James D. Motes - Teaching Statement

I have been consistently drawn toward teaching and mentoring since my graduate studies. Even when I moved to Houston, Texas to pursue a startup venture after completing my Ph.D., the incubator where I worked was home to the Houston Robotics Club, a hobbyist group of high school and community college students building AI and robotics projects, and I naturally found myself mentoring and advising these young, passionate students on the topics I also loved. These experiences, combined with my continuing work as a part-time postdoc at the University of Illinois Urbana-Champaign (UIUC), where I mentored graduate and undergraduate researchers, made it clear that my long-term home should be in a faculty role rather than at a startup.

The desire to **ground theoretical work in real-world problems** is the underlying theme of my teaching philosophy. Computer science is uniquely positioned to move ideas from the whiteboard to running code and, in my case, to physical systems that embody those ideas in the world. I adopt a constructivist, project-based, active learning approach: students build durable understanding when they apply core concepts to authentic problems they care about, work in stable, collaborative teams, and receive timely feedback. In both my courses and my mentoring, I use backward design and cognitive apprenticeship, structuring labs, projects, and research milestones so that students first practice expert behaviors with close scaffolding and then take increasing ownership of the work within a supportive, inclusive community of practice.

Courses: At the undergraduate level, I am prepared to teach core computer science courses including an introductory programming course, data structures and algorithms, and introductory artificial intelligence or machine learning. My experience teaching introductory AI and ethics, as well as my research in algorithmic robotics and multi-robot planning, positions me well to teach upper-level electives in AI, robotics, and algorithmic decision making. At the graduate level, I look forward to offering research-led seminars in algorithmic robotics, AI for multi-robot systems, and, as my research progresses, co-planning with embodied agents.

I also plan to develop advanced courses that reflect my research: an algorithmic robotics course on task and motion planning and their interaction with learning-based components; an AI for multi-robot systems course covering the complexity of multi-robot planning, major algorithmic approaches, and recent developments; and a co-planning with embodied agents course, developed in collaboration with colleagues in HRI/HCI and related fields such as cognitive science and experimental psychology, in which students build and study systems where humans and robots share decision making.

Classroom Philosophy

My core classroom philosophy is that **students learn best when they can see concepts build toward something they care about.** I approach courses with three guiding principles: (1) give students ownership by tying their work to their interests, (2) keep the semester coherent by scaffolding labs and assignments so they build toward a meaningful final project, and (3) design inclusion into the mechanics of the course. These principles reflect a constructivist, project-based view of learning, implemented through backward design and constructive alignment, and grounded in inclusive pedagogy.

This approach evolved over several semesters while I taught Discover AI, an introductory AI and ethics course for AI4ALL, in response to student feedback and outcomes. After covering foundational AI and ML concepts, student teams chose a theme from their personal interests for their project. Rather than standalone labs, every lab and assignment after that point could be used as a building block for the final project, which allowed me to monitor whether students were understanding both the current topic and how earlier concepts fit together across the course. At the end of the semester, teams presented projects they were genuinely proud of and excited to share and often incorporated concepts beyond the official curriculum that developed out of their own curiosity.

The ethics component of Discover AI leveraged the same teams in a breakout discussion format. Teams would first discuss a case study among themselves and then provide written responses on a shared digital whiteboard before we moved into a whole-class discussion. This structure offered multiple ways to participate: quieter students could test their ideas in a small group or via writing before speaking. Throughout both lab and ethics sessions, the TA and I moved between teams to help them connect their assignments and discussions to their projects. These practices increased engagement and acted as ongoing formative assessment that helped us identify struggling students early, while also lowering the barrier to asking for help. They align with inclusive pedagogy and Universal Design for Learning by providing multiple modalities for engagement and expression.

At UIUC, I also helped design and deliver a robotic motion planning crash course to new graduate and undergraduate students who joined our lab, pairing technical lectures with programming assignments in the lab's main codebase. Students worked with the same tools the group uses for research. Version control, experiment logging, and code review were explicit learning outcomes. Because implementations were extended rather than discarded, students saw why design choices, documentation, and testing matter when their code becomes the base for later work. This is an example of cognitive apprenticeship and research-led teaching, where students begin with guided tasks at the periphery of ongoing projects and progressively take on more central, independent roles.

To support these structures, I build frequent formative assessments and structured reflection into my courses. It is my practice to keep predictable weekly office hours and coordinate with teaching assistants so that small-group time, check-ins before deadlines, and timely feedback are built into the course cadence. Labs and in-class activities serve as low-stakes opportunities for students to practice new ideas and for me to gauge how well students are understanding key concepts. I solicit mid-semester feedback and use patterns I see in student questions and cumulative project work to adjust pacing, scaffolding, and support in future iterations of the course. Final projects and, in research settings, code-reviewed contributions to a shared codebase function as informative assessments that demonstrate whether students have achieved the intended learning outcomes. I use this evidence to guide improvement of my teaching over time.

Mentoring Philosophy

Mentoring has been the defining through-line of my graduate and postdoctoral career, and I believe that **research is best served by developing the researcher.** At UIUC, our lab uses a hierarchical mentoring model: postdocs and senior Ph.D. students mentor junior Ph.D. and master's students, who in turn mentor undergraduates, all under faculty supervision. Due to a generational gap in the group, I stepped into the senior student role unusually early, and from the second year of my graduate studies onward, most major projects I worked on were driven day-to-day by students I was mentoring. Through my role as a graduate student and postdoc, I have mentored 20+ graduate students and dozens of undergraduates through both local and inter-university programs like NSF DREU.

My mentoring process parallels how I approach course design and reflects a cognitive apprenticeship model. Early in a collaboration, I help scope projects to align research needs with the student's individual interests, set expectations, and establish milestones that build toward a larger goal. I am hands-on in the beginning, walking through code, modeling how to debug, and connecting specific tasks back to the larger research questions described in my research statement. As students gain expertise, I gradually remove this scaffolding. They start proposing their own research directions, making design decisions, and eventually mentoring more junior students. Throughout, I aim to create an inclusive lab environment by setting clear expectations, ensuring that students from diverse backgrounds have access to impactful projects, and normalizing questions and iteration as part of the research process.

The impact of this approach is visible in the trajectories of my mentees. For example, Irving Solis and Hannah Lee both completed their Ph.D.s after years of collaboratively advancing our multi-robot planning research, and now lead and mentor teams as a postdoc and as a robotics researcher in industry. Scott Lee, a current Ph.D. student, is carrying forward the main threads of my graduate research. His first submission in our lab is currently under review, and he is applying our research concepts as a Ph.D. Resident at Google X. In my future lab, I plan to continue this hierarchical mentorship approach with an emphasis on developing accomplished researchers. New students will be brought up to speed through a structured crash course and will be guided by both me and their senior student mentor. Senior students will graduate prepared not only to conduct research, but also to lead and mentor teams of new researchers.