Promptable Game Models

Talk at 3DVSS 2024 (31.05.2024)

Vladislav Golyanik

golyanik@mpi-inf.mpg.de
Modern Video Game Development

Horizon Zero Down Prototypes

Elden Ring 3D Model Showcase

Video sources: https://www.youtube.com/watch?v=h9tLcD1r-6w; https://www.youtube.com/watch?v=WoqDjsiRmyA
Example: TopSpin 2k25
Modern Video Game Development

Example: TopSpin 2k25 (2024)

Marker-based Motion Capture (Roger Federer)

Video sources: https://www.youtube.com/watch?v=tSS1M7oHnKM; https://www.youtube.com/watch?v=jKXj6QpGgGQ
Modern Video Game Development

Screenshots of *Top Spin 2K25* (2024)

...is extremely expensive (1-10M€ cost range)
- Software licenses (1k-10k€/year)
- Professional multi-camera systems (1k-10k€ per camera)
- HPC system with TBs of storage (>>10k€)
- Professional actors or players (10-100€/h)

Examples (leading and award-winning games):
- Battlefield 2042: €2B
- Elden Ring: €190M
- Horizon Zero Down: 100M€
- Marvel’s Spider Man: €95M
Neural Video Game Simulation


- Vast software ecosystems
- Extensible and reusable software
- Organised into multiple components
  - Rendering engine
  - Resource manager
  - Animation manager
  - Gameplay foundation system
    (game rules and AI/logic)

Classical Game Engines

References (left): Gregory, 2018, Müller et al., 2020.
Neural Video Game Simulation

- Vast software ecosystems
- Extensible and reusable software
- Organised into multiple components
  - Rendering engine
  - Resource manager
  - Animation manager
  - Gameplay foundation system (game rules and AI/logic)

Neural Game Simulation

- New Research Trend: Video game simulation using NN
- Objective: To train NN to synthesise videos based on prompts
- Games as an evolution of an environment driven by the actions of its agents
- Current SotA with discrete actions:
  - Learning discrete action representation [Menapace et al., 2022]
  - Actions as a learned set of geometric transformations [Huang et al., 2022]
  - Separating actions into a global shift and a discrete action components [Davtyan and Favaro 2022]

References (left): Gregory, 2018, Müller et al., 2020.
Playable Environments

Control Style

Control Players

Control Camera

Playable Environments

Control Style

Control Players

Player Control

Camera Control

Style Control

Vladislav Golyanik

Playable Environments

Playable Environments

Pros

- Can generate novel views
- Does not require action label in the data
- Represents complex 3D scenes (NeRF renderer)

Menapace et al., 2022.
**Playable Environments**

Pros
- Can generate novel views
- Does not require action label in the data
- Represents complex 3D scenes (NeRF renderer)

Cons
- Learns discrete action representation (no semantic control)
- Auto-regressive generation conditioned on labels: Does not support prompts for constraint- or goal-driven generation
- Adversarially trained LSTM animation module
- Comparably low image resolution/checkerboard artifacts
- Does not support small objects / human details
- Compositional NeRF is not efficient

Menapace et al., 2022.
**Enabling Fine-grained Control**

**Limitation:** Discrete action representation does not allow semantic control.

**Motivation:** We are interested in fine-grained constraint and goal-driven generation!
Enabling Fine-grained Control

**Motivation:** We are interested in fine-grained constraint and goal-driven generation!

A possible way to enable it: Game models augmented with prompts specified as a set of natural language actions and desired states.

“hit the ball with a backhand and send it to the right service box”
“the [other] player does not catch the ball”

**Limitation:** Discrete action representation does not allow semantic control.
Enabling Fine-grained Control

Limitation: Discrete action representation does not allow semantic control.

Motivation: We are interested in fine-grained constraint and goal-driven generation!

A possible way to enable it: Game models augmented with prompts specified as a set of natural language actions and desired stated.

Play the game →

“hit the ball with a backhand and send it to the right service box”

“the [other] player does not catch the ball”
Improved Rendering

Limitations of PE related to the rendering scene quality:
Low image resolution, checkerboard artefacts, low quality for small objects and details, slow/inefficient compositional NeRF.

