

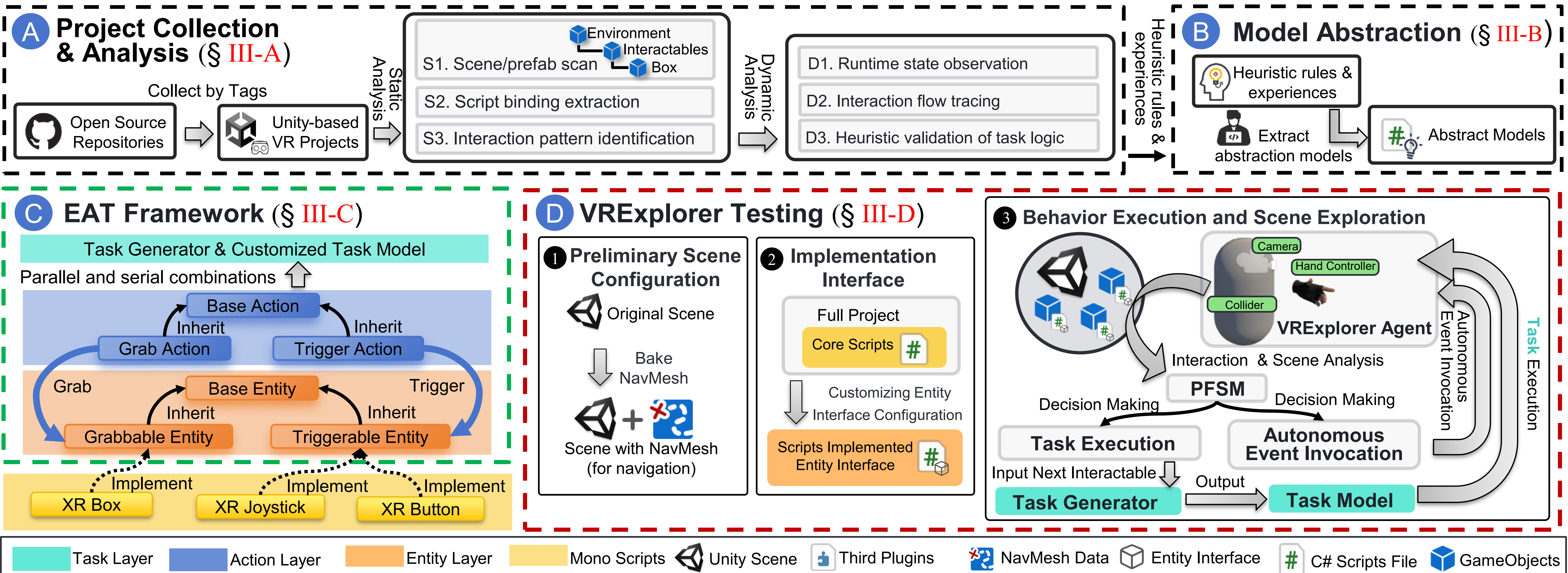
VRExplorer: A Model-based Approach for Semi-Automated Testing of Virtual Reality Scenes

Zhengyang Zhu^{1,3}, Hong-Ning Dai^{2*}, Hanyang Guo¹, Zeqin Liao¹, Zibin Zheng¹

¹School of Software Engineering, Sun Yat-sen University, Zhuhai, Guangdong, China;

²Department of Computer Science, Hong Kong Baptist University, Hong Kong, China;

³Peng Cheng Laboratory, Shenzhen, Guangdong, China



Challenges in VR Testing

- Existing tools fail to fully support diverse VR interactions (e.g., grab, press, shoot).
- Vast exploration space due to complex 3D environments and object interactions.
- Task sequences with interdependent actions (e.g., find key → unlock door → turn handle → press button).

Approach

- Project Collection & Static/Dynamic Analysis**
- Model Abstraction**: Generalizes objects and actions into abstract models.
- EAT Framework**: Three-layer design—Entity, Action, and Task (sequential/parallel composition).
- Testing Process**: Performing task-based interaction testing via NavMesh navigation & PFMSM.

