max planck institut
informatik

3D VISION SUMMER SCHOOL

May 26 - June 1, 2024 - IIIT Bangalore, India (In Person)

Promptable Game Models

Talk at 3DVSS 2024 (31.05.2024)

Vladislav Golyanik

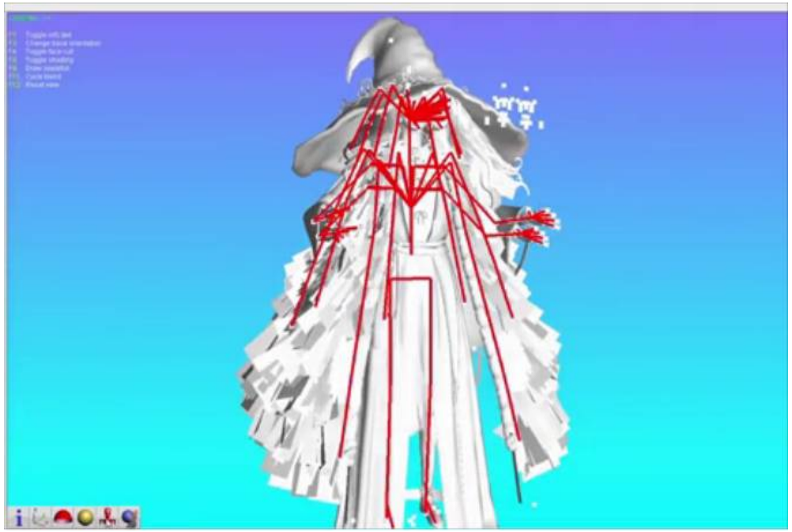
golyanik@mpi-inf.mpg.de

4D and Quantum
Vision Group

Visual Computing and
AI Department

Modern Video Game Development

Horizon Zero Down Prototypes



Elden Ring 3D Model Showcase

Modern Video Game Development

Example: TopSpin 2k25

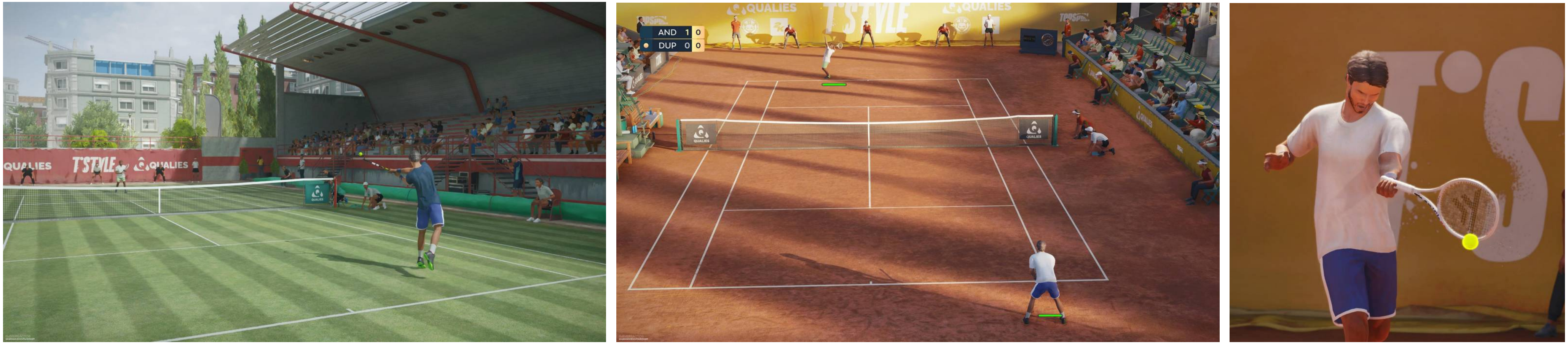
Modern Video Game Development

Example: TopSpin 2k25 (2024)

Marker-based Motion Capture
(Roger Federer)

Video sources: https://www.youtube.com/watch?v=tS51M7oHnKM; https://www.youtube.com/watch?v=jKXJ6QpPgGQ

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

Image source: https://www.gamereactor.de/top-spin-2k25-1303003/

Neural Video Game Simulation

max planck institut
informatik

Image: https://kevurugames.com/blog/best-game-engines-2022-pros-cons-and-top-picks-for-different-types-of-games/

- **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)

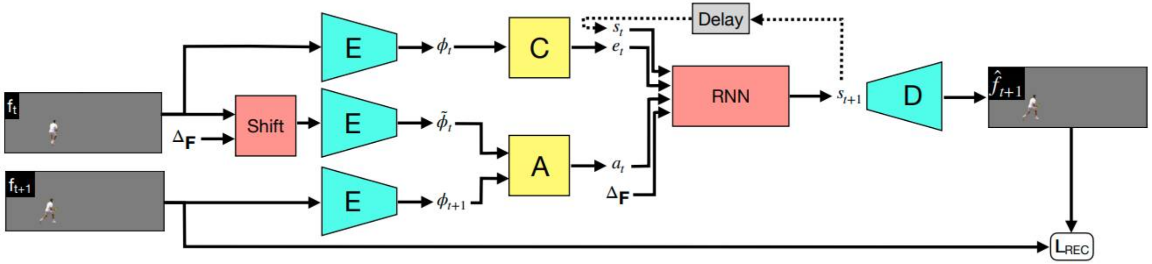
Classical Game Engines

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

Neural Video Game Simulation

max planck institut informatik

Image: https://kevurugames.com/blog/best-game-engines-2022-pros-cons-and-top-picks-for-different-types-of-games/



[Davtyan and Favaro 2022]

- **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)

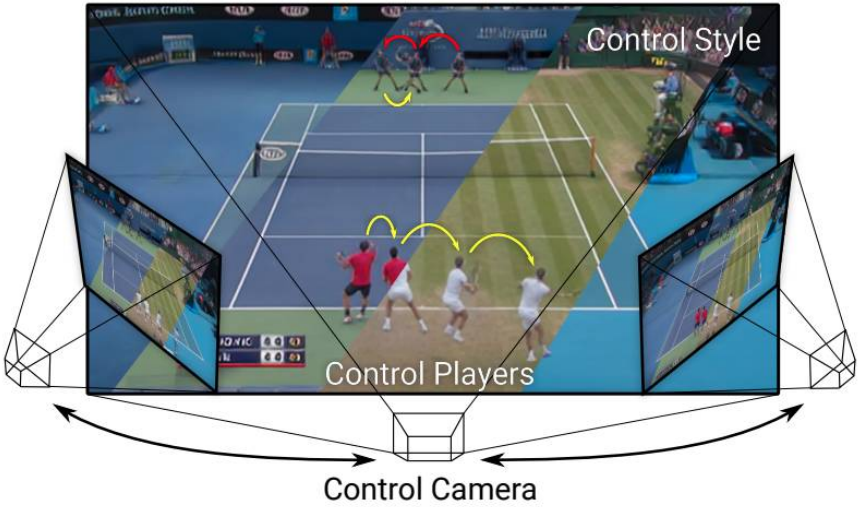
- **New Research Trend: Video game simulation using NN**
- Objective: To train NN to synthesise videos based on prompts
- Games as an evolution of an environment driven by the actions of its agents
- **Current SotA with discrete actions:**
  - Learning discrete action representation [Menapace et al., 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]

Classical Game Engines

Neural Game Simulation

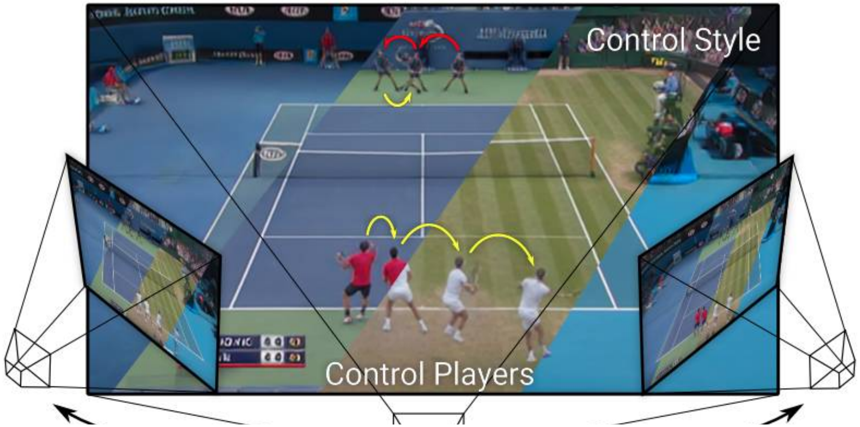
References (left): Gregory, 2018, Müller et al., 2020.

Playable Environments



Menapace et al, 2022.