[Tretschk et al., 2021]

[Fridovich-Keil et al., 2022]

[Weng et al., 2022]
Related Works

[Holden et al., 2020]

[Starke et al., 2019]

[Zhang et al., 2024]

A person walks forward, then bends down.

[Dabral et al., 2023]

[Zhang et al., 2024]

Learned Character Animation / Text-driven Generation
Related Works

Learned Character Animation / Text-driven Generation

[Holden et al., 2020]

[Starke et al., 2019]

[Dabral et al., 2023]

[Zhang et al., 2024]

[Starke et al., 2019]

[Zhang et al., 2024]

[Ghosh et al., 2023]
Promptable Game Models (PGMs): Text-guided Game Simulation via Masked Diffusion Models
"The player does not catch the ball"
Overview: PGM as a State Machine

1) models the game dynamics: player actions and interactions in the space of the environment states (evolution of the environment in time)

2) generates an image given the environment state (image renderer)
Overview: PGM as a State Machine

1) models the game dynamics: player actions and interactions in the space of the environment states (evolution of the environment in time)

2) generates an image given the an environment state (image renderer)
Overview: Control and Training

Two ways of controlling through prompts:
- Explicit manipulation or
- High-level text-based editing.

Example: Change of the tennis ball

Example: "The player takes several steps to the right and hits the ball with a backhand"

High-level, yet fine-grained control over the evolution of the environment.

Training: A dataset of camera-calibrated videos with per-frame annotations (s and a).
PGMs: Fine-grained Control

Different predicted sequences starting from the same initial state and altering the text conditioning. The model supports fine-grained control over the various tennis shots using technical terms (e.g., “forehand”, “backhand”, “volley”).
PGMs: Fine-grained Control

Different predicted sequences starting from the same initial state and altering the text conditioning. The model supports fine-grained control over the various tennis shots using technical terms (e.g., “forehand”, “backhand”, “volley”).
Animation Module (AM)

conditioning values:

- $s^c \in \mathbb{S}^T$
- text $a^c \in L^{A \times T}$
- $m^s \in \{0, 1\}^{P \times T}$
- $m^a \in \{0, 1\}^{A \times T}$

**Temporal model** based on non-autoregressive transformer pre-trained LM in a **text encoder** to model action conditioning information

$$a^{emb} = \mathcal{T}(a^c) \in \mathbb{R}^{A \times T \times N_t}$$

[Raffel et al., 2022]
Animation Module (AM)

conditioning values:
\( s^c \in \mathbb{S}^T \)
\( m^s \in \{0, 1\}^{P \times T} \)
\( m^a \in \{0, 1\}^{A \times T} \)

Text \( a^c \in \mathbb{L}^{A \times T} \)

Pre-trained LM in a text encoder to model action conditioning information

\[ a^{\text{emb}} = T(a^c) \in \mathbb{R}^{A \times T \times N_t} \]

[Raffel et al., 2022]

Temporal model based on non-autoregressive transformer

AM predicts \( s^p = s_0^p \) as a progressive denoising process \( s_0^p, ..., s_K^p \).
Animation Module (AM)

conditioning on $k$ through weight demodulation

conditioning values:

- $s^c \in \mathbb{S}^T$ 
- $a^c \in \mathbb{L}^{A \times T}$
- $m^s \in \{0, 1\}^{P \times T}$
- $m^a \in \{0, 1\}^{A \times T}$

sampled according to various strategies emulating desired inference tasks

Temporal model based on
non-autoregressive transformer
pre-trained LM in a text encoder to model action conditioning information

$$a^{\text{emb}} = \mathcal{T}(a^c) \in \mathbb{R}^{A \times T \times N_t}$$

[Raffel et al., 2022]

AM predicts $s^p = s^p_0$ as a progressive denoising process $s^p_0, ..., s^p_K$. 

