

Movement Computing Education for Middle Grades

Yoav Bergner
yoav.bergner@nyu.edu
New York University

Deborah Damast
deborah.damast@nyu.edu
New York University

Allegra Romita
aar451@nyu.edu
New York University

Anne Marie Robson Smock
ams1730@nyu.edu
New York University

ABSTRACT

This paper takes a theoretical approach to movement computing education for young learners, with a focus on middle grades (grades 6-8, ages 11-14). This age group is targeted as a lower bound because, while some elements of computational thinking may be available to still younger learners, there are abstractions involved in movement computation that pre-require a certain amount of formal operation, in the Piagetian sense. We outline a parallel foundation of key ideas in movement (specifically dance) and key ideas in computing (specifically data representations) at this age-appropriate level. We describe how these foundations might be laid down together early on so that they can later be integrated via the introduction of sensing and feedback technology. Concepts in movement and choreography are studied using words and bodies, as in traditional dance education, and later using computer simulations and motion capture. Data concepts are introduced first by appeal to general questions and later by specification to the movement of individual and collective joints and bodies.

CCS CONCEPTS

• **Applied computing** → **Performing arts; Computer-assisted instruction.**

KEYWORDS

dance analytics, motion capture, data science, education

ACM Reference Format:

Yoav Bergner, Deborah Damast, Allegra Romita, and Anne Marie Robson Smock. 2020. Movement Computing Education for Middle Grades. In *7th International Conference on Movement and Computing (MOCO '20)*, July 15–17, 2020, Jersey City/ Virtual, NJ, USA. ACM, New York, NY, USA, 5 pages. <https://doi.org/10.1145/3401956.3404238>

1 INTRODUCTION AND BACKGROUND

For as long as computers have been available to artists, they have explored and advanced the frontier of creation with these “new media” tools. Art education, however, did not rush in as quickly. To be

sure, studio art, music, and dance have traditions of practice that are thousands of years old, next to which the computer age is a recent blip. Teaching of computing in schools initially emphasized either learning how to use common software applications—sometimes referred to as information and communication technology (ICT) skills—or learning to write computer programs [18]. Recent years, however, have seen a shift towards a more holistic view of computing education. This turn has included both a broader range of computing topics [12] and problem solving skills under the banner of computational thinking [30]. Not only did artists embrace computers, but computer scientists came to embrace artists and artistically minded learners through developments in creative computing. New pedagogical approaches emerged, especially within the past decade, for e-textile crafting [7, 19], computer music creation [21], and digital media manipulation [16]. However, while the intersection of dance and computing has matured among adult artists and professionals, this inter-disciplinary domain has only just begun to make inroads into school-age study [14].

Notable efforts to bring computational science and dance together for young learners include STEM from Dance (stemfromdance.org), an organization that teaches physical computing and programming to girls, culminating in a tech-enhanced live performance; Embodied Physics (terc.edu/projects/embodied-physics), a project targeting kinesthetic knowledge of physics through dance; and Dance Party (athletesforcomputerscience.org/dance-party.html), a block-based environment for choreography with animated sprites.

In this work, we consider the problem of developing a truly bilateral movement computing education program for middle grades. Bilateral means that key movement concepts are explored through computing technologies, while learners simultaneously develop key ideas in computation through the combined experience of enacting, describing, capturing, and analyzing movement. We focus on dance because it is possible to connect to existing standards already embraced by school systems (e.g., [23]) and to train pre-service dance teachers in movement computing. We focus on middle grades as a target population because certain topics in computing require a degree of abstraction that is associated with Piaget’s formal operational stage of development [25]. At this stage, adolescents can reliably imagine hypothetical states such as alternate visual representations, and their development of time, speed, and distance concepts approaches that of adults [27].