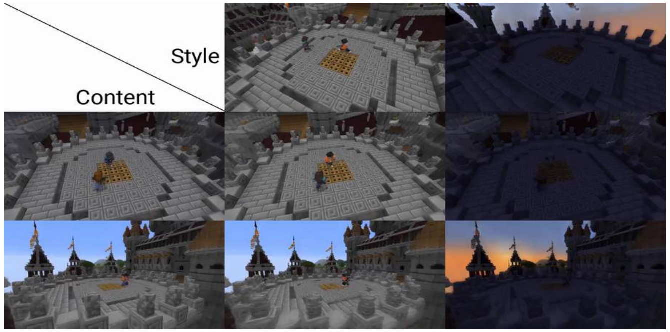
Playable Environments



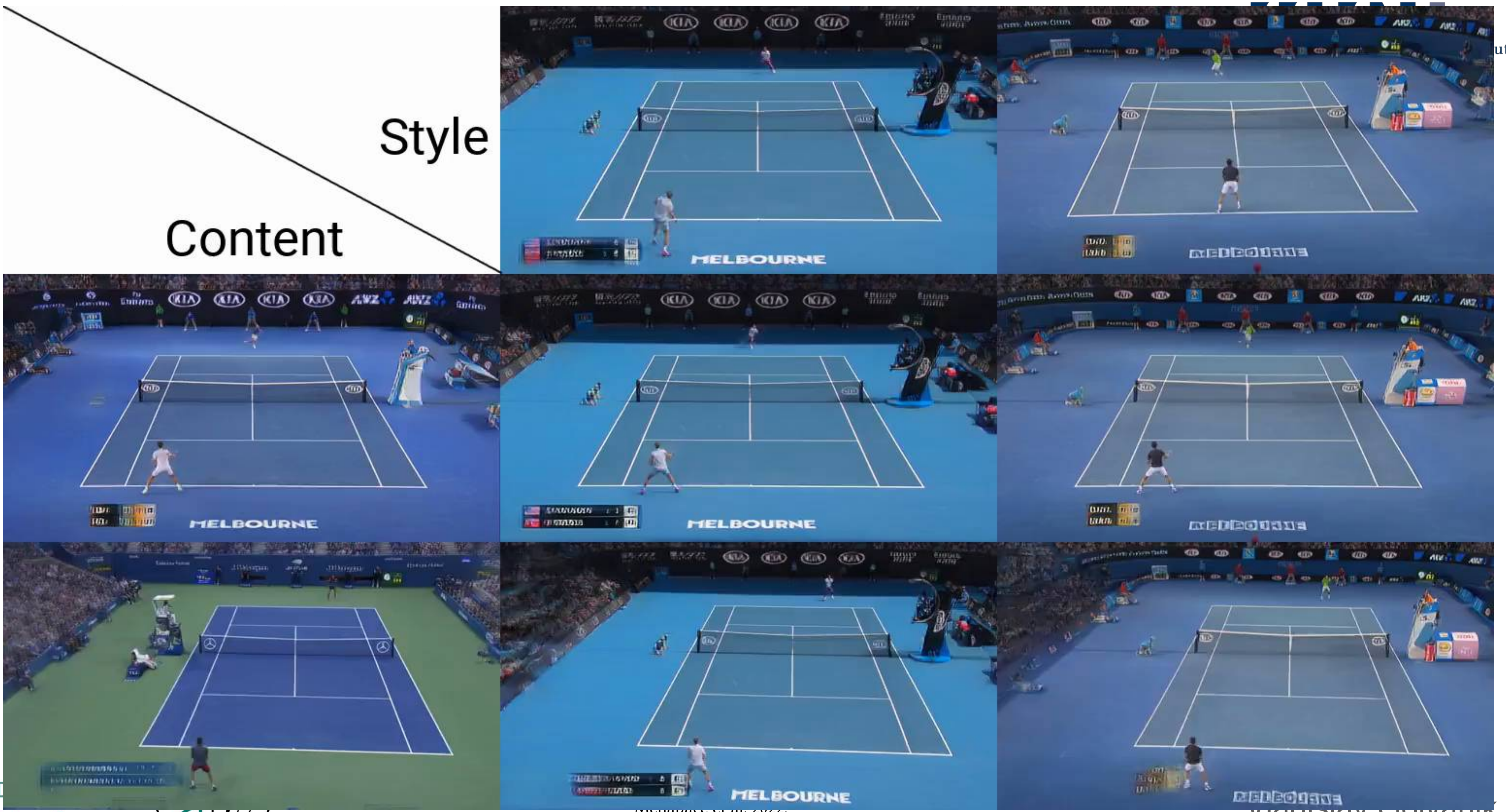
Player Control

Camera Control

Style Control

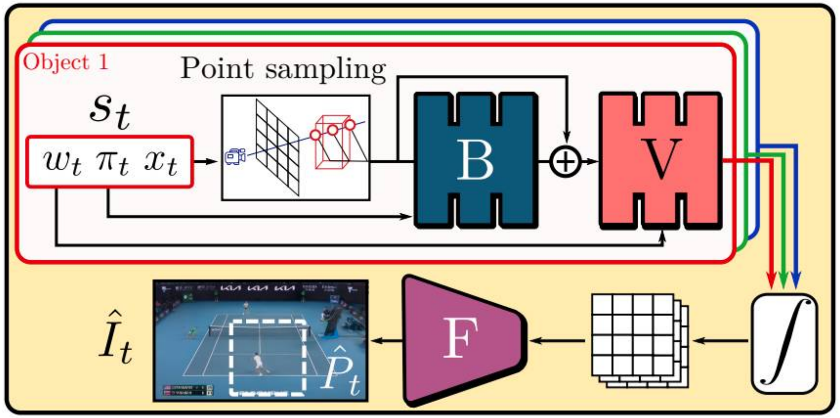


Menapace et al, 2022.

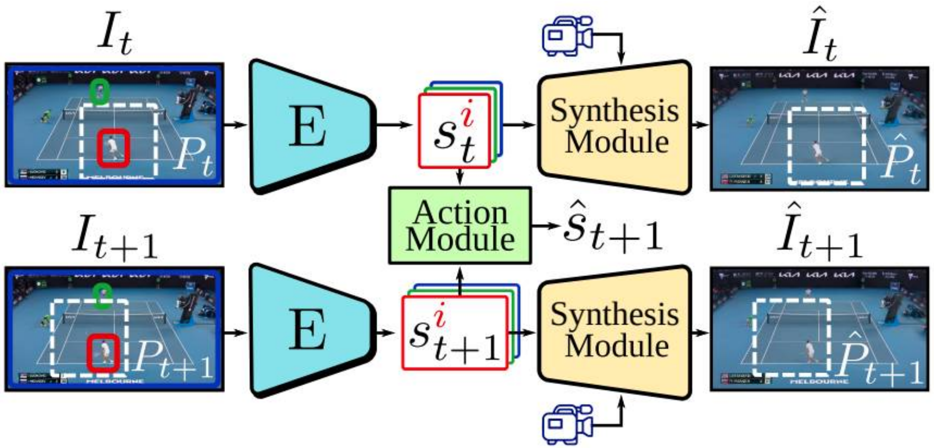

Style
Content
MELBOURNE

Menapace et al., 2022.

Playable Environments



Playable Environments



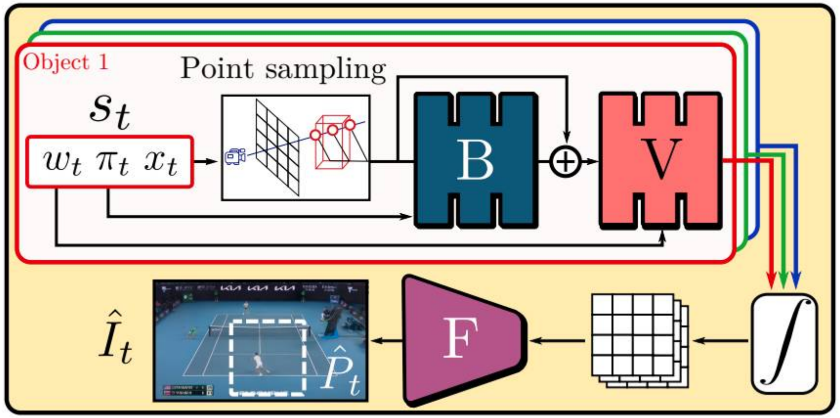
The Synthesis Module

Pros

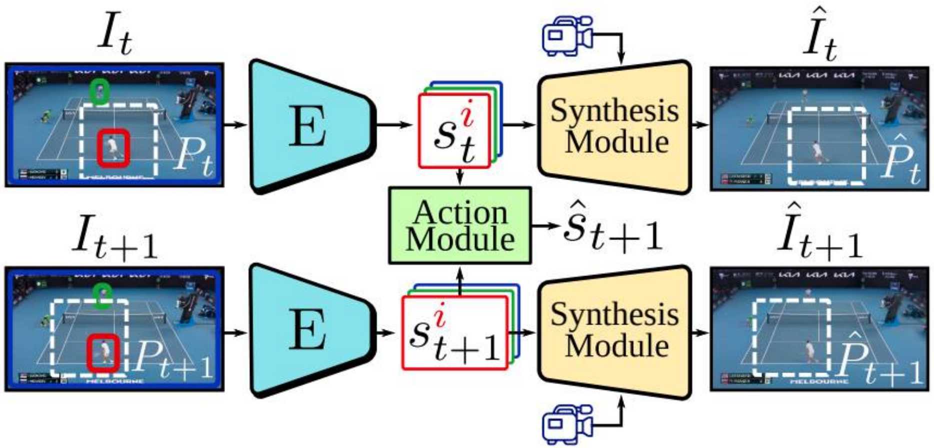
- Can generate novel views
- Does not require action label in the data
- Represents complex 3D scenes (NeRF renderer)

