10] suggests that the goal is to encourage "analysis", "evaluation," and "creation.". Based on this, we propose a "principles then theory then experiments" course structure. Topics are based on the major science themes as identified in the area and technical committee divisions that are used as part of the editorial process at major conferences with care to separate principles ("science") from methods (e.g., control, perception, learning, and planning) from application ("system") domains (e.g., medical and life sciences, industrial robotics and automation, field robotics, etc.). There is overlap between the sets (see also Fig. 2).

To achieve the above goal, learning should be focused on the ideas and underlying concepts behind the methods, rather than the methods per se. For example in motion planning, it is possible to use a bevy of methods to solve the problem of moving object - moving obstacle problem, including potential fields, rapidly exploring random tree (RRT) methods, etc. However, instead of only exposing students to such methods, we need to expose them to the underlying principles and ideas, such that students not only able to use these methods, but even more importantly, able to use the methods appropriately, selecting the right method for the right problem and understanding why the particular method is appropriate.

B. Goldilocks Problem

This approach shares the tenant of problem-based learning that practice is critical to understanding. The challenge in this is the design of the problem or task – if it is too "simple" then the best students are not challenged, whereas if it is too "complicated" then the academically weaker students struggle and, worse yet, may become despondent.

An algorithmic approach helps herein as within a principle class, various depths/complexities of methods may be explored. That is the problem may be adapted to be more challenging or rigorous by considering variations of (algorithmic) complexity. For instance:

- Adding a moving goal / environment – This adds the need to consider derivative states and to compute solutions in fixed time steps
- Adding 2.5D obstacles – This adds significant difficulty in perception, planning, and estimation.
- Introducing adversity to the system (e.g., an opponent that competes for the coins) – This would allow for the incorporation of game theory and/or AI strategies.
- Having a time optimal solution – This is still an open research problem.
- Removing structure – This removes constraints and/or structural mechanism that are often used to crutch a solution (e.g., fiducial markers, colored blocks, etc.)

C. Laboratories: Cheap and Cheerful

Deliberate practice has been noted as a mechanism for the acquisition of expert performance [11]. Using the aforementioned principal threads as a guide the initial laboratories/tutorials are structured such that each laboratory is centered around a principal concept (e.g., computational geometry) with latter (or final) laboratory systematically weaving the concepts togher to solve a "challenge task."

This invariably involves the design/selection of a kit and this has to be done in synchrony with the robotic learning problems; otherwise this will result in either robotics assignments that fail to cover all aspects in advanced robotics in a physical sense or a number of different assignments which require different setups for different problems. Designing a robot from scratch is typically beyond the scope of most robotics courses. That is, in practice, without a "platform," a considerable effort (in time and other resources) is spent building the system, leaving little time for the fundamental/algorithmic learning aspects of robotics. On the other hand, if the kit is too "established" (e.g., Nao) then the creativity is limited and a considerable amount of time and technical skill is required to learn the established system; again leaving little time to learn fundamental aspects.

From a systems point of view the approach is "cheap and cheerful" – we adopt standardized (size, weight, etc.) and widely distributed kit and objects that are easy to access at relatively low cost. There are many such robotics kits available such as Lego Mindstorms, Robotis's Bioloid/Dynamixel, Arduino-based kits, and the Engino Robotics Platform [12]. Such kits are flexible in scope (allowing for each team to have their own solution) while not so challenging as to have technical issues (such as interfacing, etc.) overwhelm the study of principles. Further As the Dynamixel hard-ware is well supported, but not overly prescriptive, students are open to explore new areas (in the case of CHARM, a more involved study of motion planning consisting of sorting coins around other coins on the plate).

We also extend this standardized object logic to the objects in focus in the laboratory. There are a bevy of such objects, but some examples include: coins, Lindor ball chocolates, Coca-Cola soda cans, etc. Such a model encourages practice (as the resource is widely-available), prevents confusion, and may be handy for distance learning and/or comparative cases.

D. Robotics to Support Related Studies

As a capstone subject, robotics integrates knowledge across multiple disciplines and topics. This highly positive and attractive characteristic makes it well suited to studying systems engineering. From a teaching perspective, robotics courses attract highly motived and engaged students due to the general enthusiasm for the subject.

