ISCAS 中国科学院软件研究所学术年会’2023
暨计算机科学国家重点实验室开放周
学术论文

Context-aware Bug Reproduction for Mobile Apps
上下文感知的移动应用缺陷自动复现技术

Yuchao Huang, Junjie Wang, Zhe Liu, Song Wang, Chunyang Chen, Mingyang Li, Qing Wang
In the 45th IEEE/ACM International Conference on Software Engineering (ICSE 23)
联系人：黄芋超，王俊杰，王青　　联系方式：{yuchao2019, junjie, wq}@iscas.ac.cn

Background

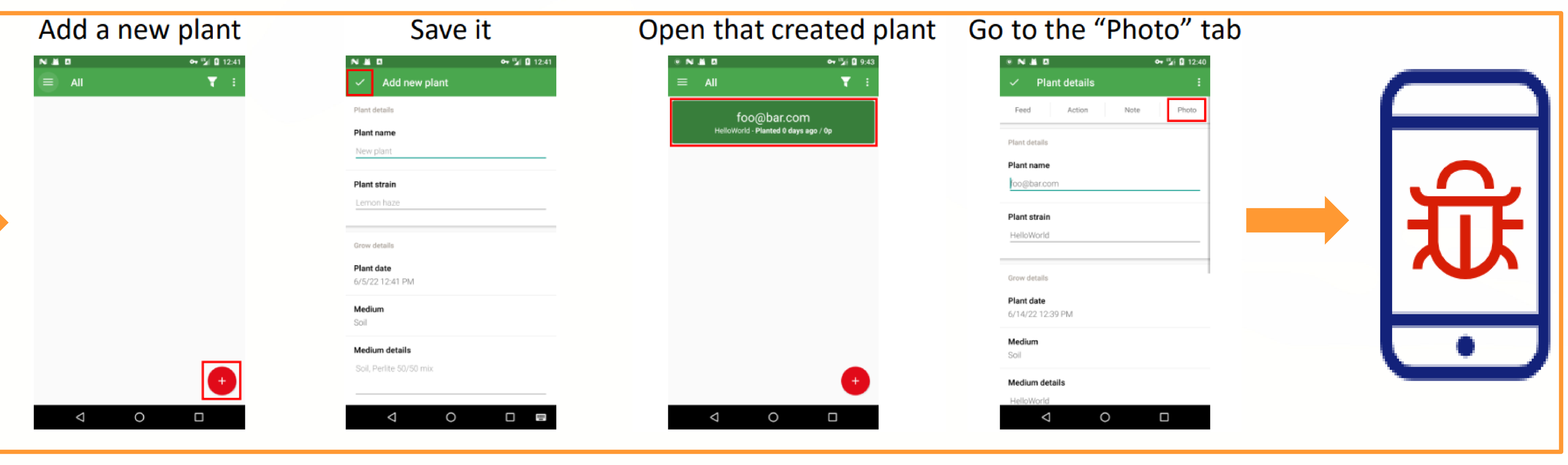
- **Reproducing mobile app bug reports**
  - A prerequisite task before fixing bugs.
  - Following reproduction steps in reports.
  - A tedious and time-consuming task
  - An automated tool is highly expected

Motivation

- **Challenges of auto bug reproduction**
  - Imprecise and incomplete reproducing step
  - Name-related information may be missing
  - Some reproducing steps may be omitted
  - Greedy-based exploration may be confused