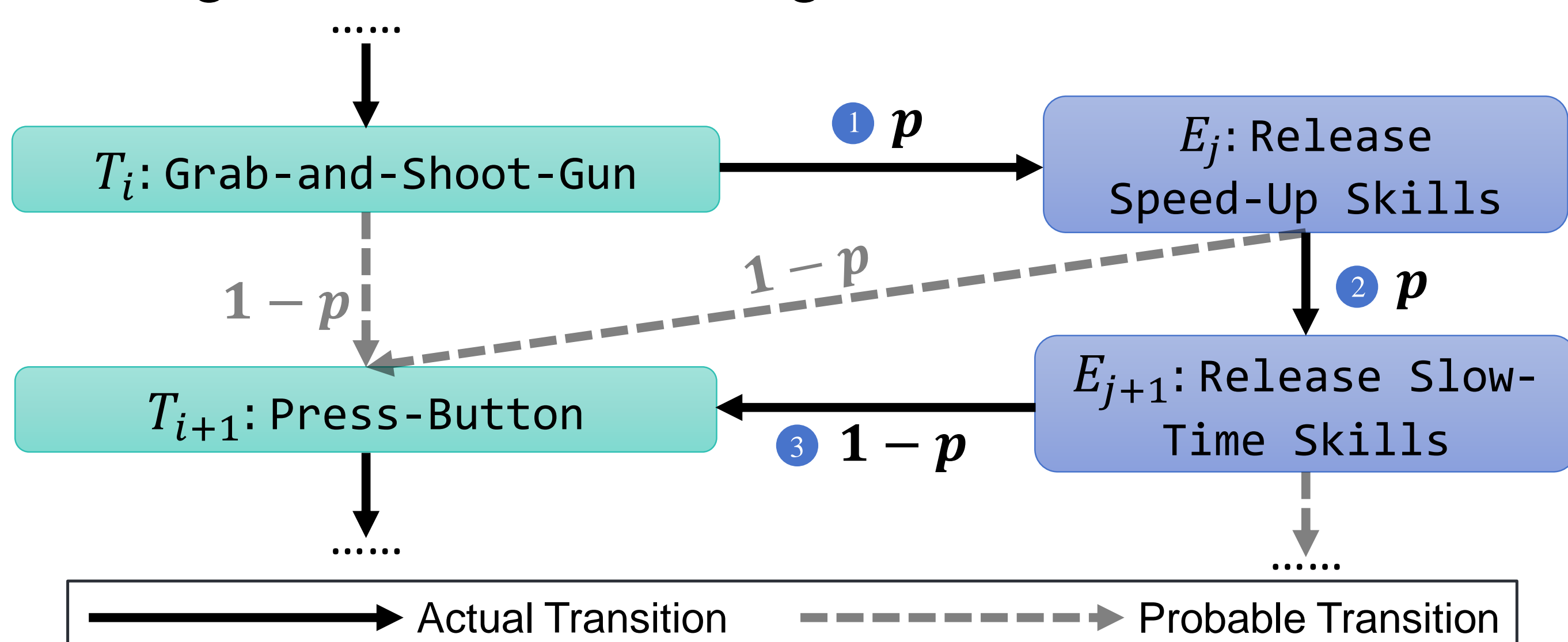


Figure: Example of PFMSM State Transition in a VR Scene.