Robots have been found effective in engaging and reinforcing student learning not only in robotics classes but also as a general learning tool to help students understand physical and mathematical concepts such as geometry and kinematics [8]. Therefore it has been an apparent choice for teaching robotics and mechatronics. In doing so, many instructors have used LEGO Mindstorm as the preferred robotics kit due to simplicity, reusability and ease of prototyping [13]. These kits have allowed instructors teach the concepts of direct and inverse kinematics and computation of simple arm trajectories. However limitations imposed by LEGO Mindstorm software and interface have also been reported. These limitations and the weakness of labs in teaching robotic algorithms such as vision and motion planning seem to correlate. Rosenblatt et *al.* [14] have reported, in designing lab assignments for a robotics course in Carnegie Mellon University, they used parts of LEGO Mindstorm kit in combination with their assorted sensors and controllers to let students come up with more creative solutions. However the labs do not provide a unified problem and therefore each requires different setup.

Other robotic kits such as Pendubot have allowed students implement theories in the labs on one setup, but due to the nature of the robots the topics become limited to specific areas such as control and systems [15]. On the other hand, mobile robots such as e-puc robot, Roomba and many others have tried to provide one platform for learning different robotics concepts in signal processing, control and distributed intelligent systems [5]. However, these applications do not form a well structured robotics problem and most of them do not support computer vision. Moreover, taking into account the number of kits needed for a (large) class, the workspace and expense may become limitations.

III. Illustrative Case Studies

A. The Coin Sorting Problem

The basic coin sorting problem is designed for mezzanine (i.e., upper undergraduate and beginning graduate) students. It is about moving coins to an appropriate bin, in the sense that all coins in the same bin must be of the same type. We add a twist to the problem to make it more interesting: the coins are placed on a rotating table (to which there can also be other objects/obstacles placed around it). Let us assume we have n coins of k different types. Suppose each coin is placed on top of a turning table that rotates with an angular velocity (which is typically, but not necessarily, constant) and suppose k empty bins are placed right outside of the turning table. The goal is to move each coin from its initial position to one of the bins, such that all coins inside a bin belong to the same type. The problem ends when no more coins are on the rotating table with all bins having the same type of coins.

This coin sorting problem spurs study in multiple areas:

1) **Hardware design.** At the very least, students must be able to design and build a simple 3-DOF arm that can push a coin. In addition, the basic coin sorting problem teaches the trade-off between hardware design and algorithmic difficulty. For instance, to solve the problem, one can build an arbitrary shaker, which is easy to built. However, such a shaker would make the process error prone and stochastic, especially when the shaker is controlled by imprecise, saturated motors (as is sometimes the case with low-cost motors).
2) **Perception.** The system must be able to identify, differentiate, and tracks multiple moving objects. While coins might be a standard object, variations in lighting and coin wear/quality additionally highlight (and teach) the challenges in developing robust sensing techniques. Since the types of coins can be increased easily, the coin sorting problem also teaches the challenges in scaling various vision and/or object classification techniques.
3) **Motion Planning.** The motion planning component of the coin sorting problem can be defined more generally as follows. Given n closed and planar objects on a 2D Euclidean plane, where all objects orbit with the same constant angular velocity around a common barycenter. Let $g_i \subseteq \mathbb{R}^2$ be the goal region of object-i for $i \in [1, n]$, and let each goal region acts as a sink, in the sense no object can leave the region. Suppose the initial position of each object is known and is within a finite distance from the barycenter. Then, we want to find a continuous path that moves each object-i from its initial position to its goal region g_i ($i \in [1, n]$) without colliding with any other object. This problem is not just complex, it is at least PSPACE hard, as the problem of motion planning with movable obstacles where the obstacles must end at a pre-specified location – a PSPACE hard problem – is a special case of the above problem [16].
4) **Estimation.** The use of low-cost motors require students to be able to estimate and filter out errors when controlling the robot. Furthermore, to perform well, students must also estimate the angular speed of the turning table. Although this estimation can be done easily by placing markers on the turn table, the ability to identify the need for such estimation and the use of such markers are skills that could come handy in developing more complicated robotics system.