Playable Environments

max planck institut informatik



Playable Environments



The Synthesis Module

Pros

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

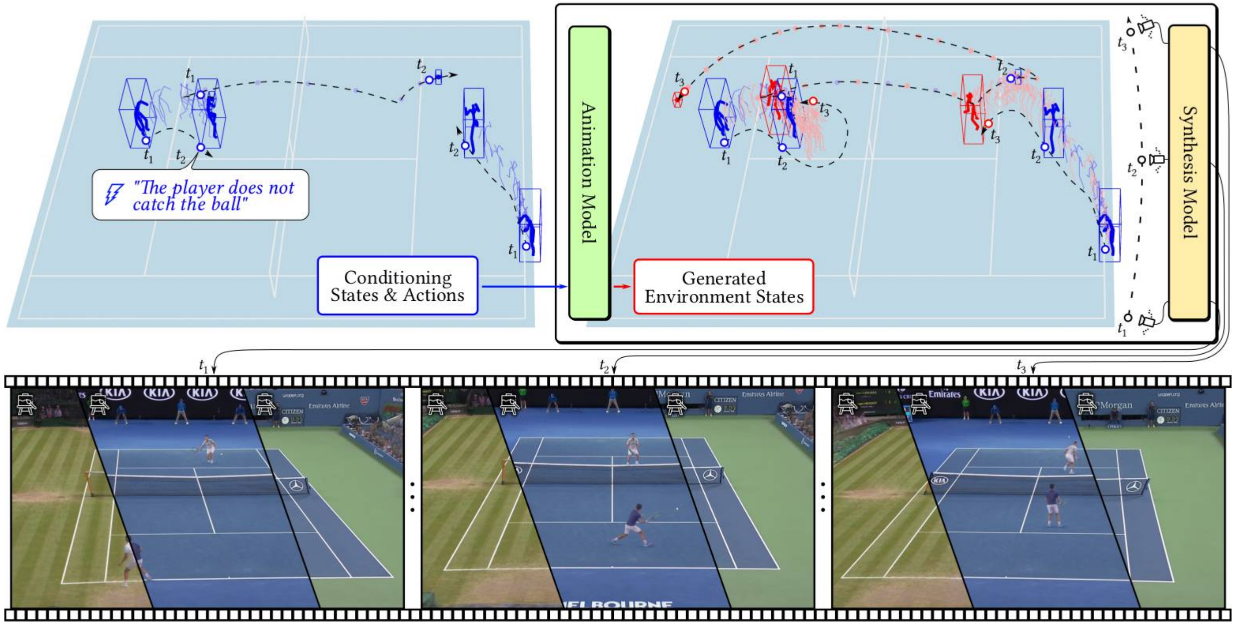
Cons

- 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

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!*

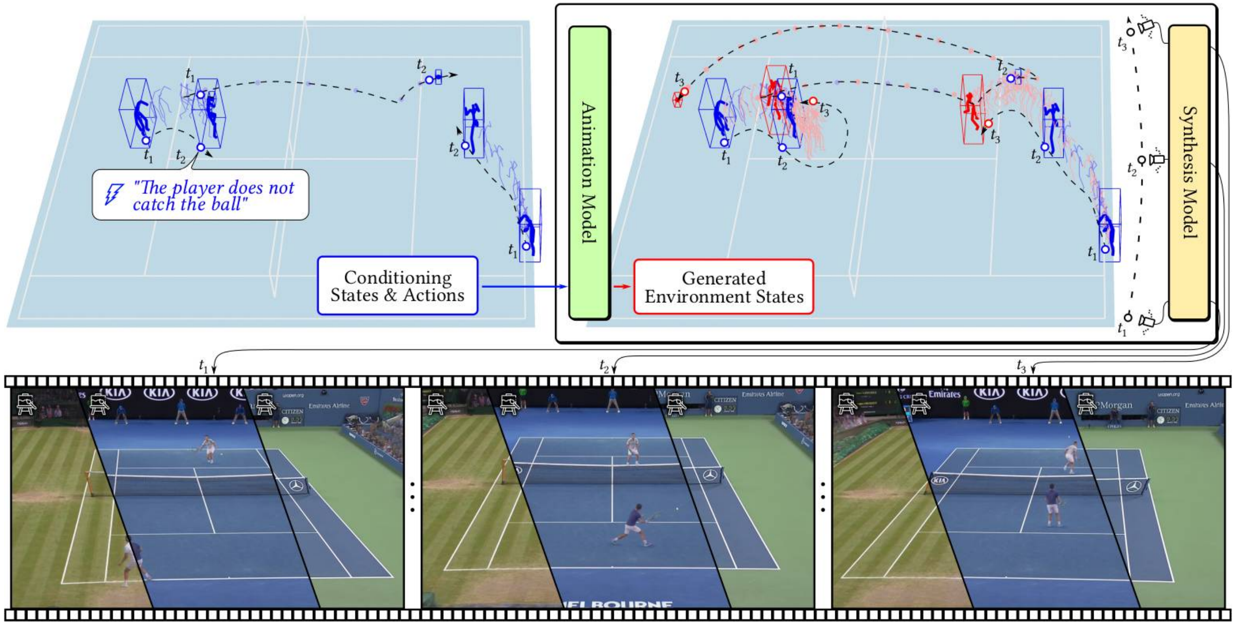


Enabling Fine-grained Control

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.*



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

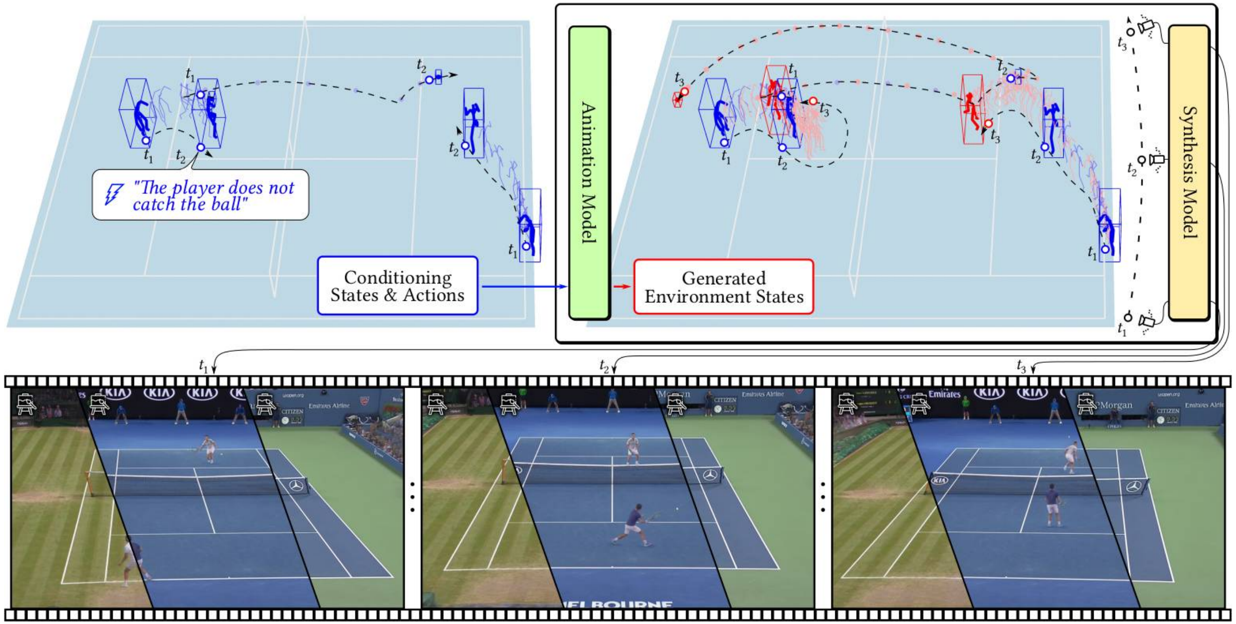
"the [other] player does not catch the ball"

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 →

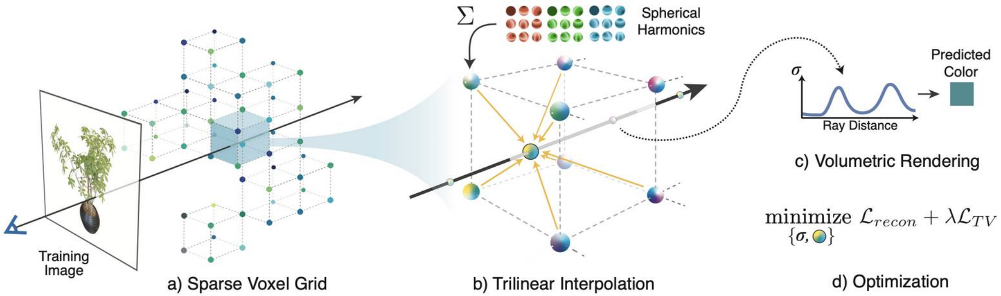
“hit the ball with a backhand and send it to the right service box”

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

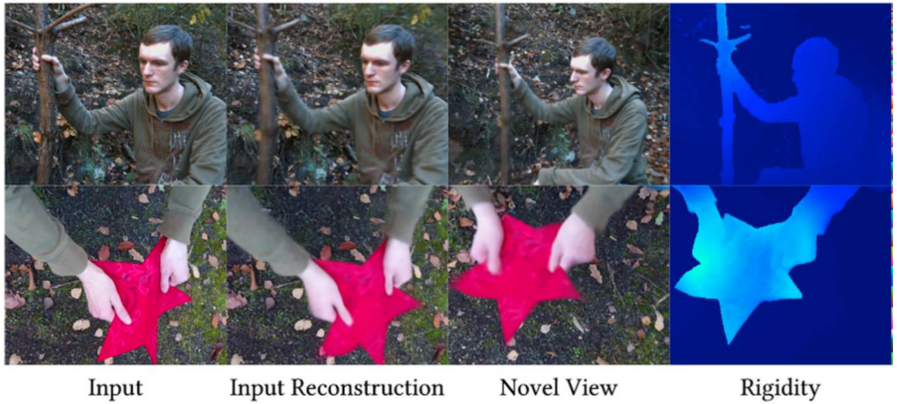
Improved Rendering

max planck institut
informatik

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.



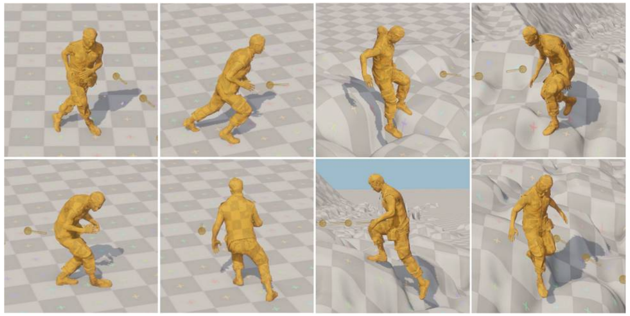
[Fridovich-Keil et al., 2022]



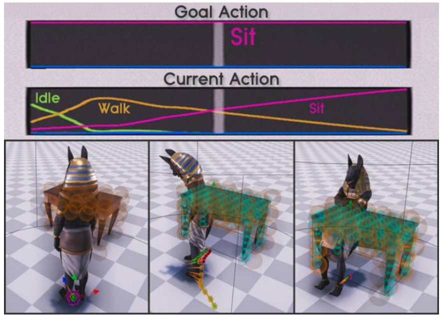
[Tretschk et al., 2021]

[Weng et al., 2022]

Related Works



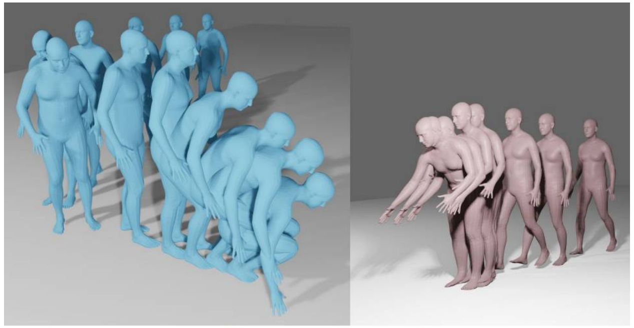
[Holden et al., 2020]



[Starke et al., 2019]

[Zhang et al., 2024]

(b) *Walking happily*



A person walks forward, then bends down.

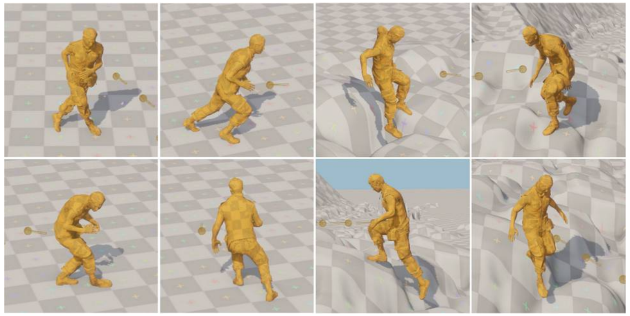
[Dabral et al., 2023]

[Zhang et al., 2024]

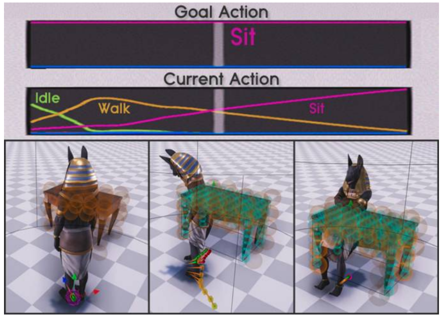
Learned Character Animation / Text-driven Generation

max planck institut informatik

Related Works

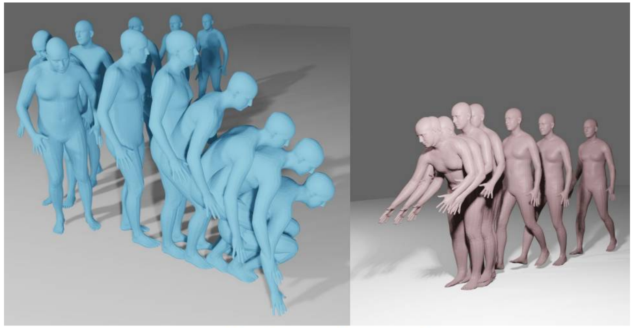


[Holden et al., 2020]



[Starke et al., 2019]

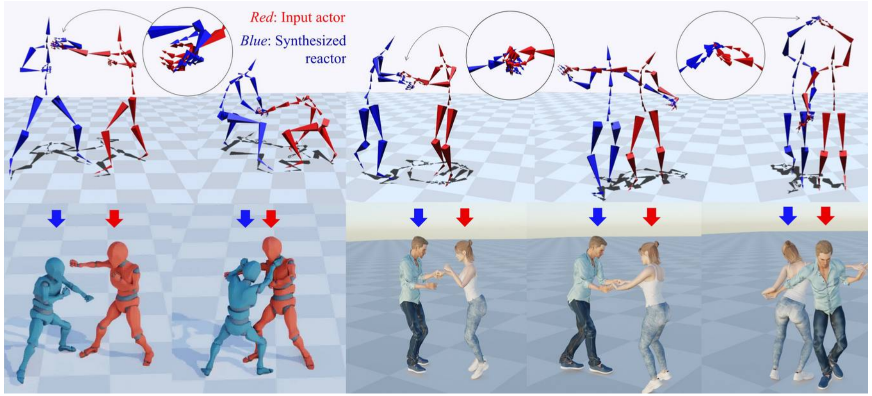
[Zhang et al., 2024] (b) *Walking happily*



A person walks forward, then bends down.