Computing in the context of dance learning is compelling for several reasons. For one, dance is a fertile ground for developing culturally relevant and sustaining pedagogy [20]. Second, embodied learning of computing concepts can emerge from the pairing

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

MOCO '20, July 15–17, 2020, Jersey City/ Virtual, NJ, USA

© 2020 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 978-1-4503-7505-4/20/07...\$15.00

<https://doi.org/10.1145/3401956.3404238>

of dance and computing [3, 29]. Third, computing-style abstractions can inform dance education, just as professional artists' use of computers informs their dance practice. Dual-coding theory [11] suggests that concept encoding (learning) is improved when verbal and non-verbal modalities are integrated. Computer visualizations and simulations, along with movement, complement verbal and symbol systems. Finally, exposing young dance students to authentic computing experiences may yield unique perspectives into human-computer interaction.

El Raheb and collaborators [14] recently conducted a thorough survey of dance interactive learning systems (DILS). The essential characteristics of DILSs is that they are (a) designed for educational purposes, (b) use whole-body movement, and (c) include visualization modalities (e.g., mixed/virtual/augmented reality). DILSs do not, however, centralize the importance of learning about computing in the process of learning about dance. By contrast, we consider putting the learning objectives of movement and computing on equal footing. Our focus on the needs of younger learners is also informed by perspectives of cognitive science and interaction design. State-of-the-art systems for annotating or creating computer choreography (e.g., DanceForms [8] or the Choreographic Language Agent [13]) have been designed for expert users. While professional choreographers require advanced creative control—and are willing to pay a price in complex, idiosyncratic interfaces [1]—young learners have different needs. Movement computing for learners requires simplified systems that segment and scaffold instruction, pre-train necessary skills, and signal the salient features using multiple representations.

Our approach builds on related scholarship at the intersection of movement analysis and education. Projects using computational analysis and data visualization with high school step dance teams [3] and in STEAM Makerspaces [10] have shown how students' interests in dance can be an engaging entry point into science and technology. Previous work has also explored practical applications including feedback of correct postural and skeletal alignments (specific to codified techniques) [22] and assessment of rhythmic patterns performed in the body [17]. Several studies have investigated the analysis and notation of movement through the use of motion capture technologies and Laban Movement Analysis or Labanotation [2, 5, 26].

2 SCOPING THE LEARNING GOALS

Either dance or computing alone can easily fill the time allocated to a middle-grades curriculum. Teaching both topics requires deliberate selection. This selection ought to prioritize topics from movement and computing that are complementary, so that these ways of thinking mutually inform each other. It is advisable to trade breadth for depth. In this section, we examine a selection of key ideas with reference to educational standards from both dance—namely the *Blueprint* of the New York City Department of Education [23]—and computing (the CSTA K-12 standards [12]).

Key Ideas in Movement

The NYC *Blueprint* has five strands of focus for grades Pre-K through 12: dance making; developing dance literacy; making connections; working with community and cultural resources; and exploring

careers and life long learning. The first three of these strands are more relevant to a movement computing program. A selection of relevant sub-categories include the following: transfer spatial patterns from visual to kinesthetic; analyze basic structural elements; record changes in a dance sequence through writing, symbols, or a form of media technology; analyze how movement and relationship create meaning.

Consider the following activity from the perspective of dance literacy. One dancer stands in the center of the space while another orbits around them in a circular pathway facing into the circle. The orbiter then maintains their pathway but changes their facing to outside of the circle, away from the dancer in the center. Observers will attempt to create meaning from this movement exploration. They will identify and articulate the perceived relationship between the two bodies in each movement scenario. Their observations will be discussed and translated into notation. This is one example of a basic exercise that could be built into a unit exploring directions, facings, and pathways, and the meaning they elicit.