The aforementioned list is not fixed. Depending on the systems design, some problems may be more prominent than others, and additional problems may occur. Furthermore, aside from varying levels of difficulty in each sub-field, the coin sorting problem also provides the opportunity to study the interplay between various aforementioned components, such as between perception and motion-planning. That is, for example, certain camera (sensor) placements may reduce occlusions (simplifying perception), but may not allow as encompassing a view (complicating planning/control).

The coin sorting problem can be extended or simplified easily to better cater to various student and/or learning requirements. For instance, it could be simplified by having a stationary table and direct sensing of the target and its location (e.g., augmenting the coin/object with a magnet for easier detection and/or to incorporate switches, encoders, or inductive sensors under the table to simplify object localization). Control performance can be simplified, for example, by providing larger bins (relative the the target size).

The problem of autonomously sorting moving coins (see Fig. 3) addresses the above impetus in a step-wise manner that allows teams of students to study these principles at increasing levels of complexity so as to match their level of learning and engagement [17], [18]. While coin sorting mechanisms abound, the sub-tasks involved present several challenges including vision [19] and control [20], especially for autonomous operations. More generally, the coin sorting task has subproblems that involve each of the aforementioned algorithmic areas – end-effector position placement, velocity control (Jacobeans), object recognition, obstacle avoidance, system identification, autonomous operation, and underactuated (saturated) controls.

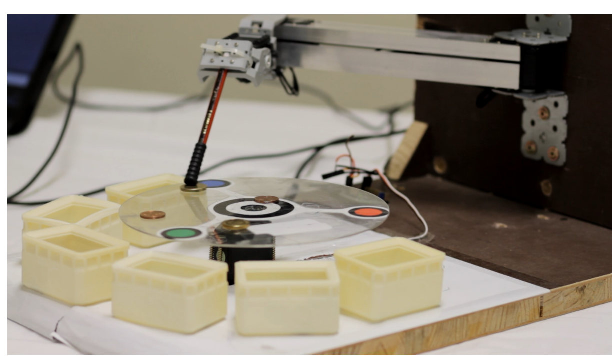

Fig. 3. The CHARM robot autonomously sorts coins randomly placed on its turnable. The robot involves several algorithmic principles including object recognition, kinematics/control, and motion planning in dynamic environments

B. The Cup Stacking & Filling Problem

The cup stacking and filling challenge involves filling a cup (with dry condiments for the sake of convenience and the safety of not dealing with fluids) autonomously. The initial location of the the cups is not known, operations are fully autonomous. For added complexity, different cup sizes are added to the scene (and have to be factored as part of the "orders" completed). As with the coin sorting problem, the cups may be placed on a moving turntable.

An example solution to this problem is the Delta Cup Robot (shown in Fig. 1).

IV. Broader Outcomes & Program Impacts

The structure architects an associated introductory, mezzanine-level (fourth-year undergraduate plus first-year Masters students), robotics course at The University of Queensland, METR 4202. This course has been taught annually using this approach for four times (from Semester 2, 2012 to Semester 2, 2015) and has been evaluated over these years.

The course starts with homogeneous coordinates/affine geometry (and motivates this in the context of kinematics and basic vision tasks); it then presents statistics principles (which underpin a discussion of computer vision "feature detectors" (e.g. SIFT)); it follows this with computational geometry and linear dynamical systems (leading to methods in motion planning and state-space controls). Over the course of a semester, these concepts are brought together.

As demonstrated by the CHARM robot (Fig. 3), the coin sorting "challenge" task integrates several aspects. Clearly such tasks require resources in the form of teaching staff time, tutorial assistance, apparatus and facilities. The experience with METR 4202, as an example, shows that these are similar to traditional laboratory based courses. In particular, there were more demands on teaching staff and tutor time (because of the diversity of tasks involved and the requisite time needed to assist and assess these tasks), but the apparatus was similar in cost/scope.