(Raffel et al., 2022)
Animation Module (AM)

conditioning on $k$ through weight demodulation

conditioning values:
- $s^c \in \mathbb{S}^T$
- text $a^c \in \mathbb{L}^{A \times T}$
- $m^s \in \{0, 1\}^{P \times T}$
- $m^a \in \{0, 1\}^{A \times T}$

sampled according to various strategies emulating desired inference tasks

Temporal model based on non-autoregressive transformer

pre-trained LM in a text encoder to model action conditioning information

\[ a^{\text{emb}} = \mathcal{T}(a^c) \in \mathbb{R}^{A \times T \times N_t} \]

[Raffel et al., 2022]

AM predicts $s^p = s^p_0$ as a progressive denoising process $s^p_0, ..., s^p_K$.

$\mathcal{A}$ acts as a noise estimator predicting Gaussian noise $\epsilon_k$ in the noisy sequence of unknown states $s^p_k$: $\epsilon^p_k = \mathcal{A}(s^p_k | s^c, a^{\text{emb}}, m^s, m^a, k)$. 
Animation Module (AM)

Minimising the DDPM training objective [Ho et al., 2020]:

$$
\mathbb{E}_{k \sim \mathcal{U}(1,K), \epsilon \sim \mathcal{N}(0,I)} \| \epsilon_k^p - \epsilon_k \|
$$

Training details:
- ADAM optimiser [Kingma and Ba, 2015]
- LR of 10e-4
- Cosine schedule
- 10k warm-up steps
- 2.5M training steps in total
- batch size of 32
- T = 16
- K = 1000
- Linear noise schedule
Synthesis Module (SM)

- Coarsely follows Playable Environments [Menapace et al., 2022]
Synthesis Module (SM)

Point in the deformed ray space

Deformation Model [Weng et al., 2022]

Canonical Pose

- Coarsely follows Playable Environments [Menapace et al., 2022]

Composition of independent objects
(parametrised with voxel grids) + fully-opaque planes
[Fridovich-Keil et al., 2022]
Synthesis Module (SM)

- Coarsely follows Playable Environments [Menapace et al., 2022]
- Composition of independent objects (parametrised with voxel grids) + fully-opaque planes [Fridovich-Keil et al., 2022]

- Feature Enhancer (CNN): G in, an RGB image out
- Ray Casting
- Feature Grid
- Style Encoder
- Canonical Pose
- Deformation Model [Weng et al., 2022]
- Point in the deformed ray space
- Ground Truth
- Reconstruction Losses
- Action a
- Pose
- Location
- Velocity
- Style ω
- Camera
- Framerate V
SM: Object-specific Rendering

Tennis scenes with and without inserted rackets.

Ball rendering

shutter opens

d
shutter closes

x-axis

Tennis scenes with and without inserted rackets.
Synthesis Module (SM)

Imposed on samples image patches:
- L2 reconstruction loss
- Perceptual loss [Johnson et al. 2016]

Training details:
- ADAM optimiser [Kingma and Ba, 2015]
- LR of 10e-4, exponential decrease to 10e-5
- 10k warm-up steps
- 300k training steps in total
- Videos of 1024x576px resolution
- 180x180px patch resolution
Applications
Unconditional Sequence Generation

The player moves to the left corner waiting for the serve

The player serves the ball to the left corner of the field
Application: Opponent Modelling

response by running to the right (top)

- "The player runs to the right and performs another backhand to the left side of the field"
- "The player moves slightly to the left"
- "The player takes steps to the right and sends the ball to the right side of no man's land with a backhand"
- "The player starts moving to the left"

Game AI Actions (Bottom player)
Game AI Actions (Top player)

response by running towards the net (bottom)

- "The player jumps to the left and sends the ball to the left part of the service line with a backhand"
- "The player jumps diagonally backwards to the left and waits for the ball"
- "The player moves forward and sends the ball to the right service box with a backhand"
- "The player moves to the right for the hit but the game is ended"
Action-conditioned Sequence Generation

Initial State

The player serves to the right service box

The player jumps forward and waits for the ball
While in the original sequence the bottom player aims its response to the center of the field where the opponent is waiting, the model now successfully generates a winning set of moves for the bottom player that sends the ball along the left sideline, too far for the top player to be reached.
“The [top] player does not catch the ball”

original video = bottom player loses

Example 1

The player rushes diagonally to the upper right and hits the ball with a forehand to the net
“The [top] player does not catch the ball”