Computer simulations—whose value for concept learning in science is well-documented [28]—can enhance this learning experience. For example, a sandbox environment under development allows students to select combinations of facings and pathways and animate dancers (circles or icons) in a bird's eye view of the stage. This kind of simulation may be contrasted with creativity support tools used by professionals and with "edutainment" software such as Dance Party. Simulators are deliberately limited and simplified. For example, pathway choices are circular or linear. Each choice is accompanied by some parametric degrees of freedom (e.g., radius, path-length). Facings can be relative to the dancer-self (forwards, left), to the stage (front), or to another dancer. Using simulation tools in addition to verbal descriptions and enacted movements helps to reinforce the connection between symbolic, iconic, and enactive representations [6].

In sports medicine and physical therapy, computing systems and motion sensors are commonly used to analyse the biomechanics of a patient to help re-train ineffective or harmful movement patterns. In contrast, many dance forms value the aesthetic of the codified technique and prioritize the structure and outward appearance of the body. Motion capture and data analysis can be used to provide feedback for the biomechanical and aesthetic requirements of dance [9]. Computing can also be used to elaborate differences between basic locomotor and non-locomotor movements along with elements of composition such as directions, facings, and pathways. The lessons will guide students to embody Laban effort intent and discover dance literacy through the Laban notation system and choreographic toolbox.

Key Ideas in Computing

The CSTA standards make explicit reference to five concept areas in the K-12 framework (k12cs.org): computing systems; networks and the internet; data and analysis; algorithms and programming; and impacts of computing. Insofar as a movement computing curriculum involves learning about hardware systems—either wearable systems, external sensors, or hybrid motion capture—this would fall under computing systems. If students learn to write programs to simulate movement or process movement data (including machine

Table 1: Foundation units in dance and data: Questions and key ideas

Unit Questions Ideas	Introduction to Dance What is dance? What are movements that we can use to build a dance? Dances can be created by combining individual movements into a phrase.	Introduction to Data What are data? Data are observations/values that carry information about people or events of interest.
Unit Questions Ideas	Introduction to the Choreographic Toolbox What kinds of choices can we make to create a new dance? We can use choreographic tools to manipulate a basic phrase and make it unique.	Asking Questions with Data What kinds of questions can we ask of data? We can use data to describe what and how, but not why. We can group and classify using data.
Unit Questions Ideas	Conveying Ideas through Choreography What ideas may be conveyed by choreographic choices? Can people get different ideas from watching the same dances? Choreography can be intentional in creating a message or conveying an idea to our audience.	Capturing Data What is captured when we capture data? What is lost in the process? Are data objective? The question we want to answer (the intention) and the tools we use influence the nature of the data we capture.
Unit Questions Ideas	Exploring Movement Effort How can we explore and use movement effort? What can movement effort convey within a choreographic structure? Effort is another layer we can add to our movement to make it more interesting, unique, and effective.	Making Sense of Data How do we turn data into information? What is the data analysts toolbox? In order to be able to make sense of data and make it meaningful, we often need a simplification (model) of reality.
Unit Questions Ideas	Effort Phrasing How do different choices of phrasing impact the movement and choreographic ideas? Effort phrasing is an effective tool in conveying a concept of a dance.	Visualizing Data How do different choices of representation result in different ways of knowing and communicating? Data visualizations can help us take in a lot of information at once, but visualization choices have an impact on the information conveyed.
Unit Questions Ideas	Communicating with and Critiquing Dance What questions can we ask to learn about the choreography of others? What are helpful ways to give feedback? What criteria do we value when watching dance? We can learn from observing dances and discussing our thoughts and reactions to them. Receiving feedback is an important step in revising and improving our dances.	Communicating with and Critiquing Data What questions are important to ask when encountering a data analysis? Data analysis can tell a story, but the recipient must engage with it. What is missing from the data analysis is sometimes as important as what is present.

learning classification), these topics would fall under algorithms and programming. While sensing technologies (systems) and visualization programs will inevitably be a part of the movement computing experience, we suggest that a good starting point for instruction is the data and analysis category.

