October 7th ,2025

TEXAS
The University of Texas at Austin

VizDoom

Justin Medich

Motivation

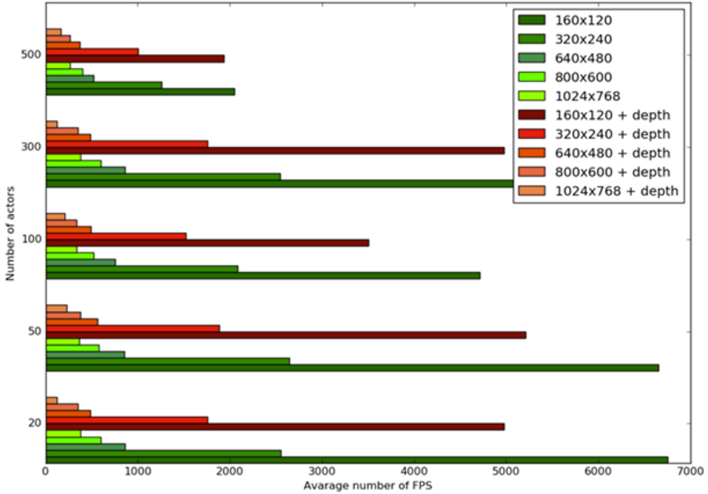
- Older benchmarks used only 2D visual information
  - ex. Atari 2600 games for DQN
- Benchmarks do not represent real-world tasks
- Many RL methods had mostly “solved” previous benchmarks
- Need a more challenging environment with additional reasoning possibilities

VizDoom Overview

- Based on well known video game
  - Highly Customizable
- First person perspective
- Simplistic semi-realistic 3D world
- Very lightweight game engine
  - Can be implemented on many platforms

Performance and Key Features

- ~7000 FPS on single CPU core
- Off-screen rendering
- Frame skipping support
- Depth buffer access
- Multiple resolution options



Why Doom?

Features / Game	Doom	Doom 3	Quake III: Arena	Half-Life 2	Unreal Tournament 2004	Unreal Tournament	Cube
Game Engine	ZDoom[1]	id tech 4	ioquake3	Source	Unreal Engine 2	Unreal Engine 4	Cube Engine
Release year	1993	2003	1999	2004	2004	not yet	2001
Open Source	✓	✓	✓			✓	✓
License	GPL	GPLv3	GPLv2	Proprietary	Proprietary	Custom	ZLIB
Language	C++	C++	C	C++	C++	C++	C++
DirectX		✓		✓		✓	
OpenGL	✓[3]	✓	✓	✓	✓	✓	✓
Software Render	✓						
Windows	✓	✓	✓	✓	✓	✓	✓
Linux	✓	✓	✓	✓	✓	✓	✓
Mac OS	✓	✓	✓	✓	✓	✓	
Map editor	✓	✓	✓	✓	✓	✓	✓
Screen buffer access	✓	✓	✓			✓	✓
Scripting	✓	✓		✓	✓	✓	✓
Multiplayer mode	✓	✓	✓		✓	✓	✓
Small resolution	✓	✓	✓	✓	✓	✓	✓
Custom assets	✓	✓	✓	✓	✓	✓	✓
Free original assets						✓	✓
System requirements	Low	Medium	Low	Medium	Medium	High	Low
Disk space	40MB	2GB	70MB	4,5GB	6GB	>10GB	35MB
Code complexity	Medium	High	Medium	-	-	High	Low
Active community	✓	✓	✓	✓		✓	
Brand recognition	31.5		16.8	18.7	1.0		0.1