[Dabral et al., 2023]

[Zhang et al., 2024]



[Ghosh et al., 2023]

Learned Character Animation / Text-driven Generation

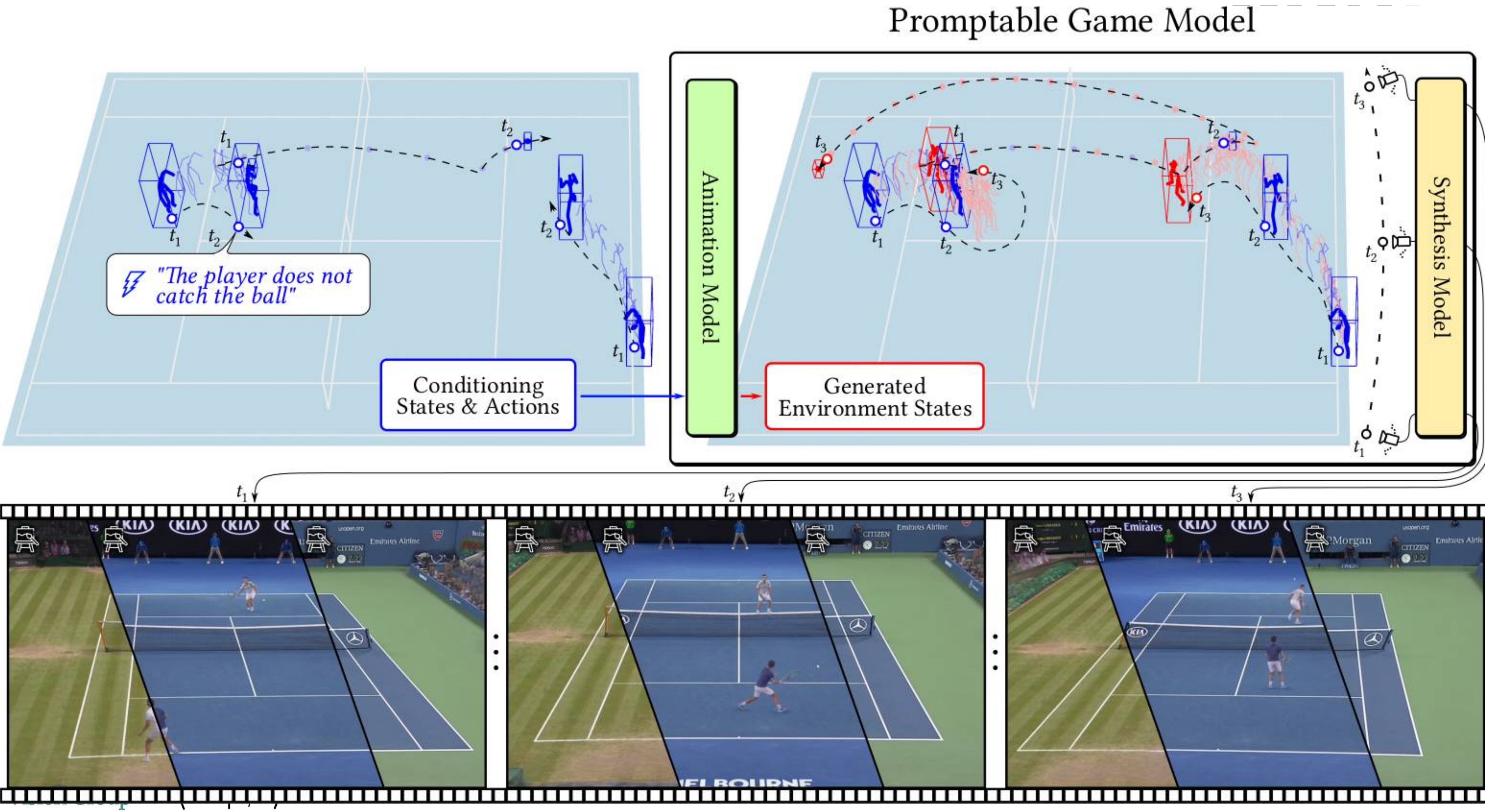
max planck institut
informatik

Promptable Game Models (PGMs): Text-guided Game Simulation via Masked Diffusion Models

4D and Quantum Vision Group

Vladislav Golyanik

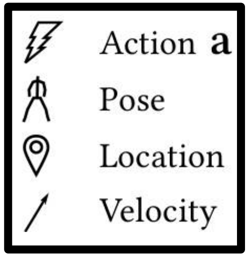

KIA
KIA
MELBOURNE



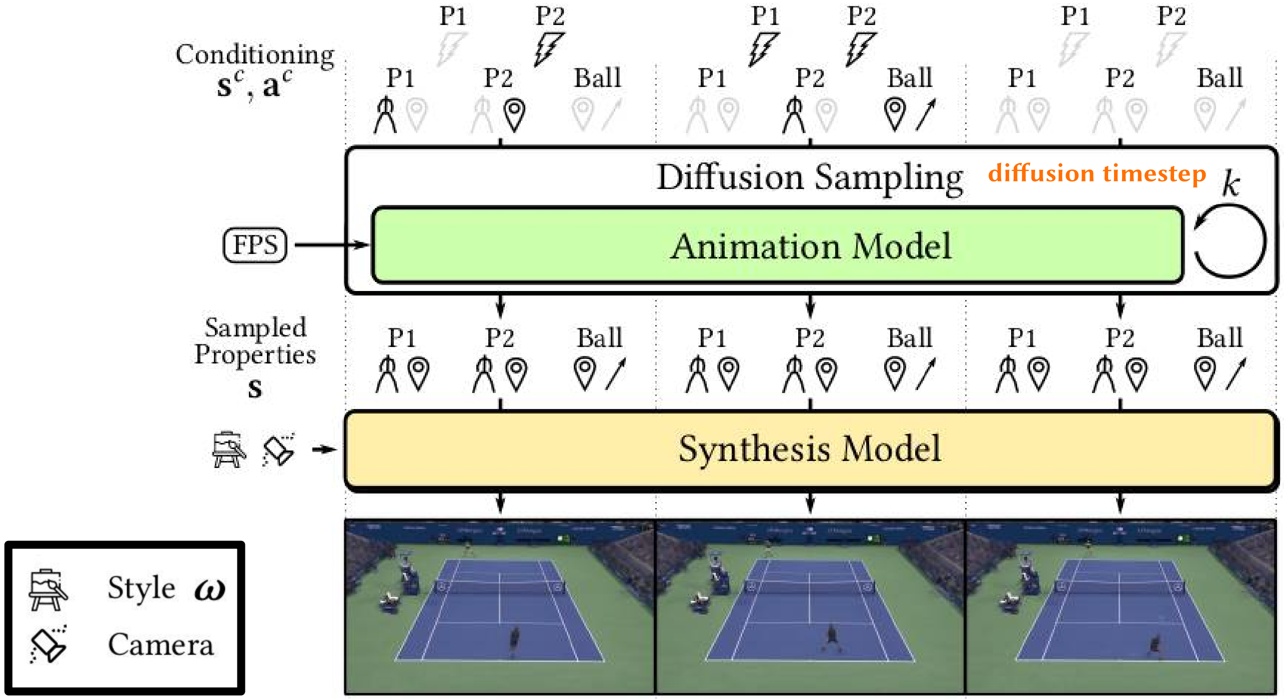
Promptable Game Model
"The player does not catch the ball"
Conditioning States & Actions
Animation Model
Generated Environment States
Synthesis Model