There are a few reasons for starting with data. For one thing, dance notation systems that are not computer based, e.g., Labanotation, already expose students to the idea of symbolic abstraction by representing movement using a precise labeling system. Thus students will have available to them analogies between dance “language” and data representations. Analogies provide multiple access points and ways of reasoning that strengthen encoding/learning [15]. Physical computing systems, while important, do not synergize in quite the same way with early concept learning about dance movement. The computing systems may seem like entirely separate problem domains. Moreover, hardware is an extensive subject to master, and system failure can be very frustrating. We are in favor of productive failure when it comes to learning about hardware. But when learning about movement computing, we feel that hardware challenges contribute to cognitive overload [24], preventing the learner from staying focused on key ideas. Similar considerations apply in choosing data representations over coding techniques.

Level 2 (grades 6-8) CSTA standards under the data and analysis strand include the following: represent data using multiple encoding schemes; collect data using computational tools and transform the data to make it more useful and reliable; and refine computational models based on the data they have generated. (Create interactive data visualizations using software tools is a level 3 standard—grades 9-10). It should be easy to imagine how movement-related data can be used to anchor these experiences. For example, the learner can establish coordinate systems in which to situate the dancer’s body;

analyze absolute and relative positions (between joints or between bodies); and compare actual movements or postures to idealized values (computing “distances” in a classifier). Motion capture devices, described in section 4, will present concrete examples of data encoding schemes, while visualization tools will employ data transformation.

3 FOUNDATION AND INTEGRATION

Having considered the scope of learning objectives from the movement and computing fields, we now provide an example guide for curriculum design. We have selected a small range of topics and outlined each one using guiding questions and key ideas rather than explicit activities. This scoping supports either designing a shorter course (e.g., one semester) or achieving more depth through extended exposure. The latter would be ideal, but an experimental course might be short. The structure is as follows. We establish a basic foundation in movement (dance) and computing (data) through six units. Building on this, we describe six integration units that weave dance and data concepts together.

The foundational units are summarized in Table 1. The guiding principle is to establish two parallel ways of thinking from the start of the course. Dance vocabulary and data vocabulary need to be articulated independently. A class that meets twice a week might allocate one day to dance and the other to data. That said, effort is made not only to anticipate the integration of these topics but to follow similar structure of inquiry for both. Through the key ideas and questions, we consider it essential to anticipate the limits of movement computing. For example, what are the problems of data reduction, of operationalization choices, and of subjective experience?

Following the introductory units, data and dance become more integrated in the next group of units. We suggest four potential units and provide a brief synopsis for each. We have already included in Section 2 to the enactment and simulation of simple combinations of *Facings and Pathways*. Motion capture can also be included here, given suitable technology (see below). *Definitions and Boundaries (Efforts + Distinctions)* is a unit dedicated to Laban effort qualities, with an acknowledgement of the limitations observed in perceptual experience[4]. What is the definition of direct/indirect, quick/sustained? Where is the boundary between the ends of each scale, and how would a computer be instructed to tell them apart? *Feedback and Perception* is dedicated to posture classification and proprioception and introduces the possibility of responsive/interactive technology. *Underdetermination* considers what is left out from choreographic annotation systems and data visualizations. For example, rather different movements can still be described by the same pathway, if we only track one part of the body on the plane of the stage.

4 TECHNOLOGY REQUIREMENTS

The technology frontier occupied by researchers and professionals is a bridge too far for schools, where systems need to be reasonably cheap and portable while still reliable. We consider some requirements for technology in a movement computing curriculum and propose some specific options. For example, it must be possible to track the positions of at least two dancers on the scale of a small stage (e.g., 5 meters). It must also be possible to track whole bodies and the relative position of body parts.

Although computer-vision (CV) skeletonization is possible using ordinary video, we do not recommend it for most classrooms. The computational burden for real time analysis is high, and CV algorithms are noisy compared with results from Kinect sensors. Errors, such as body parts becoming detached or mis-connected to other bodies, are likely to be distracting. Meanwhile, Kinect sensors are inexpensive and require only a basic classroom computer. Kinectron (kinectron.github.io) allows the Kinect feed to be broadcast and received by multiple PCs running web apps. Thus students can interact individually or in small groups with a single Kinect feed.