Experiment Results

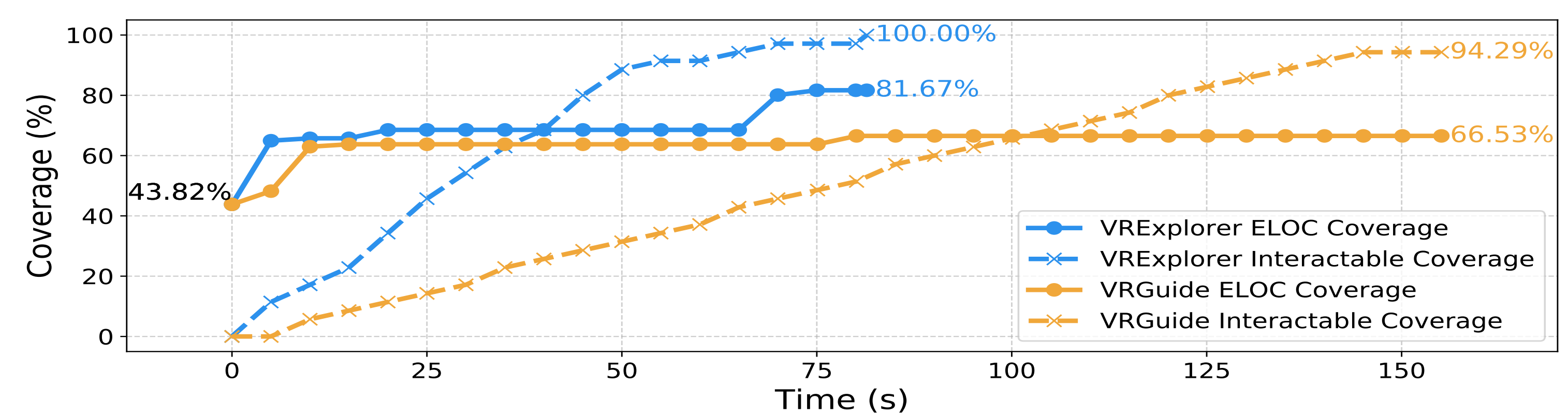


Figure: Coverage versus Time in A Project

- Performance**: Average gains of +122.8% EC and +52.8% MC vs SOTA on 11 Projects
- Real Bug Detection**: Found 3 real-world bugs (2 functional, 1 non-functional)
- Faster Convergence**: Achieves higher coverage with reduced time cost