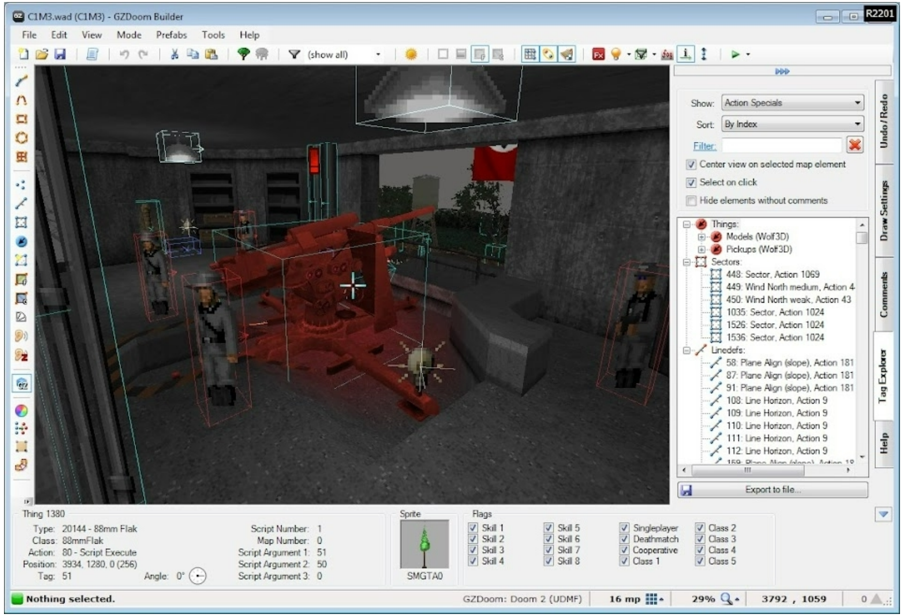
API

- 3 Different Control Modes
  - **Synchronous:** Game waits for agent decision
  - **Asynchronous:** Game runs at 35 FPS continuously
  - **Player/Spectator:** Agent controls or observes

```python
from vizdoom import *
from random import choice
from time import sleep, time

game = DoomGame()
game.load_config("../config/basic.cfg")
game.init()

# Sample actions. Entries correspond to buttons:
# MOVE_LEFT, MOVE_RIGHT, ATTACK
actions = [[True, False, False],
    [False, True, False], [False, False, True]]
# Loop over 10 episodes.
for i in range(10):
  game.new_episode()
  while not game.is_episode_finished():
    # Get the screen buffer and and game variables
    s = game.get_state()
    img = s.image_buffer
    misc = s.game_variables
    # Perform a random action:
    action = choice(actions)
    reward = game.make_action(action)
    # Do something with the reward...

  print("total reward:", game.get_total_reward())
```

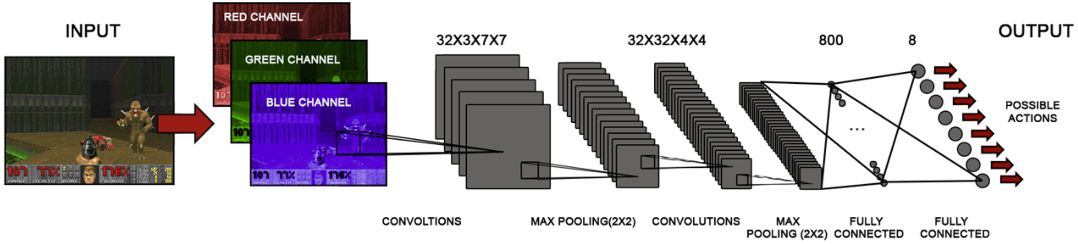
Scenarios

- Leverages Existing Doom Community tools
- Custom Maps, environment triggers, end conditions, and rewards
- Paper explores some example scenarios



Basic Experiment

- Utilizes basic environment
- Rewards defined as follows:
  - 101 killing monster
  - -5 for miss, -1 for each action
- Utilizes DQN for policy learning