The course and project were assessed using course feedback surveys. Compared to the previous year the course was administered, the course showed a slight increase in how well the materials helped them with the learning (from 75% of students in agreement to 91% of students in agreement). Overall course marks are also high (4.81/5, with 100% of students rating the course as satisfactory, the strongest response for a class in the Mechatronics program).

The approach also is indicated in class results (Fig. 4). In METR 4202, the laboratory marks are tiered based on the level of depth of principal and methodological areas. That is, in the marking scheme a team will receive more marks if they can integrate tasks together (e.g., 10 for A or B, but 25 marks for A and B together). Thus, at some level of approximation, higher marks indicates that teams are figuring out how to integrate multiple methods. While a variable metric, the results for 2012 and 2013 show that teams are motivated to take on stronger challenges (such as motion planning in space and time).

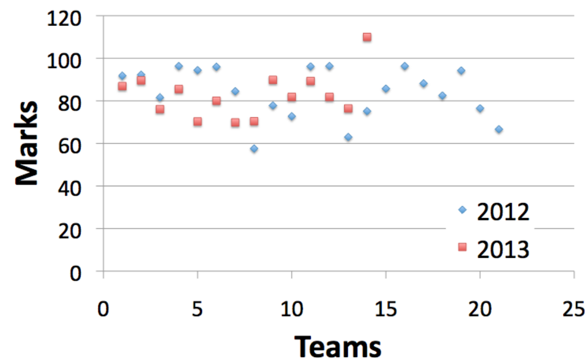

Fig. 4. Team performance in METR 4202 with an Algorithmic Structure. Teams are anonymized by a random number with scores based on a principled rubric where by every passing letter grade represents double the understanding (A [85-100] is roughly double a B [75-85] and four times a C [65-75]). The spike in 2013 (110) indicated that the students are very motivated by the problem and took the project to a new heights well beyond the rubric and also indicates that this approach encourage learning over diverse students' capability.

V. Conclusion

The evolution of robotics and automation has resulted in the growing significance of training and educating the future engineers, researchers and technicians with a good grasp of mechatronic systems. Many of the topics in this area can be mastered through learning by doing. Robots are naturally one of the main tools in teaching robotics and mechatronics and they play an increasingly important role in education from high school to postgraduate studies. Their application does not stop at robotics education but also extends to other areas such as mathematics and physics. For this reason development of suitable robotic kits can prove to be very useful for students interested in robotics.

An algorithmic approach to robotics education fosters critical thinking and creativity. It is not just about solving the immediate problem at hand, but allowing the student(s) to challenge themselves and explore facets in further dept and detail. That is part of the approach is to delineate science from technical methods. While tacit technical skills, such as Lego/Dyanmixel programming interfaces, will change quickly; the principles are long-lasting. It is also compact as it focuses on foundations (the tree) from which the vast field (the leaves) can be understood to stem from.

By framing the discipline and encouraging teaching independent study/implementation of particular concepts, it also supports latter independent learning and research goals. We also show that extensive learning need not require complicated or expensive apparatus, indeed the simpler kits may be better as they are faster to master and receive feedback from, fostering deliberative feedback.

Finally, to make an analogy to cooking, the goal is to teach the theory of flavor and have go though a couple of items so that they can make their own course in the end. While a collection of 1000 recipes is tempting, it is can be myopic as students often learn to memorize tacit details and can not adapt to new conditions.

REFERENCES

[1] S. P. N. Singh, H. Kurniawati, K. S. Naveh, J. Song, and T. Zastrow, "CHARM: A platform for algorithmic robotics education & research," in *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, 2014, pp. 2602–2607.

[2] J. Weinberg and X. Yu, "Robotics in education: Low-cost platforms for teaching integrated systems," *Robotics Automation Magazine, IEEE*, vol. 10, no. 2, pp. 4 – 6, June 2003.

[3] Z. Dodds, L. Greenwald, A. Howard, S. Tejada, and J. Weinberg, "Components, curriculum, and community: Robots and robotics in undergraduate ai education," *AI magazine*, vol. 27, no. 1, p. 11, 2006.