1/2 original video +
*text prompt*= bottom player wins

Example 1
“The [top] player does not catch the ball”

the man smashes the ball with the forehand to the right service box

original video = bottom player loses

Example 2

The player steps to the right and stops unable to save the ball
“The [top] player does not catch the ball”

1/2 original video + text prompt = bottom player wins

Example 2
Style Swap
Tennis and Minecraft Datasets
Tennis and Minecraft Dataset

Tennis dataset (broadcast tennis matches):
- 7.1k video sequences (1920x1080px at 25 fps)
- 15.5h
- 1.12M fully-annotated frames
- 25.5k unique captions and 915 unique words
Tennis and Minecraft Dataset

Minecraft dataset (from the video game):
- 61 videos (1024x576px at 20fps)
- 1.21h
- 68.5k fully-annotated frames
- 1.24k unique captions with 117 unique words
Tennis and Minecraft Dataset

- The player serves and sends the ball to the right service box.

- The player moves to the right and hits the ball with a forehand to the no man's land.

- The player sprints and jumps on the first block of the second area.
Annotation and Training Costs

Full model:
- Eight A100 (40GB Global Memory)
- Tennis dataset: Four days (844$)
- Minecraft dataset: Two days (422$)

Reduced model:
- Four A100 (40GB Global Memory)
- Tennis dataset: Three days (317$)
- Minecraft dataset: Two days (211$)

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Resolution</th>
<th>Frames Per Second</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tennis</td>
<td>1920x1080</td>
<td>25fps</td>
</tr>
<tr>
<td></td>
<td>1024x576</td>
<td>20fps</td>
</tr>
<tr>
<td></td>
<td>1080p</td>
<td>30fps</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Captions</th>
<th>Unique Words</th>
<th>Avg. Words</th>
<th>Avg. Span</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tennis</td>
<td>84.1k</td>
<td>25.5k</td>
<td>11.8k</td>
<td>13.3k</td>
</tr>
<tr>
<td></td>
<td>81.6k</td>
<td>22.4k</td>
<td>11.6k</td>
<td>13.3k</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Nouns</th>
<th>Verbs</th>
<th>Adjectives</th>
<th>Adverbs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tennis</td>
<td>32.3%</td>
<td>11.9%</td>
<td>3.08%</td>
<td>2.79%</td>
</tr>
<tr>
<td></td>
<td>36.2%</td>
<td>17.4%</td>
<td>6.48%</td>
<td>11.7%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Pronouns</th>
<th>Articles</th>
<th>Prepositions</th>
<th>Numerals</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tennis</td>
<td>0.18%</td>
<td>26.4%</td>
<td>7.89%</td>
<td>0.11%</td>
</tr>
<tr>
<td></td>
<td>0.00%</td>
<td>8.03%</td>
<td>6.98%</td>
<td>0.00%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Particle</th>
<th>Punctuation</th>
<th>Others</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tennis</td>
<td>9.28%</td>
<td>1.76%</td>
<td>0.00%</td>
</tr>
<tr>
<td></td>
<td>1.50%</td>
<td>1.12%</td>
<td>0.00%</td>
</tr>
</tbody>
</table>

Tennis dataset:
- Professional labelling team (833$/h)
- Initial annotation: 13.5k$ in total
- Comparable amount for remaining annotation and training (development)
Experimental Results
Quantitative Results (AM and SM)

### Tennis

<table>
<thead>
<tr>
<th></th>
<th>Position L2</th>
<th>Position FD</th>
<th>Root angle L2</th>
<th>Root angle FD</th>
<th>Joints 3D L2</th>
<th>Joints 3D FD</th>
</tr>
</thead>
<tbody>
<tr>
<td>PE</td>
<td>3.291</td>
<td>229.112</td>
<td>1.126</td>
<td>15.953</td>
<td>0.303</td>
<td>53.242</td>
</tr>
<tr>
<td>Rec. LSTM</td>
<td>1.597</td>
<td>7.253</td>
<td>0.907</td>
<td>7.051</td>
<td>0.193</td>
<td>16.735</td>
</tr>
<tr>
<td>Rec. Transf.</td>
<td>1.074</td>
<td>4.402</td>
<td>0.767</td>
<td>6.838</td>
<td>0.175</td>
<td>14.845</td>
</tr>
<tr>
<td>Ours Small</td>
<td>1.380</td>
<td>1.443</td>
<td>1.014</td>
<td>0.560</td>
<td>0.148</td>
<td>1.253</td>
</tr>
<tr>
<td>Ours</td>
<td>1.099</td>
<td>0.929</td>
<td>0.844</td>
<td>0.356</td>
<td>0.129</td>
<td>0.836</td>
</tr>
</tbody>
</table>