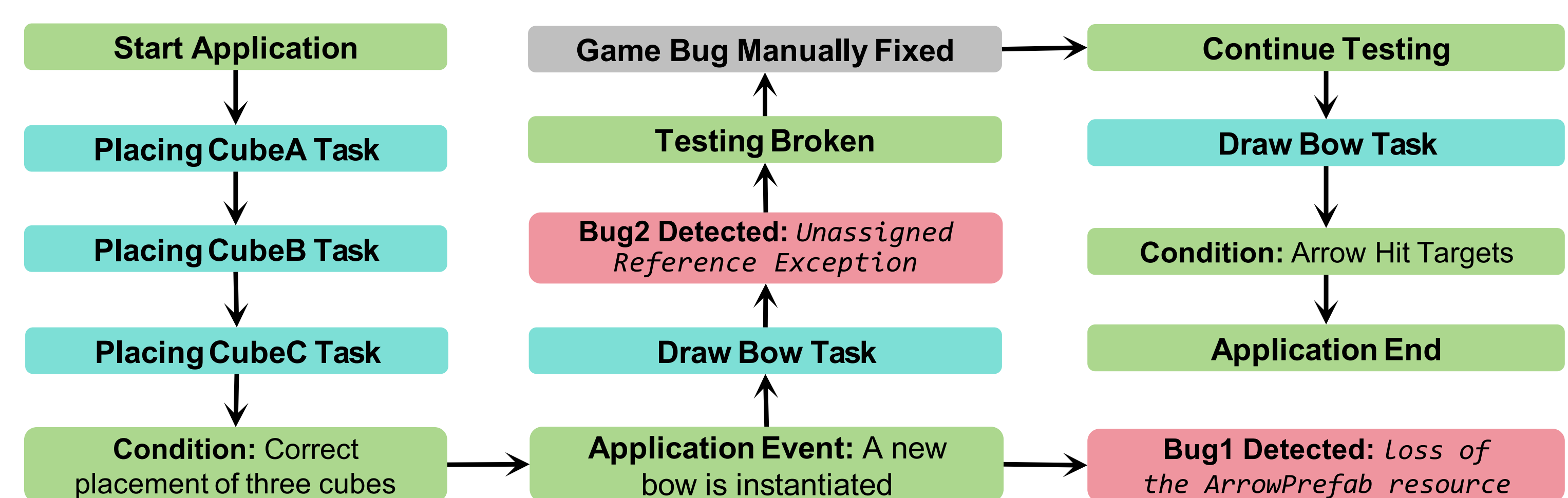


Figure: Example of how VRExplorer explores scenes and detects bugs

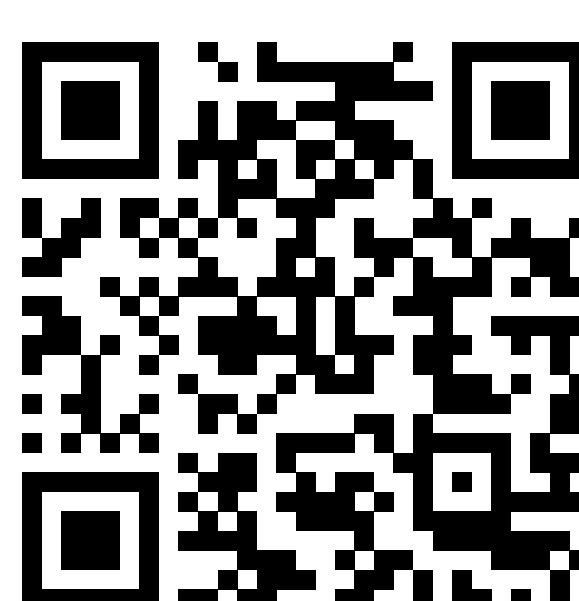
Contact

Zhengyang Zhu

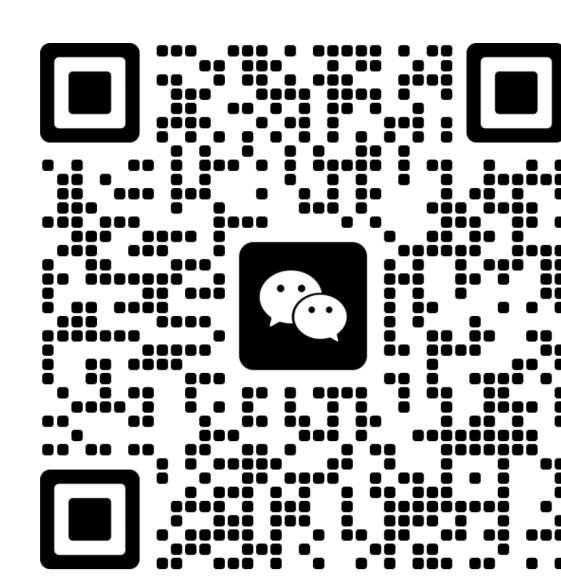
School of Software Engineering, Sun Yat-sen University

Email: zhuzhy57@mail2.sysu.edu.cn

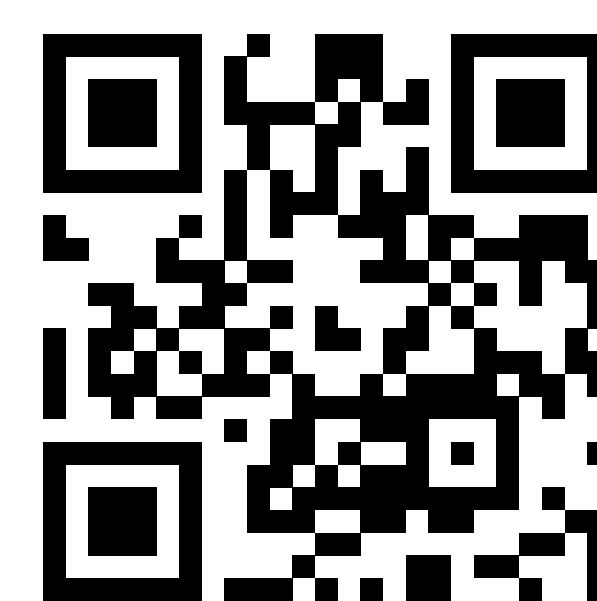
tsingpig@163.com



Video



WeChat



Homepage



Paper