While the Kinect sensor is adequate for “local” body movements, it has some drawbacks in tracking the position of one or more bodies on the scale of a stage. This is because there is a limited range of both camera angle and depth sensitivity. Two different Kinect sensors can be used to track a body, with each one tracking one extent dimension of the room (i.e., width and length). An easier option is to use radio-frequency position tracking devices for this task. An example is Pozyx (pozyx.io), a slightly pricey system for classroom use (\$1000) but very accurate. The feed from such tracking systems can be broadcast to interactive web apps using the MQTT protocol.

5 CONCLUSION

We believe that a movement computing education program for middle grades is tractable and worth pursuing. We have outlined in brief some design considerations, taking into account established learning objectives from dance and computing while seeking to maximize the synergy from their interaction.

REFERENCES

- [1] Sarah Fdili Alaoui, Kristin Carlson, and Thecla Schiphorst. 2014. Choreography as mediated through compositional tools for movement: Constructing a historical perspective. *ACM International Conference Proceeding Series* (2014), 1–6. <https://doi.org/10.1145/2617995.2617996>
- [2] Anastasios Ballas, Tossaporn Santad, Kingkarn Sookhanaphibarn, and Worawat Choensawat. 2017. Game-based system for learning labanotation using Microsoft Kinect. In *6th Global Conference on Consumer Electronics (GCCE)*. IEEE, 1–3.
- [3] Yoav Bergner, Shiri Mund, Ofer Chen, and Willie Payne. 2019. First steps in dance data science: Educational design. In *Proceedings of the 6th International Conference on Movement and Computing*. 1–8.
- [4] Ulysses Bernardet, Sarah Fdili Alaoui, Karen Studd, Karen Bradley, Philippe Pasquier, and Thecla Schiphorst. 2019. Assessing the reliability of the Laban Movement Analysis system. *PLoS one* 14, 6 (2019).
- [5] Ran Bernstein, Tal Shafir, Rachelle Tsachor, Karen Studd, and Assaf Schuster. 2015. Laban movement analysis using kinect. *Int. J. Comput. Electr. Autom. Control Inform. Eng* 9 (2015), 1394–1398.
- [6] Jerome S. Bruner. 1964. The course of cognitive growth. *American Psychologist* 19, 1 (1964), 1–15. <https://doi.org/10.1037/h0044160>
- [7] Leah Buechley. 2006. A construction kit for electronic textiles. In *2006 10th IEEE international symposium on wearable computers*. IEEE, 83–90.
- [8] Tom Calvert, W Wilke, Rhonda Ryman, and Ilene Fox. 2005. Applications of computers to dance. *IEEE computer Graphics and Applications* 25, 2 (2005), 6–12.
- [9] Antonio Camurri, Gualtiero Volpe, Stefano Piana, Maurizio Mancini, Radoslaw Niewiadomski, Nicola Ferrari, and Corrado Canepa. 2016. The Dancer in the Eye: Towards a Multi-Layered Computational Framework of Qualities in Movement. *Proceedings of the 3rd International Symposium on Movement and Computing* (2016), 6:1–6:7.
- [10] Dionne N Champion. 2018. *The STEAM dance makerspace: A context for integration: An investigation of learning at the intersections of STEM, art, making and embodiment*. Ph.D. Dissertation. Northwestern University.
- [11] James M Clark and Allan Paivio. 1991. Dual Coding Theory and Education. *Educational Psychology Review* 3, 3 (1991), 149–210.
- [12] Computer Science Teachers Association [CSTA]. 2017. *CSTA K–12 Computer Science Standards - Revised 2017*. Technical Report July. 1–28 pages.