### Minecraft

<table>
<thead>
<tr>
<th></th>
<th>Position L2</th>
<th>Position FD</th>
<th>Root angle L2</th>
<th>Root angle FD</th>
<th>Joints 3D L2</th>
<th>Joints 3D FD</th>
</tr>
</thead>
<tbody>
<tr>
<td>PE</td>
<td>2.739</td>
<td>105.973</td>
<td>1.620</td>
<td>31.232</td>
<td>0.311</td>
<td>39.572</td>
</tr>
<tr>
<td>Rec. LSTM</td>
<td>2.292</td>
<td>47.296</td>
<td>1.702</td>
<td>49.971</td>
<td>0.489</td>
<td>99.843</td>
</tr>
<tr>
<td>Rec. Transf.</td>
<td>2.154</td>
<td>53.198</td>
<td>1.430</td>
<td>36.123</td>
<td>0.385</td>
<td>69.977</td>
</tr>
<tr>
<td>Ours Small</td>
<td>1.084</td>
<td>4.461</td>
<td>1.077</td>
<td>6.016</td>
<td>0.140</td>
<td>3.590</td>
</tr>
<tr>
<td>Ours</td>
<td>1.065</td>
<td>4.815</td>
<td>0.956</td>
<td>4.083</td>
<td>0.132</td>
<td>3.360</td>
</tr>
</tbody>
</table>

*results averaged over all tasks

### Animation Module

- Video prediction conditioned on actions
- Unconditioned video prediction

### Synthesis Module

- Opponent modeling
- Sequence completion

<table>
<thead>
<tr>
<th></th>
<th>LPIPS</th>
<th>FID</th>
<th>FVD</th>
<th>ADD</th>
<th>MDR</th>
</tr>
</thead>
<tbody>
<tr>
<td>PE [Menapace et al. 2022]</td>
<td>0.188</td>
<td>11.5</td>
<td>349</td>
<td>3.74</td>
<td>0.200</td>
</tr>
<tr>
<td>PE+ [Menapace et al. 2022]</td>
<td>0.232</td>
<td>40.4</td>
<td>2432</td>
<td>132.3</td>
<td>49.7</td>
</tr>
<tr>
<td>w/o enhancer $F$</td>
<td>0.167</td>
<td>15.6</td>
<td>570</td>
<td>3.02</td>
<td>0.0728</td>
</tr>
<tr>
<td>w/o explicit deformation in $D$</td>
<td>0.156</td>
<td>13.3</td>
<td>524</td>
<td>3.10</td>
<td>0.0587</td>
</tr>
<tr>
<td>w/o planes in $C$</td>
<td>0.241</td>
<td>30.4</td>
<td>1064</td>
<td>2.94</td>
<td>0.0611</td>
</tr>
<tr>
<td>w/o voxels in $C$</td>
<td>0.170</td>
<td>17.1</td>
<td>757</td>
<td>3.03</td>
<td>0.0399</td>
</tr>
<tr>
<td>w/o our encoder $E$</td>
<td>0.174</td>
<td>15.0</td>
<td>600</td>
<td>3.18</td>
<td>0.0564</td>
</tr>
<tr>
<td>Ours Small</td>
<td>0.156</td>
<td>13.4</td>
<td>523</td>
<td>2.88</td>
<td>0.0470</td>
</tr>
<tr>
<td>Ours</td>
<td>0.152</td>
<td>12.8</td>
<td>516</td>
<td>2.88</td>
<td>0.0423</td>
</tr>
<tr>
<td>PE [Menapace et al. 2022]</td>
<td>0.0235</td>
<td>13.9</td>
<td>21.5</td>
<td>5.77</td>
<td>0.0412</td>
</tr>
<tr>
<td>PE+ [Menapace et al. 2022]</td>
<td>0.0238</td>
<td>15.5</td>
<td>51.7</td>
<td>120.6</td>
<td>0.939</td>
</tr>
<tr>
<td>Ours Small</td>
<td>0.00996</td>
<td>3.56</td>
<td>8.83</td>
<td>2.02</td>
<td>0.0529</td>
</tr>
<tr>
<td>Ours</td>
<td>0.00814</td>
<td>2.81</td>
<td>7.08</td>
<td>1.98</td>
<td>0.0508</td>
</tr>
</tbody>
</table>
### Comparison to PE and Ablation Study