Overview: PGM as a State Machine

max planck institut
informatik



= ***text***
= ***kinematic tree***

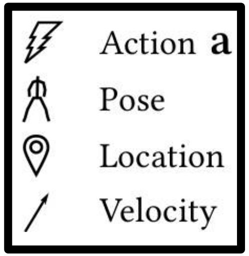


frame i   frame i+1   frame i+2   …   T

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: PGM as a State Machine

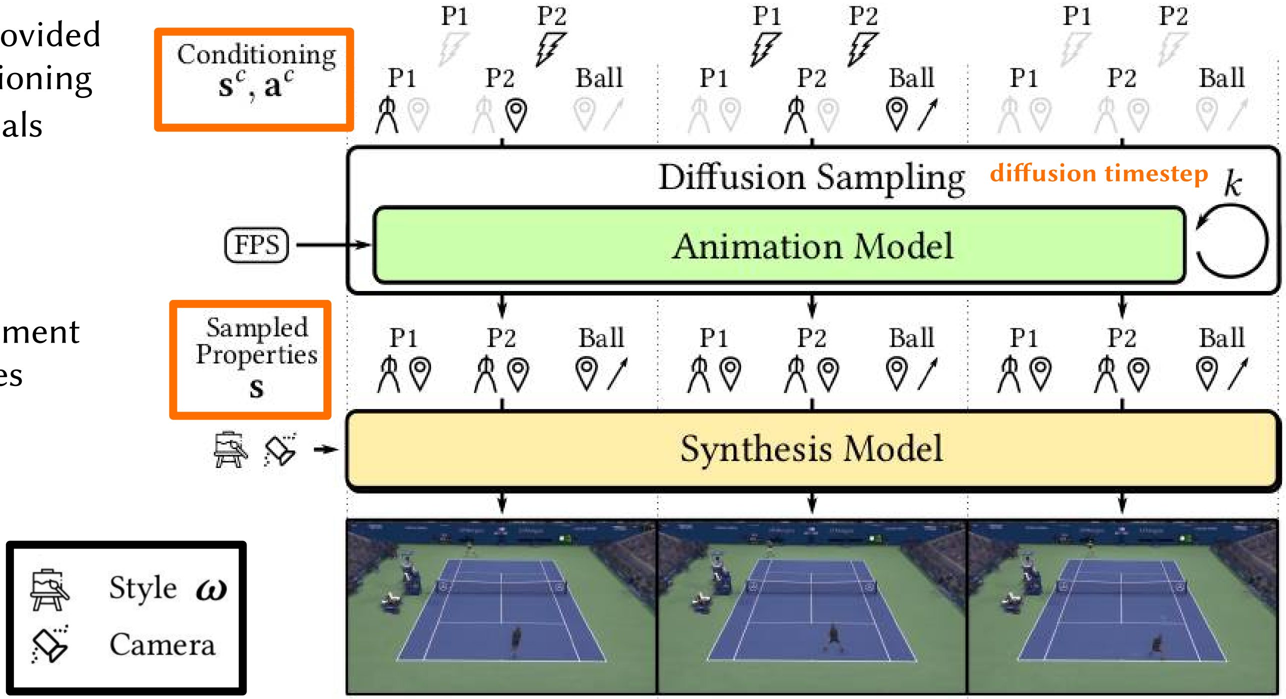


= *text*

= *kinematic tree*

user-provided conditioning signals

environment states

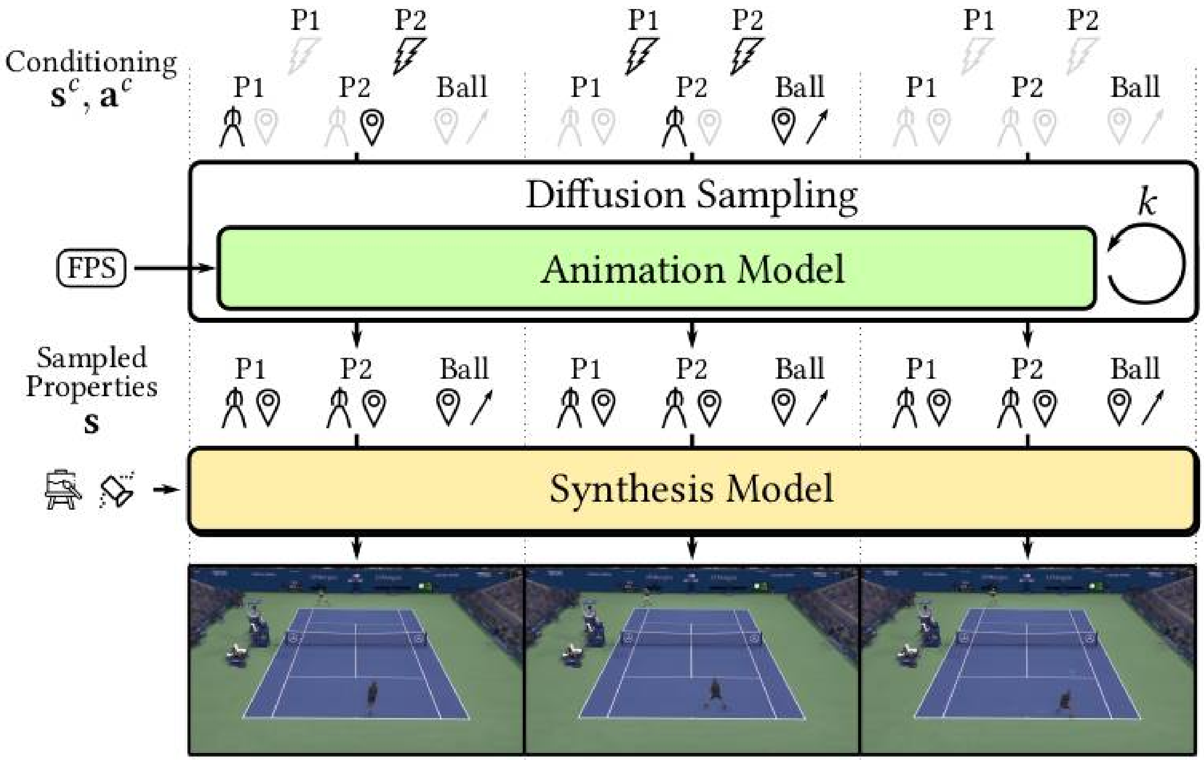


frame i frame i+1 frame i+2 … T

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

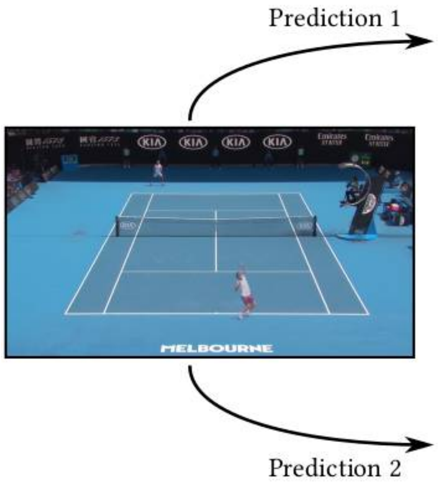
Example: *"The player takes several steps to the right and hits the ball with a backhand"*

↓

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

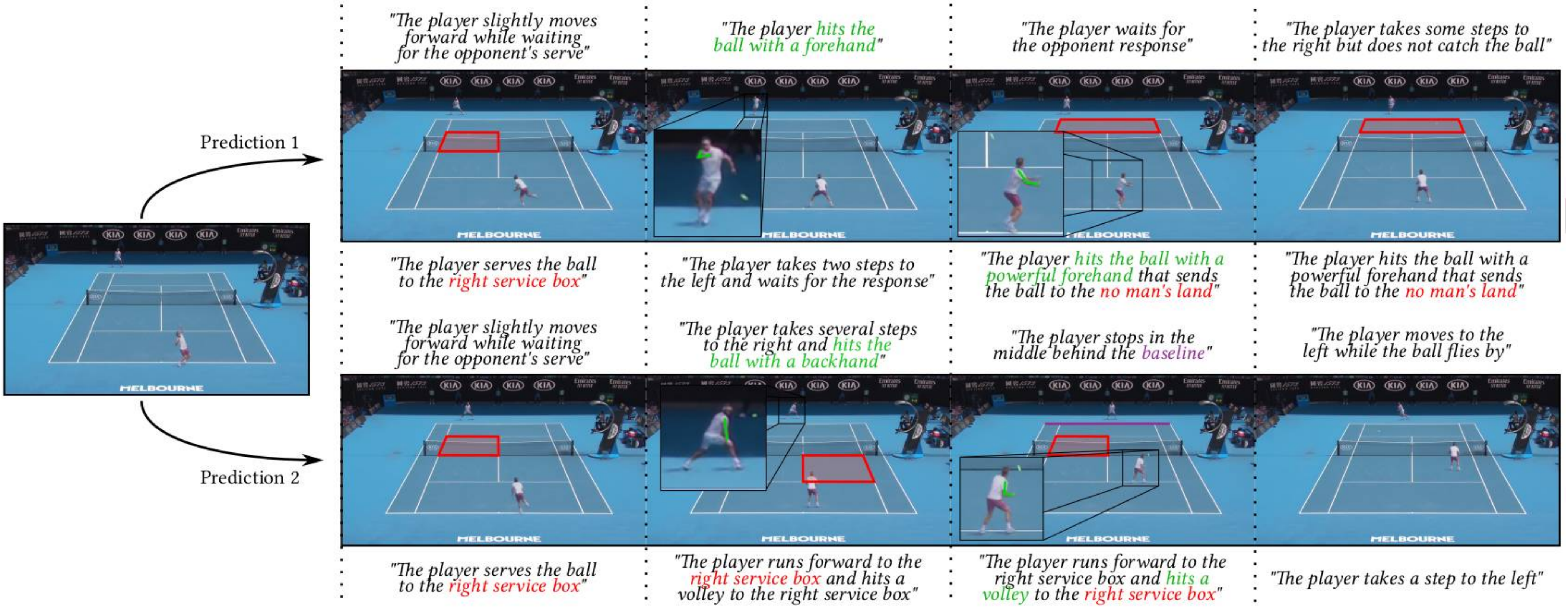
Training: A dataset of camera-calibrated videos with per-frame annotations (s and a).

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)

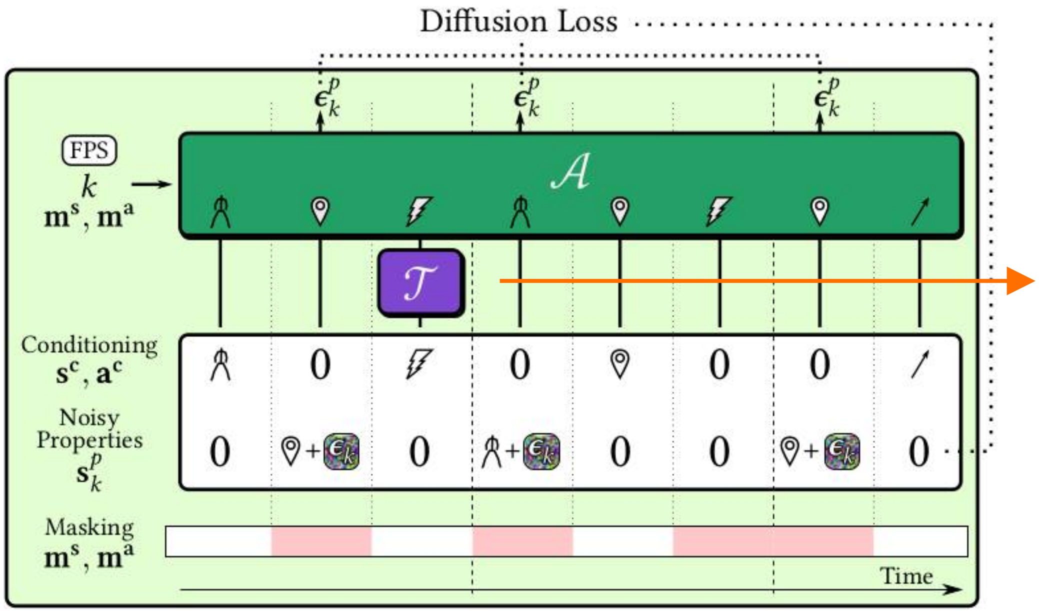
conditioning values:

$\mathbf{s}^c \in \mathbb{S}^T$

text $\mathbf{a}^c \in \mathbb{L}^{A \times T}$

$\mathbf{m}^\mathrm{s} \in \{0, 1\}^{P \times T}$

$\mathbf{m}^\mathrm{a} \in \{0, 1\}^{A \times T}$



Temporal model based on non-autoregressive transformer

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

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

[Raffel et al., 2022]