[4] S. P. N. Singh, R. Fitch, and S. Williams, "A research-driven approach to undergraduate robotics education," *Computers in Education Journal*, vol. 1, no. 4, pp. 21–27, 2010.

[5] F. Mondada, M. Bonani, X. Raemy, J. Pugh, C. Cianci, A. Klaptocz, S. Magnenat, J.-C. Zufferey, D. Floreano, and A. Martinoli, "The e-puck, a robot designed for education in engineering," in *Proceedings of the 9th conference on autonomous robot systems and competitions*, vol. 1, no. 1, 2009, pp. 59–65.

[6] M. W. Spong and D. J. Block, "The pendubot: A mechatronic system for control research and education," in *Decision and Control, 1995., Proceedings of the 34th IEEE Conference on*, vol. 1. IEEE, 1995, pp. 555–556.

[7] W. E. Dixon, D. M. Dawson, B. T. Costic, and M. S. De Queiroz, "A matlab-based control systems laboratory experience for undergraduate students: toward standardization and shared resources," *Education, IEEE Transactions on*, vol. 45, no. 3, pp. 218–226, 2002.

[8] R. Mitnik, M. Nussbaum, and A. Soto, "An autonomous educational mobile robot mediator," *Autonomous Robots*, vol. 25, no. 4, pp. 367–382, 2008.

[9] K. Trigwell and M. Prosser, "Improving the quality of student learning: the influence of learning context and student approaches to learning on learning outcomes," *Higher education*, vol. 22, no. 3, pp. 251–266, 1991.

[10] L. W. Anderson, D. R. Krathwohl, and B. S. Bloom, *A taxonomy for learning, teaching, and assessing: A revision of Bloom's taxonomy of educational objectives*. Allyn & Bacon, 2001.

[11] K. A. Ericsson, R. T. Krampe, and C. Tesch-Römer, "The role of deliberate practice in the acquisition of expert performance." *Psychological review*, vol. 100, no. 3, p. 363, 1993.

[12] G. A. Demetriou, *Mobile robotics in education and research*. IN-TECH Open Access Publisher, 2011.

[13] S. Galvan, D. Botturi, A. Castellani, and P. Fiorini, "Innovative robotics teaching using lego sets," in *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*. IEEE, 2006, pp. 721–726.

[14] M. Rosenblatt and H. Choset, "Designing and implementing hands-on robotics labs," *Intelligent Systems and Their Applications, IEEE*, vol. 15, no. 6, pp. 32–39, 2000.

[15] A. G. Alleyne, D. J. Block, S. P. Meyn, W. R. Perkins, and M. W. Spong, "An interdisciplinary, interdepartmental control systems laboratory," *Control Systems, IEEE*, vol. 25, no. 1, pp. 50–55, 2005.

[16] G. Wilfong, "Motion planning in the presence of movable obstacles," *Annals of Mathematics and Artificial Intelligence*, vol. 3, no. 1, pp. 131–150, 1991. [Online]. Available: http://dx.doi.org/10.1007/BF01530890

[17] S. A. Ambrose, M. W. Bridges, M. DiPietro, M. C. Lovett, and M. K. Norman, *How learning works: Seven research-based principles for smart teaching*. John Wiley & Sons, 2010.

[18] S. Sheppard, K. Macatangay, A. Colby, and W. M. Sullivan, *Educating engineers: Designing for the future of the field*. Jossey-Bass San Francisco, CA, 2009, vol. 9.

[19] R. Huber, H. Ramoser, K. Mayer, H. Penz, and M. Rubik, "Classification of coins using an eigenspace approach," *Pattern Recognition Letters*, vol. 26, no. 1, pp. 61–75, 2005.

[20] M. Furst, G. Kronreif, C. Wogerer, M. Rubik, I. Hollander, and H. Penz, "Development of a mechatronic device for high-speed coin sorting," in *Industrial Technology, 2003 IEEE International Conference on*, vol. 1. IEEE, 2003, pp. 185–189.

VI. APPENDIX 1: VIDEO ATTACHMENT

A video of the CHARM robot sorting coins also online at: `http://youtu.be/hH5_9t7lPhQ`.