<table>
<thead>
<tr>
<th>PE</th>
<th>baselines</th>
<th>PE+</th>
<th>ours small</th>
<th>our method</th>
<th>ours</th>
<th>ground truth</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Image" /></td>
<td><img src="image2.png" alt="Image" /></td>
<td><img src="image3.png" alt="Image" /></td>
<td><img src="image4.png" alt="Image" /></td>
<td><img src="image5.png" alt="Image" /></td>
<td><img src="image6.png" alt="Image" /></td>
<td><img src="image7.png" alt="Image" /></td>
</tr>
<tr>
<td>ours w/o enhancer $\mathcal{E}$</td>
<td>ours w/o explicit deformation in $\mathcal{D}$</td>
<td>ablations</td>
<td>ours w/o planes in $\mathcal{C}$</td>
<td>ours w/o voxels in $\mathcal{C}$</td>
<td>ours w/o our encoder $\mathcal{C}$</td>
<td></td>
</tr>
</tbody>
</table>

PGM generates sharper players and static scene elements. PE and PE+ produce checkerboard artifacts.
The model uses the first-frame object properties and all actions as conditioning.
### Evaluation Breakdown and Other Tests

- Robustness to prompt variations
- AM (Inference Tasks)
- AM Masking Strategies Ablation
- AM Dataset Size Ablation
- Alternative Samplers

---

**Table: Action conditioned video prediction**

<table>
<thead>
<tr>
<th></th>
<th>Position L2</th>
<th>Position FD</th>
<th>Root angle L2</th>
<th>Root angle FD</th>
<th>Joints 3D L2</th>
<th>Joints 3D FD</th>
</tr>
</thead>
<tbody>
<tr>
<td>PE</td>
<td>3.117</td>
<td>87.688</td>
<td>1.822</td>
<td>12.627</td>
<td>0.277</td>
<td>30.711</td>
</tr>
<tr>
<td>Rec. LSTM</td>
<td>1.753</td>
<td>7.413</td>
<td>1.100</td>
<td>8.416</td>
<td>0.234</td>
<td>18.455</td>
</tr>
<tr>
<td>Rec. Transf.</td>
<td>1.183</td>
<td>2.996</td>
<td>0.913</td>
<td>7.566</td>
<td>0.212</td>
<td>15.976</td>
</tr>
<tr>
<td>Ours Small</td>
<td>1.244</td>
<td>1.071</td>
<td>1.187</td>
<td>6.010</td>
<td>0.178</td>
<td>1.570</td>
</tr>
<tr>
<td>Ours</td>
<td>1.064</td>
<td>0.846</td>
<td>0.961</td>
<td>0.421</td>
<td>0.153</td>
<td>1.049</td>
</tr>
</tbody>
</table>

**Table: Unconditional video prediction**

<table>
<thead>
<tr>
<th></th>
<th>Position L2</th>
<th>Position FD</th>
<th>Root angle L2</th>
<th>Root angle FD</th>
<th>Joints 3D L2</th>
<th>Joints 3D FD</th>
</tr>
</thead>
<tbody>
<tr>
<td>PE</td>
<td>3.973</td>
<td>146.019</td>
<td>1.604</td>
<td>30.448</td>
<td>0.437</td>
<td>78.835</td>
</tr>
<tr>
<td>Rec. LSTM</td>
<td>2.064</td>
<td>11.283</td>
<td>1.224</td>
<td>14.860</td>
<td>0.264</td>
<td>28.736</td>
</tr>
<tr>
<td>Rec. Transf.</td>
<td>1.649</td>
<td>10.514</td>
<td>1.123</td>
<td>15.648</td>
<td>0.251</td>
<td>27.258</td>
</tr>
<tr>
<td>Ours Small</td>
<td>2.352</td>
<td>2.271</td>
<td>1.455</td>
<td>0.781</td>
<td>0.213</td>
<td>1.827</td>
</tr>
<tr>
<td>Ours</td>
<td>1.925</td>
<td>1.377</td>
<td>1.277</td>
<td>0.518</td>
<td>0.192</td>
<td>1.261</td>
</tr>
</tbody>
</table>