Animation Module (AM)

max planck institut informatik

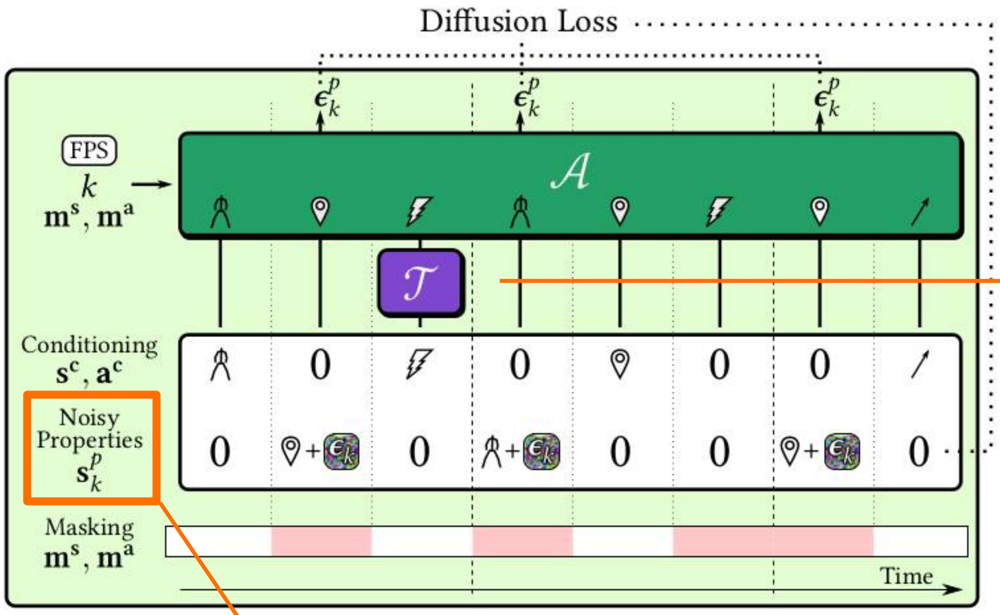
conditioning values:

$\mathbf{s}^c \in \mathbb{S}^T$

text $\mathbf{a}^c \in \mathbb{L}^{A \times T}$

$\mathbf{m}^\mathbf{s} \in \{0, 1\}^{P \times T}$

$\mathbf{m}^\mathbf{a} \in \{0, 1\}^{A \times T}$



Temporal model based on non-autoregressive transformer

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

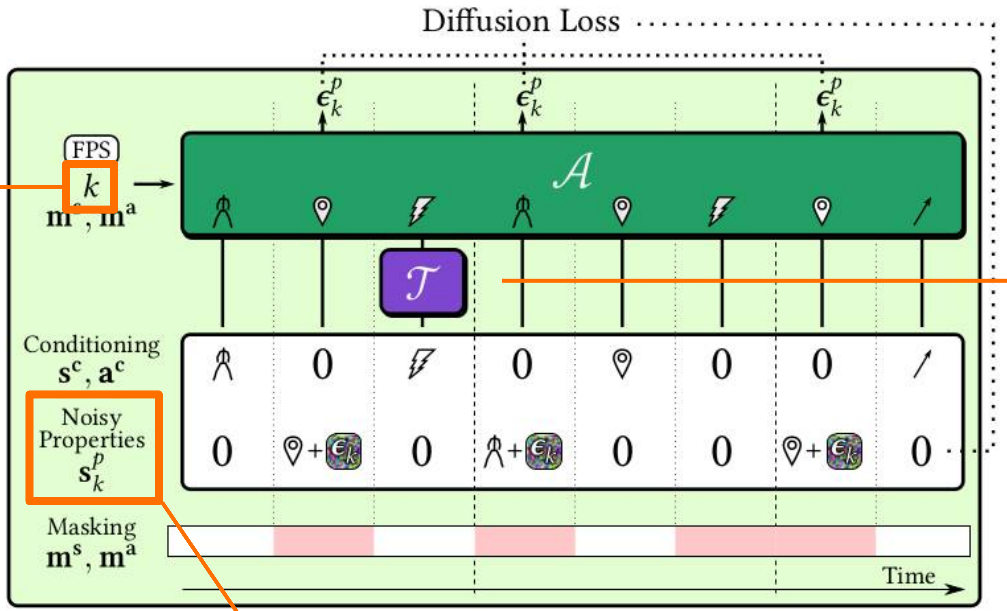
$\mathbf{a}^{\text{emb}} = \mathcal{T}(\mathbf{a}^c) \in \mathbb{R}^{A \times T \times N_t}$

[Raffel et al., 2022]

AM predicts $\mathbf{s}^p = \mathbf{s}_0^p$ as a progressive denoising process $\mathbf{s}_0^p, \ldots, \mathbf{s}_K^p$.

Animation Module (AM)

conditioning on k through weight demodulation



Temporal model based on non-autoregressive transformer

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

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

[Raffel et al., 2022]

conditioning values:

$\mathbf{s}^c \in \mathbb{S}^T$

text $\mathbf{a}^c \in \mathbb{L}^{A \times T}$

$\mathbf{m}^s \in \{0,1\}^{P \times T}$

$\mathbf{m}^a \in \{0,1\}^{A \times T}$

sampled according to various strategies emulating desired inference tasks

AM predicts $\mathbf{s}^p = \mathbf{s}_0^p$ as a progressive denoising process $\mathbf{s}_0^p, \ldots, \mathbf{s}_K^p$.

Animation Module (AM)

max planck institut informatik

conditioning on k through weight demodulation

conditioning values:

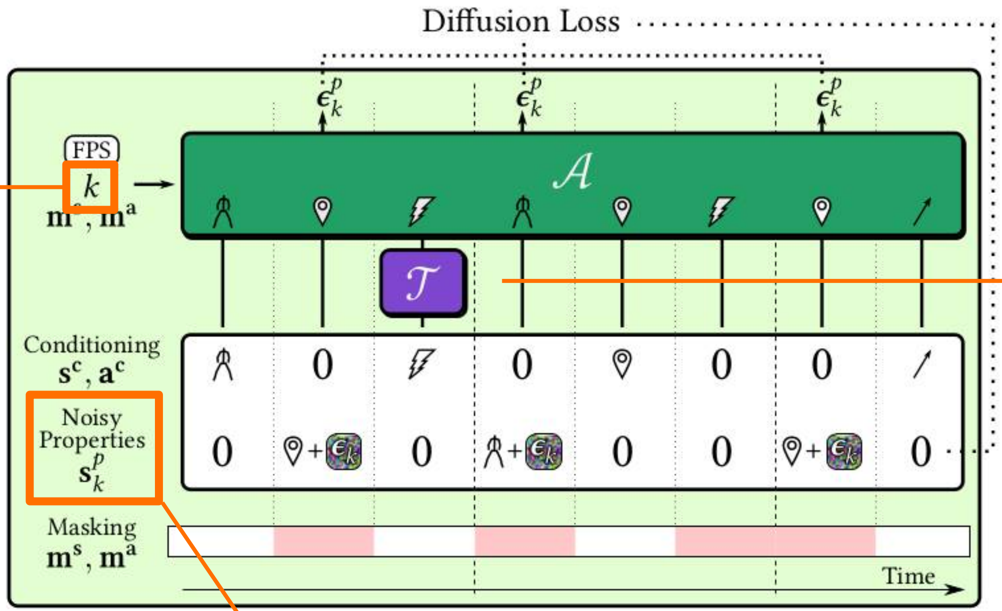
$\mathbf{s}^c \in \mathbb{S}^T$

text $\mathbf{a}^c \in \mathbb{L}^{A\times T}$

$\mathbf{m}^s \in \{0,1\}^{P\times T}$

$\mathbf{m}^a \in \{0,1\}^{A\times T}$

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

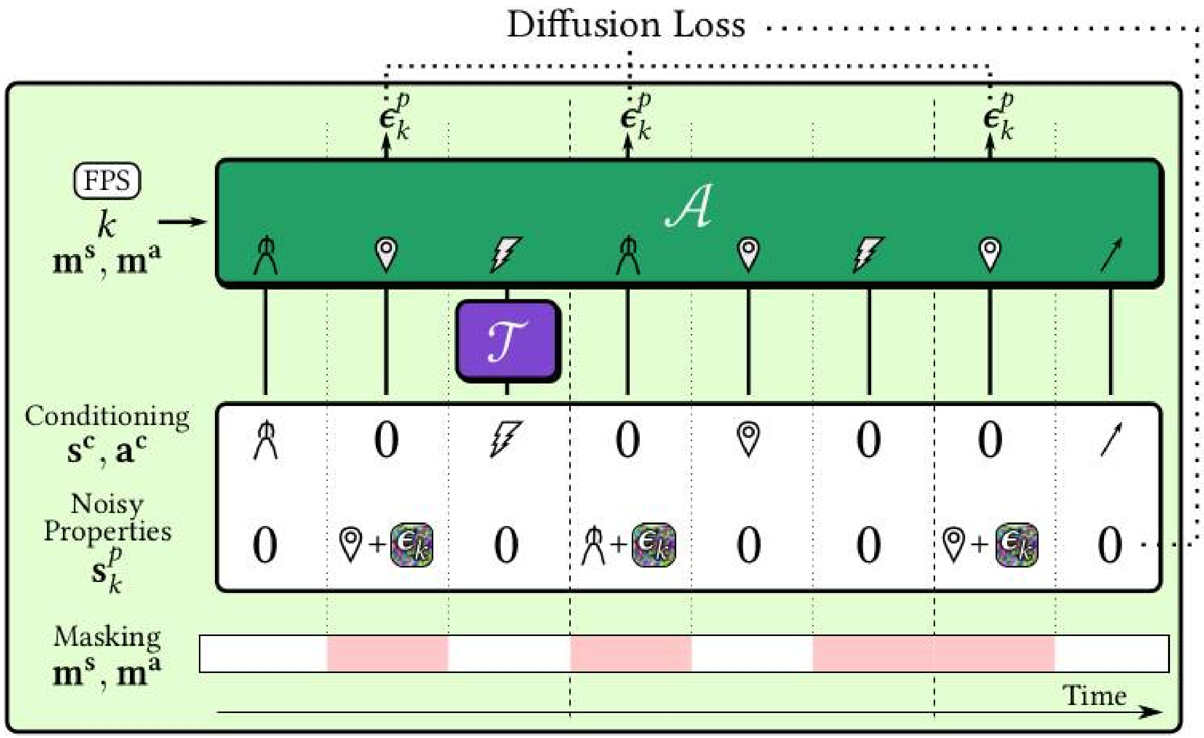
$\mathbf{a}^{\text{emb}} = \mathcal{T}(\mathbf{a}^c) \in \mathbb{R}^{A\times T\times N_t}$

[Raffel et al., 2022]

AM predicts $\mathbf{s}^p = \mathbf{s}_0^p$ as a progressive denoising process $\mathbf{s}_0^p, \dots, \mathbf{s}_K^p$.

$\mathcal{A}$ acts as a noise estimator predicting Gaussian noise ϵ_k in the noisy sequence of unknown states $\mathbf{s}_k^p$: $\epsilon_k^p = \mathcal{A}(\mathbf{s}_k^p|\mathbf{s}^c, \mathbf{a}^{\text{emb}}, \mathbf{m}^s, \mathbf{m}^a, k)$.