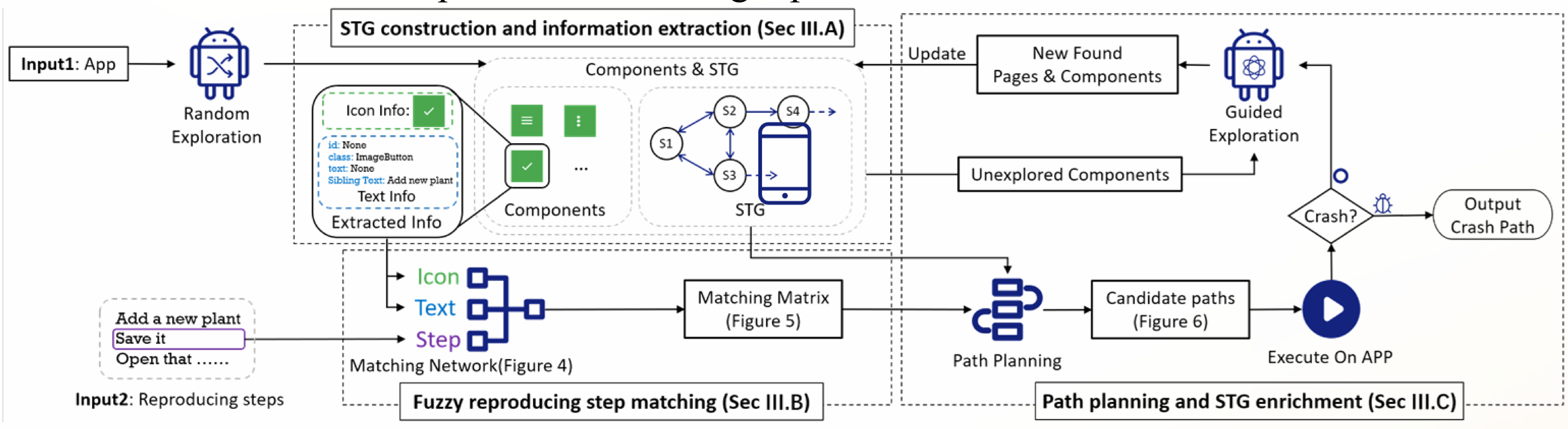
Example

To reproduce the bug:
1 Add a new plant
2 Save it
3 Open that created plant
4 Go to the “Photo” tab



Approach

◆ We propose **ScopeDroid, a context-aware bug reproduction approach** to reproduces crash from a text description of mobile bug reports.



Challenge 1 & 2: Addressed by our multi-modal neural matching network
Challenge 3 & 4: Addressed by our context-aware path planning

- **I: Construct STG & Info extraction**
  - Constructs an initial State Transition Graph (STG) with DroidBot
  - Extracts information from GUI components
    - Textual/Context information, Icon image

- **II: Fuzzy reproducing step matching**
  - A multi-modal neural network for matching reproducing steps with the GUI events
  - It derives a fuzzy matching matrix between each reproducing steps and each GUI events
  - We propose a heuristic data generation approach to train this network

- **III: Path planning & STG enrichment**
  - Planning the path that triggers the bug from a global perspective
  - We design a weighted algorithm to select to select the optimal path
  - Enriching the STG to include more states, update the path planning

Results and Analysis

- **RQ 1: Effectiveness & Efficiency**
  - ScopeDroid reproduce 65 (63%) crash reports in an average of 68 seconds
- **RQ 2: Matching Accuracy**
  - ScopeDroid’s matching module achieves 0.62-0.709 MRR on step-event matching
- **RQ 3: Usefulness**
  - ScopeDroid achieves a mean score of 4.14 in the user survey
- **RQ 4: Robustness**
  - ScopeDroid can still achieve a high success rate on the mutate reports

TABLE II: Performance of reproduction success rate

# Dataset	ReCDroid	ReCDroid+Maca	ScopeDroid
ReCDroid’s Dataset (33 Reports)	**32 (97%)**	**32 (97%)**	30 (91%)
AndroR2 Dataset (22 Reports)	6 (27%)	6 (27%)	**11 (50%)**
ScopeDroid’s Dataset (47 Reports)	9 (19%)	11 (23%)	**24 (51%)**
# Total (102 Reports)	47 (46%)	49 (48%)	**65 (63%)**

TABLE IV: Performance of the matching module for each approach

# Match Module	ReCDroid’s Dataset				
	Hit@1	**Hit@3**	**Hit@5**	**Hit@10**	**MRR**
ReCDroid	0.35	0.44	0.52	0.54	0.411
Maca	0.27	0.31	0.35	0.42	0.307
ScopeDroid	**0.61**	**0.77**	**0.85**	**0.91**	**0.709**
# Match Module	**AndroR2 Dataset**				
	Hit@1	**Hit@3**	**Hit@5**	**Hit@10**	**MRR**
ReCDroid	0.38	0.44	0.55	0.55	0.441
Maca	0.22	0.38	0.44	0.55	0.33
ScopeDroid	**0.55**	**0.83**	**0.83**	**0.94**	**0.682**
# Match Module	**ScopeDroid’s Dataset**				
	Hit@1	**Hit@3**	**Hit@5**	**Hit@10**	**MRR**
ReCDroid	0.29	0.36	0.36	0.36	0.319
Maca	0.17	0.21	0.23	0.36	0.213
ScopeDroid	**0.53**	**0.7**	**0.74**	**0.78**	**0.62**