**Table: Opponent modeling**

<table>
<thead>
<tr>
<th></th>
<th>Position L2</th>
<th>Position FD</th>
<th>Root angle L2</th>
<th>Root angle FD</th>
<th>Joints 3D L2</th>
<th>Joints 3D FD</th>
</tr>
</thead>
<tbody>
<tr>
<td>PE</td>
<td>4.353</td>
<td>641.976</td>
<td>0.903</td>
<td>13.955</td>
<td>0.251</td>
<td>62.981</td>
</tr>
<tr>
<td>Rec. LSTM</td>
<td>1.581</td>
<td>5.507</td>
<td>0.697</td>
<td>2.517</td>
<td>0.143</td>
<td>10.443</td>
</tr>
<tr>
<td>Rec. Transf.</td>
<td>1.169</td>
<td>3.735</td>
<td>0.635</td>
<td>2.514</td>
<td>0.138</td>
<td>10.519</td>
</tr>
<tr>
<td>Ours Small</td>
<td>1.578</td>
<td>2.243</td>
<td>0.832</td>
<td>0.560</td>
<td>0.114</td>
<td>0.851</td>
</tr>
<tr>
<td>Ours</td>
<td>1.153</td>
<td>1.349</td>
<td>0.703</td>
<td>0.288</td>
<td>0.101</td>
<td>0.558</td>
</tr>
</tbody>
</table>

**Table: Sequence completion**

<table>
<thead>
<tr>
<th></th>
<th>Position L2</th>
<th>Position FD</th>
<th>Root angle L2</th>
<th>Root angle FD</th>
<th>Joints 3D L2</th>
<th>Joints 3D FD</th>
</tr>
</thead>
<tbody>
<tr>
<td>PE</td>
<td>1.720</td>
<td>40.766</td>
<td>0.814</td>
<td>6.783</td>
<td>0.246</td>
<td>40.441</td>
</tr>
<tr>
<td>Rec. LSTM</td>
<td>0.990</td>
<td>4.809</td>
<td>0.606</td>
<td>2.411</td>
<td>0.132</td>
<td>9.305</td>
</tr>
<tr>
<td>Rec. Transf.</td>
<td>0.294</td>
<td>0.364</td>
<td>0.486</td>
<td>1.625</td>
<td>0.100</td>
<td>5.628</td>
</tr>
<tr>
<td>Ours Small</td>
<td>0.344</td>
<td>0.187</td>
<td>0.581</td>
<td>0.301</td>
<td>0.088</td>
<td>0.765</td>
</tr>
<tr>
<td>Ours</td>
<td>0.252</td>
<td>0.143</td>
<td>0.437</td>
<td>0.198</td>
<td>0.069</td>
<td>0.478</td>
</tr>
</tbody>
</table>

**Table: Average**

<table>
<thead>
<tr>
<th></th>
<th>Position L2</th>
<th>Position FD</th>
<th>Root angle L2</th>
<th>Root angle FD</th>
<th>Joints 3D L2</th>
<th>Joints 3D FD</th>
</tr>
</thead>
<tbody>
<tr>
<td>PE</td>
<td>3.291</td>
<td>229.112</td>
<td>1.126</td>
<td>15.953</td>
<td>0.303</td>
<td>53.242</td>
</tr>
<tr>
<td>Rec. LSTM</td>
<td>1.597</td>
<td>7.253</td>
<td>0.907</td>
<td>7.651</td>
<td>0.193</td>
<td>16.735</td>
</tr>
<tr>
<td>Rec. Transf.</td>
<td>1.074</td>
<td>4.402</td>
<td>0.767</td>
<td>6.838</td>
<td>0.175</td>
<td>14.845</td>
</tr>
<tr>
<td>Ours Small</td>
<td>1.380</td>
<td>1.443</td>
<td>1.014</td>
<td>0.560</td>
<td>0.148</td>
<td>1.253</td>
</tr>
<tr>
<td>Ours</td>
<td>1.099</td>
<td>0.929</td>
<td>0.844</td>
<td>0.356</td>
<td>0.129</td>
<td>0.836</td>
</tr>
</tbody>
</table>
Implausible Actions