Animation Module (AM)



$$\epsilon_k^p = \mathcal{A}(\mathbf{s}_k^p | \mathbf{s}^c, \mathbf{a}^{\text{emb}}, \mathbf{m}^s, \mathbf{m}^a, k)$$

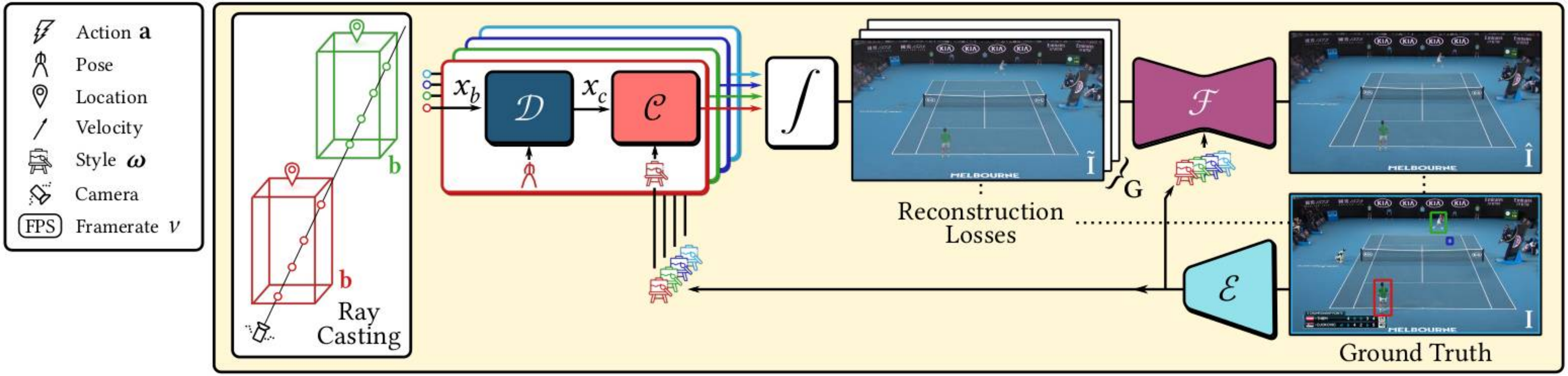
Minimising the DDPM training objective [Ho *et al.*, 2020]:

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

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)



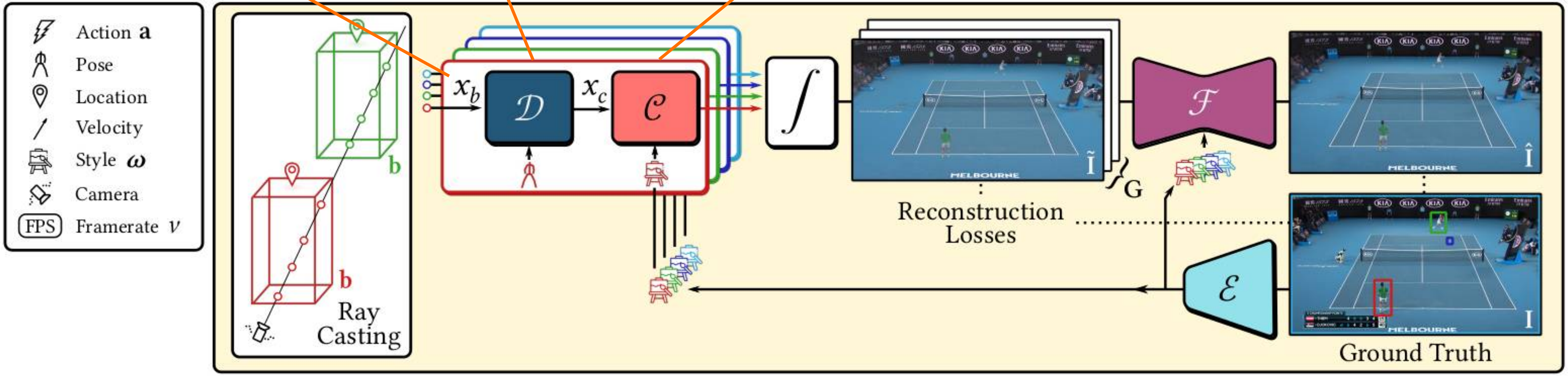
- Coarsely follows Playable Environments [Menapace et al., 2022]

Synthesis Module (SM)

Point in the deformed ray space

Deformation Model [Weng et al., 2022]

Canonical Pose



composition of independent objects
(parametrised with voxel grids)

+ fully-opaque planes

[Fridovich-Keil et al., 2022]

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

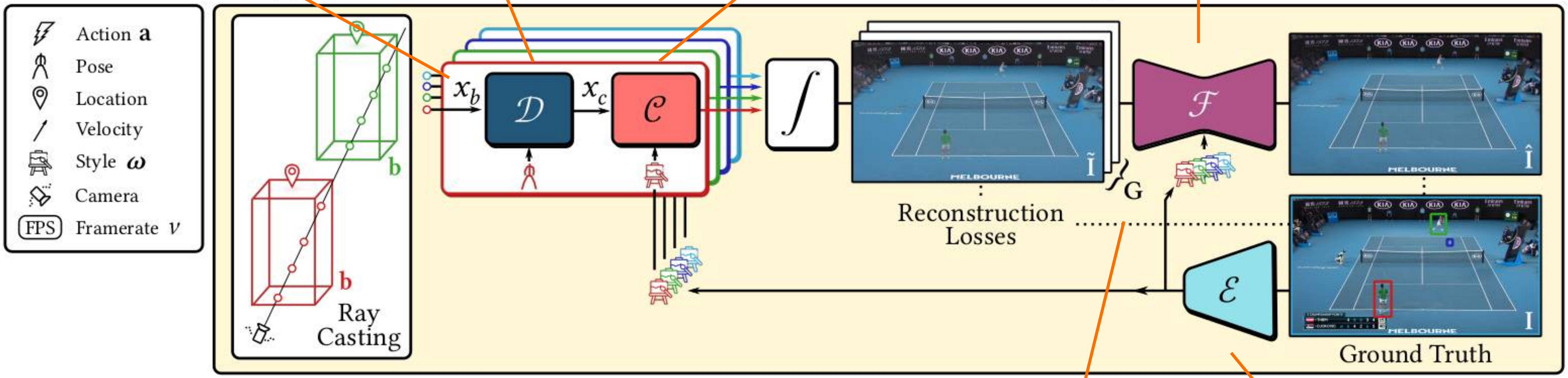
Synthesis Module (SM)

Point in the deformed ray space

Deformation Model [Weng et al., 2022]

Canonical Pose

Feature Enhancer (CNN): G in, an RGB image out



composition of independent objects
(parametrised with voxel grids)
[Fridovich-Keil et al., 2022]