- [13] Scott DeLahunta. 2016. Wayne McGregor’s Choreographic Language Agent. In *Transmission in Motion: The Technologizing of Dance*, Maaïke Bleeker (Ed.). Routledge, New York, 108–117. <https://doi.org/10.4324/9781315524177>
- [14] Katerina El Raheb, Marina Stergiou, Akrivi Katifori, and Yannis Ioannidis. 2019. Dance interactive learning systems: A study on interaction workflow and teaching approaches. *Comput. Surveys* 52, 3 (2019). <https://doi.org/10.1145/3323335>
- [15] Mary L Gick and Keith J Holyoak. 1983. Schema induction and analogical transfer. *Cognitive Psychology* 15 (1983), 1–38.
- [16] Mark Guzdial. 2013. Exploring hypotheses about media computation. In *Proceedings of the ninth annual international ACM conference on International computing education research*. 19–26.
- [17] Ta-Che Huang, Yu-Chuan Cheng, and Cheng-Chin Chiang. 2013. Automatic dancing assessment using Kinect. In *Advances in Intelligent Systems and Applications-Volume 2*. Springer, 511–520.
- [18] Peter Hubwieser, Michal Armoni, Michail N Giannakos, and Roland T Mittermeier. 2014. Perspectives and visions of computer science education in primary and secondary (K-12) schools. *ACM Transactions on Computing Education (TOCE)* 14, 2 (2014), 1–9.
- [19] Yasmin Kafai, Deborah Fields, and Kristin Searle. 2014. Electronic textiles as disruptive designs: Supporting and challenging maker activities in schools. *Harvard Educational Review* 84, 4 (2014), 532–556.
- [20] Ladson-Billings and Gloria. 2014. Culturally Relevant Pedagogy 2.0: a.k.a. the Remix. *Harvard Educational Review* 84, 1 (2014), 76–84.
- [21] Brian Magerko, Jason Freeman, Tom McKlin, Scott McCoid, Tom Jenkins, and Elise Livingston. 2013. Tackling engagement in computing with computational music remixing. In *Proceeding of the 44th ACM technical symposium on Computer science education*. 657–662.
- [22] Zoe Marquardt, João Beira, Natalia Em, Isabel Paiva, and Sebastian Kox. 2012. Super Mirror: a kinect interface for ballet dancers. In *CHI’12 Extended Abstracts on Human Factors in Computing Systems*. 1619–1624.
- [23] New York City Department of Education. 2015. *Blueprint for Teaching and Learning in Dance: Grades PreK-12*. Technical Report.
- [24] Fred Paas and John Sweller. 2014. Implications of cognitive load theory for multimedia learning. In *The Cambridge Handbook of Multimedia Learning, Second Edition* (2 ed.). Cambridge University Press, Cambridge, 27–42. <https://doi.org/10.1017/CBO9781139547369.004>
- [25] Jean Piaget. 1970. *Science of education and the psychology of the child*. Viking Press.
- [26] Ioannis Rallis, Apostolos Langis, Ioannis Georgoulas, Athanasios Voulodimos, Nikolaos Doulamis, and Anastasios Doulamis. 2018. An embodied learning game using kinect and labanotation for analysis and visualization of dance kinesiology. In *2018 10th International Conference on Virtual Worlds and Games for Serious Applications (VS-Games)*. IEEE, 1–8.

- [27] Robert S Siegler and D Dean Richards. 1979. Development of time, speed, and distance concepts. *Developmental Psychology* 15, 3 (1979), 288.
- [28] Carl E Wieman, Wendy K Adams, and Katherine K Perkins. 2008. PhET: Simulations that enhance learning. *Science* 322, 5902 (2008), 682–683.
- [29] Margaret Wilson. 2002. Six views of embodied cognition. *Psychonomic Bulletin & Review* 9, 4 (2002), 625–636.
- [30] Jeannette M Wing. 2006. Computational thinking. *Commun. ACM* 49, 3 (2006), 33–35.