"The player sidesteps to the right and performs a forehand that sends the ball to the right side of no man's land."

"The player rushes to the left and hits with another forehand to the left side of no man's land."

"The player jumps on the oak pillar."

The left movement command is ignored.
Camera Manipulation

Original camera

Manipulated camera

Manipulated camera depth
Limitations

- No AM conditioning on scene geometry
- Tennis scenario: Overfitting with less than 60% of the data
- Foot sliding artefacts

- No explicit physics modelling (everything is learnt from data)
- Not real-time (AM: 1.08fps)
Conclusion and Take-home Messages

- Textual action representation is **crucial for unlocking fine-grained control** over the generation.
- PGMs outperforms previous PE approach in the rendering quality, generation of state sequences and obeying the conditioning signals (thanks to recent advances in ML and neural rendering).
  - DM in the animation module learns the multimodal distribution well.
- PGMs enable **compelling constraint-and goal-driven generation** applications (such as opponent modelling, state inpainting, game analysis).
- There are many possible **future extensions**.

Project page: snap-research.github.io/promptable-game-models/
Today’s Talk

Intern at 4DQV/MPI-INF, 2021-2022

With Willi Menapace (University of Trento), Aliaksandr Siarohin (Snap Inc.), Stéphane Lathuilière (LTCP, Télécom Paris, Institut Polytechnique de Paris), Panos Achlioptas (Snap Inc.), Sergey Tulyakov (Snap Inc.) and Elisa Ricci (University of Trento).

Project page: snap-research.github.io/promptable-game-models/

Menapace et al., arXiv:2303.13472

Promptable Game Models: Text-Guided Game Simulation via Masked Diffusion Models

WILLI MENAPACE, University of Trento, Italy
ALEKSEJ SIAROHIN, Snap Inc., USA
STÉPHANE LATHUILIÈRE, LTCI, Télécom Paris, Institut Polytechnique de Paris, France
PANOS ACCHILOPTAS, Snap Inc., USA
VLADIMIR GOLDENKIND, MPI for Informatics, SC, Germany
SERGEY TULYAKOV, Snap Inc., USA
ELISA RICCI, University of Trento, Fondazione Bruno Kessler, Italy

Fig. 1: We propose Promptable Game Models (PGMs), controllable models of games that are trained from scratch. Our PGMs model the production of (simulated) states, estimating a distribution of states using “gaming” strategies, such as plan-based, object localization, and detailed control (See text for details). A black-outlined frame highlights the region of interest (ROI) used in the proposed algorithm for computing next-best state and actions. A white-outlined frame highlights the region of interest (ROI) used in the proposed algorithm for computing next-best state and actions. A yellow-outlined frame highlights the region of interest (ROI) used in the proposed algorithm for computing next-best state and actions. A green-outlined frame highlights the region of interest (ROI) used in the proposed algorithm for computing next-best state and actions. A blue-outlined frame highlights the region of interest (ROI) used in the proposed algorithm for computing next-best state and actions.
Sec. 3 Fundamentals of Diffusion Models

Sec. 7 Towards 4D Spatio-temporal Diffusion

7.3 4D Scene Generation and Editing

Po, Wang, Golyanik et al. EUROGRAPHICS, 2024.
4DQV: Research Interests

3D/4D Reconstruction and Neural Rendering

4D Generative Models

Quantum CV

Images: Kappel et al., 2024, Shimada et al., 2023, Millerdurai et al., 2024, Shimada et al., 2024, Dabral et al., 2023, Seelbach Benkner et al., 2023, Bhatia et al., 2023.
Thanks! Questions?