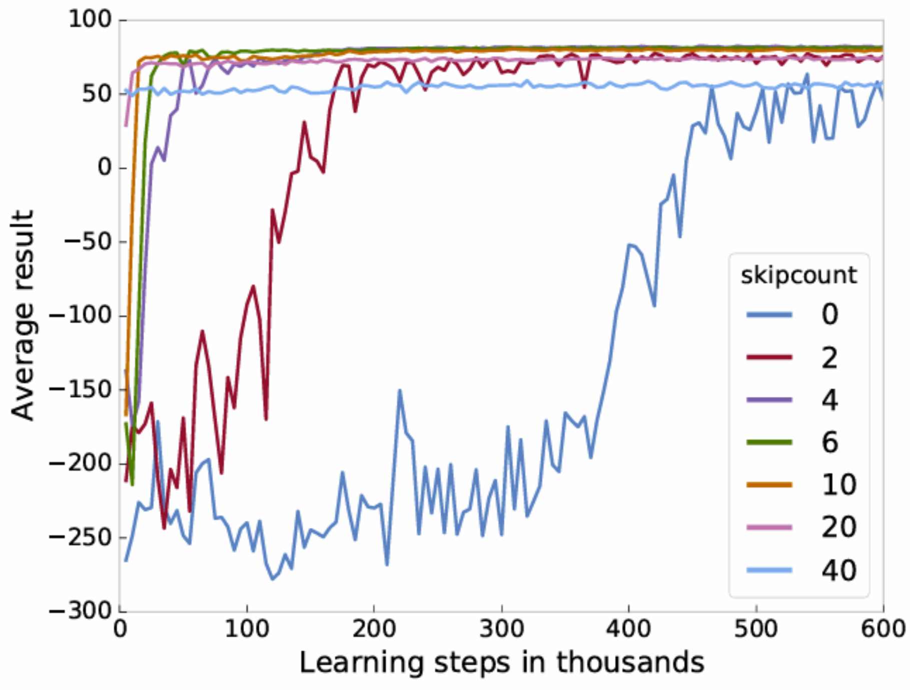


Table II

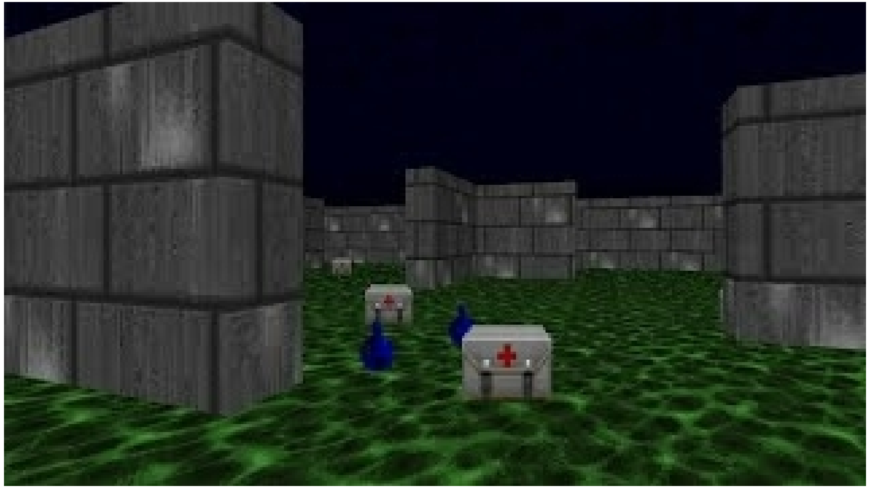
AGENTS' FINAL PERFORMANCE IN THE FUNCTION OF THE NUMBER OF SKIPPED FRAMES ('NATIVE'). ALL THE AGENTS WERE ALSO TESTED FOR SKIPCOUNTS $\in \{0, 10\}$.

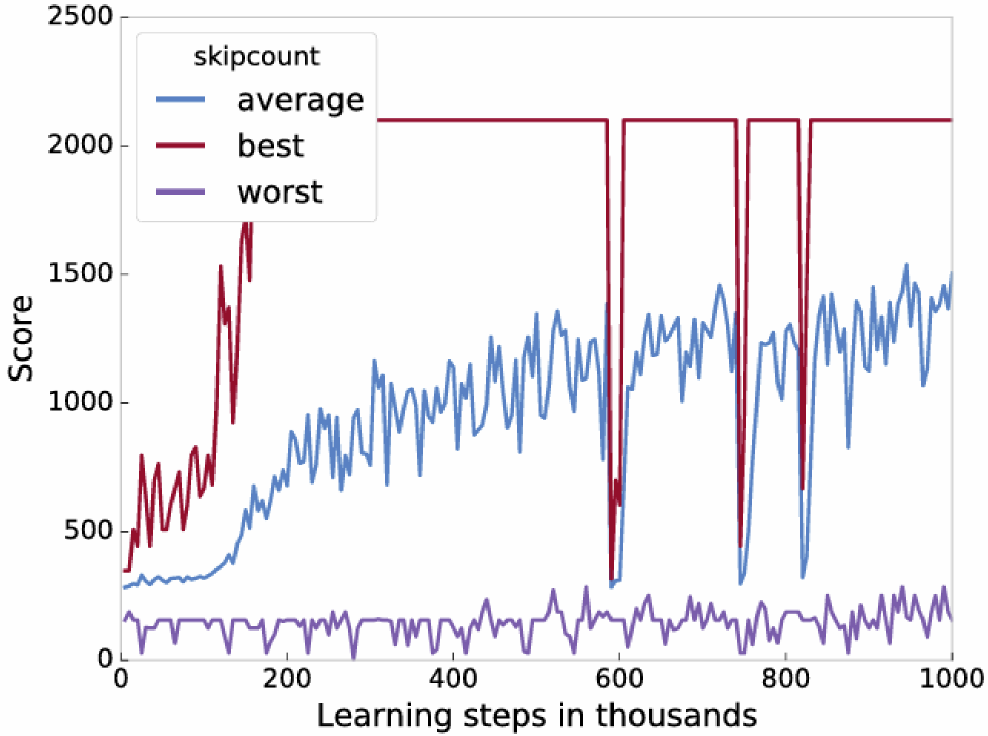
skipcount	average score ± stdev native	0	10	episodes	learning time [min]
0	51.5 ± 74.9	51.5 ± 74.9	36.0 ± 103.6	6961	91.1
1	69.0 ± 34.2	69.2 ± 26.9	39.6 ± 93.9	29 378	93.1
2	76.2 ± 15.5	71.8 ± 18.1	47.9 ± 47.6	49 308	91.5
3	76.1 ± 14.6	75.1 ± 15.0	44.1 ± 85.4	65 871	93.4
4	82.2 ± 9.4	81.3 ± 11.0	76.5 ± 17.1	104 796	93.9
5	81.8 ± 10.2	79.0 ± 13.6	75.2 ± 19.9	119 217	92.5
6	81.5 ± 9.6	78.7 ± 14.8	76.3 ± 16.5	133 952	92
7	81.2 ± 9.7	77.6 ± 15.8	76.9 ± 17.9	143 833	95.2
10	80.1 ± 10.5	75.0 ± 17.6	80.1 ± 10.5	171 070	92.8
15	74.6 ± 14.5	71.2 ± 16.0	73.5 ± 19.2	185 782	93.6
20	74.2 ± 15.0	73.3 ± 14.0	71.4 ± 20.7	240 956	94.8
25	73 ± 17	73.6 ± 15.5	71.4 ± 20.8	272 633	96.9
30	61.4 ± 31.9	69.7 ± 19.0	68.9 ± 24.2	265 978	95.7
35	60.2 ± 32.2	69.5 ± 16.6	65.7 ± 26.1	299 545	96.9
40	56.2 ± 39.7	68.4 ± 19.0	68.2 ± 22.8	308 602	98.6

TEXAS
The University of Texas at Austin
WHAT STARTS HERE CHANGES THE WORLD

Medkit Experiment

- More complex
- Maze with constant damage encouraging strong pathfinding
- Rewards Defined as follows
  - 1 point per tick survived
  - -100 for dying
- Same learning strategies with additional convolution layer and more parameters

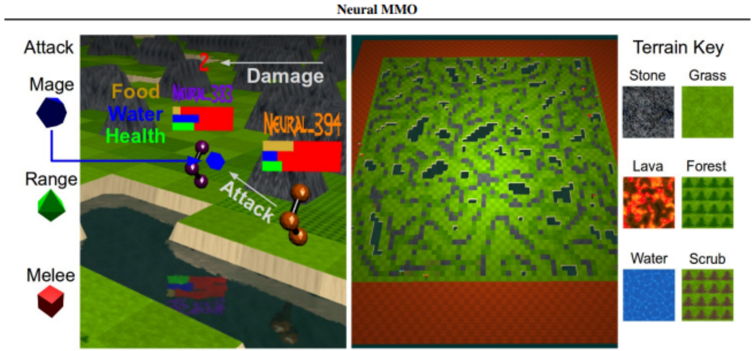



skipcount
average
best
worst
Score
2500
2000
1500
1000
500
0
0
200
400
600
800
1000
Learning steps in thousands

Impact on RL

- Became widely adopted as a research platform (600+ citations)
- Annual competitions at major conferences drove algorithm development
- Used in 100+ papers on navigation, multi-agent RL, curiosity-driven exploration
- First platform to systematically test 3D navigation from pixels
- Bridged the gap between toy 2D problems and real-world robotics, inspiring later platforms like DeepMind Lab, AI2-THOR, and Habitat

Future Testbeds



Questions?