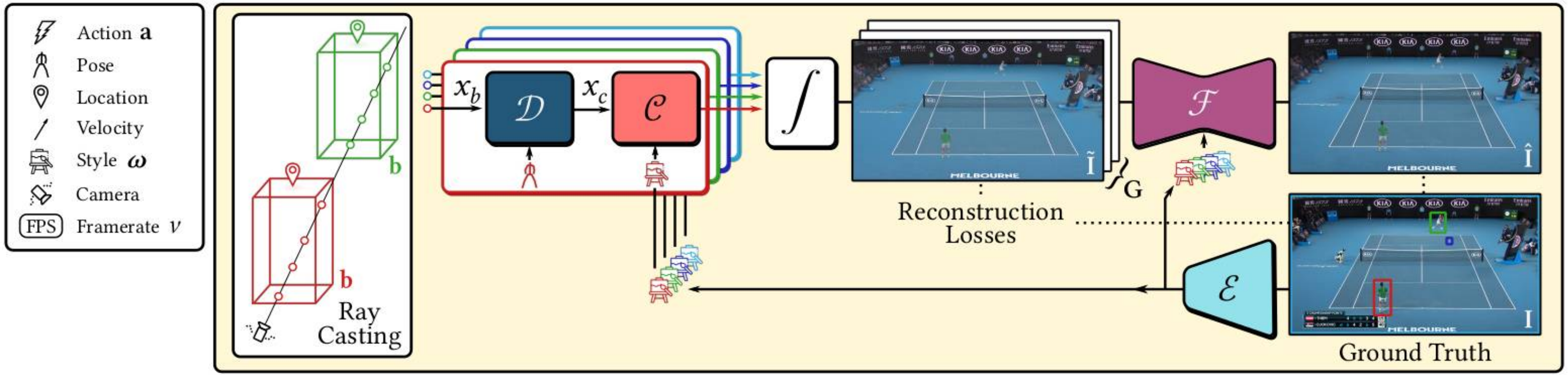
+ fully-opaque planes

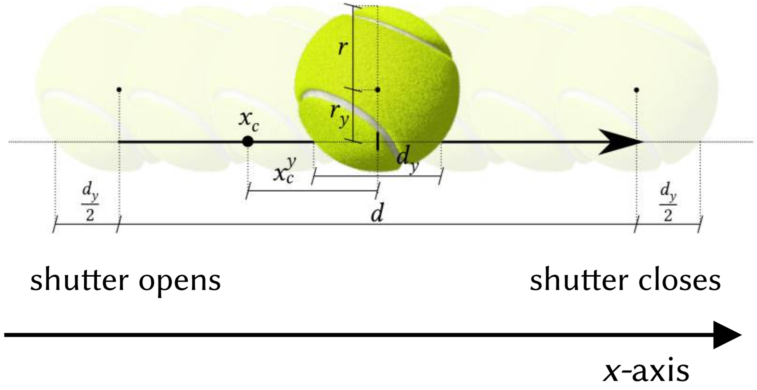
Feature Grid

Style Encoder

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

SM: Object-specific Rendering

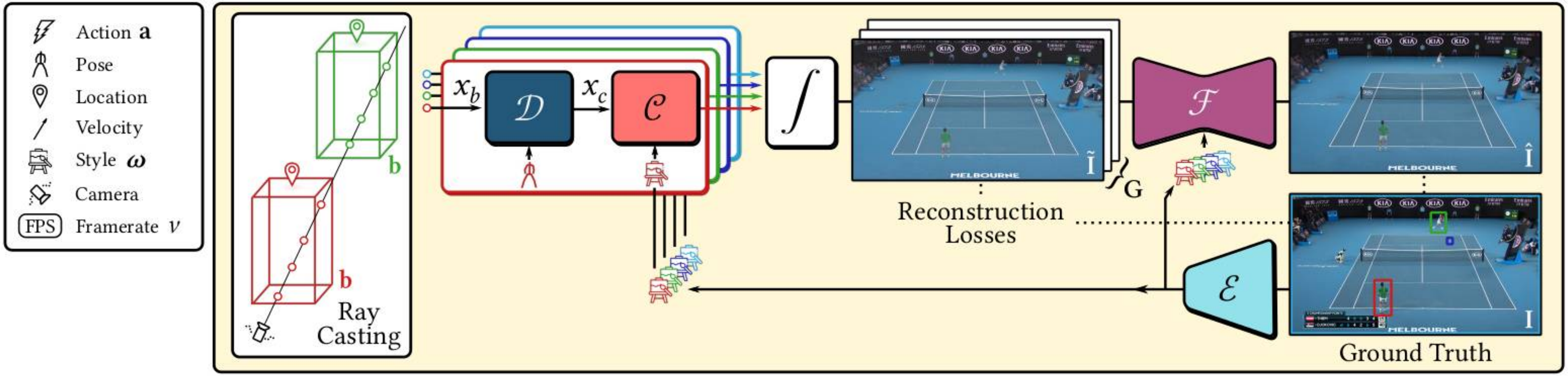




Ball rendering

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

max planck institut informatik

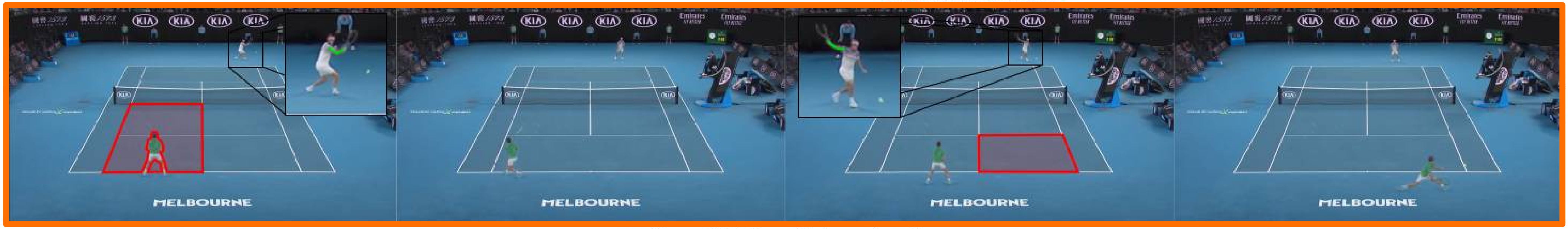
response by running to the right (top)

"The player runs to the right and performs another backhand to the left side of the field"

"The player moves slightly to the left"

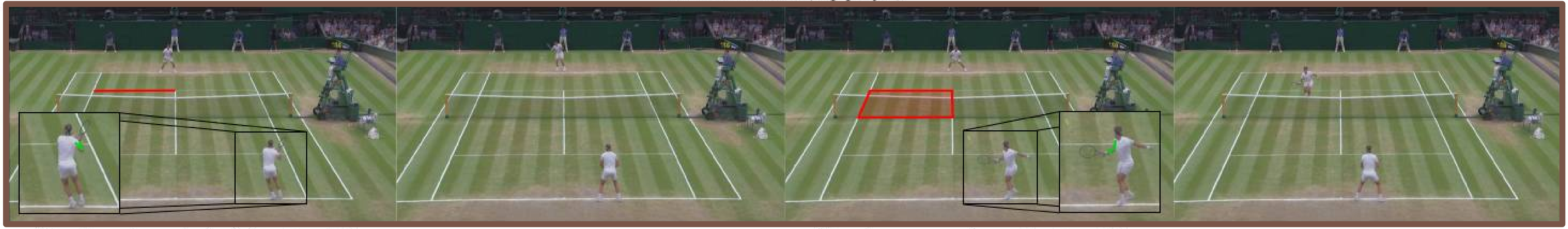
"The player takes steps to the right and sends the ball to the right side of no man's land with a backhand"

"The player starts moving to the left"



Game AI Actions (Bottom player)

Game AI Actions (Top player)



"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"

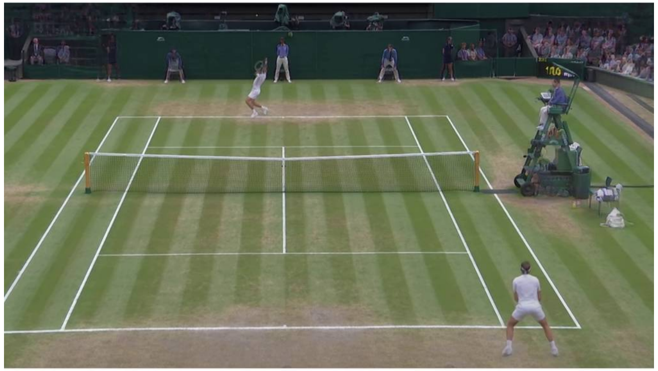
"The player moves to the right for the hit but the game is ended"

response by running towards the net (bottom)

Action-conditioned Sequence Generation

max planck institut informatik

The player serves to the right service box



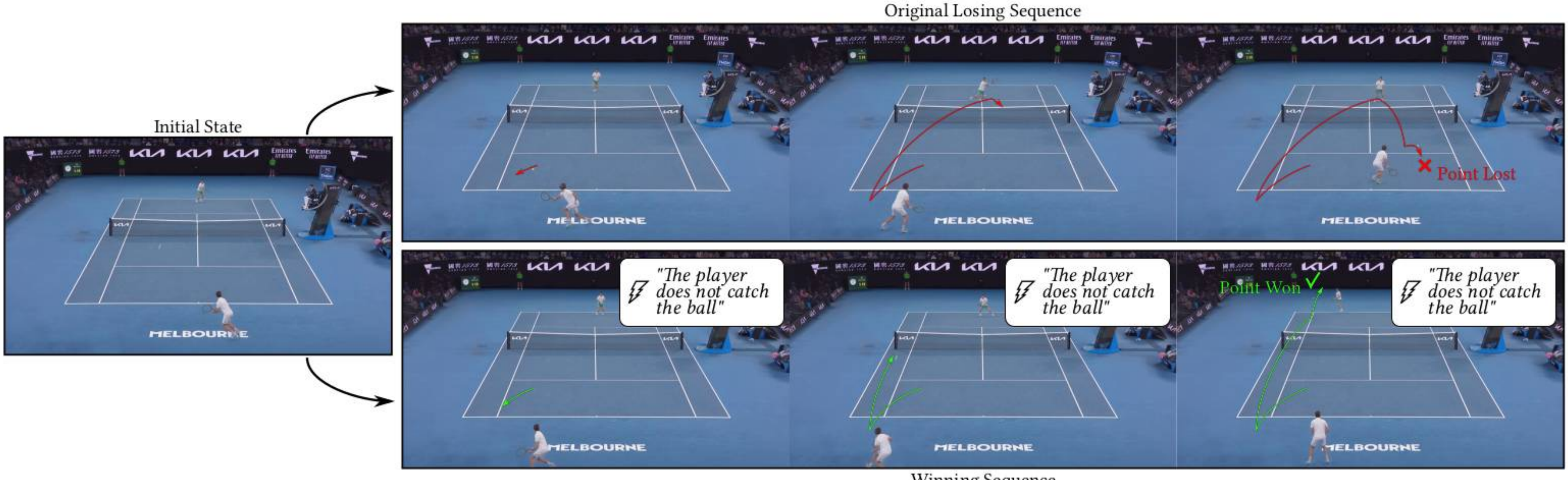
The player jumps forward and waits for the ball

Initial State

The player serves to the right service box

The player jumps forward and waits for the ball

Prompt-based Sequence Modification



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.

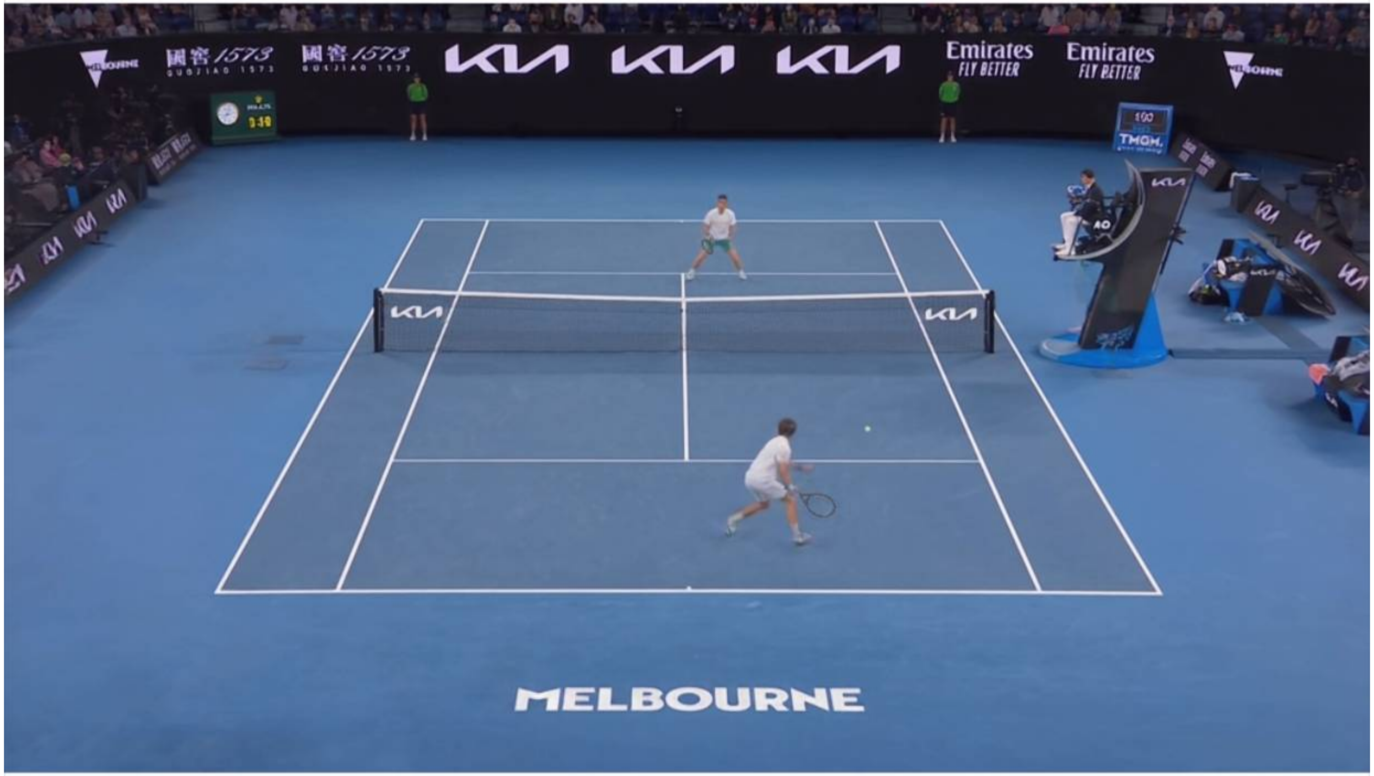
“The [top] player does not catch the ball”

max planck institut
informatik

**original video =
bottom player loses**

Example 1

the player makes two steps backwards while waiting for the response



The player rushes diagonally to the upper right and hits the ball with a forehand to the net

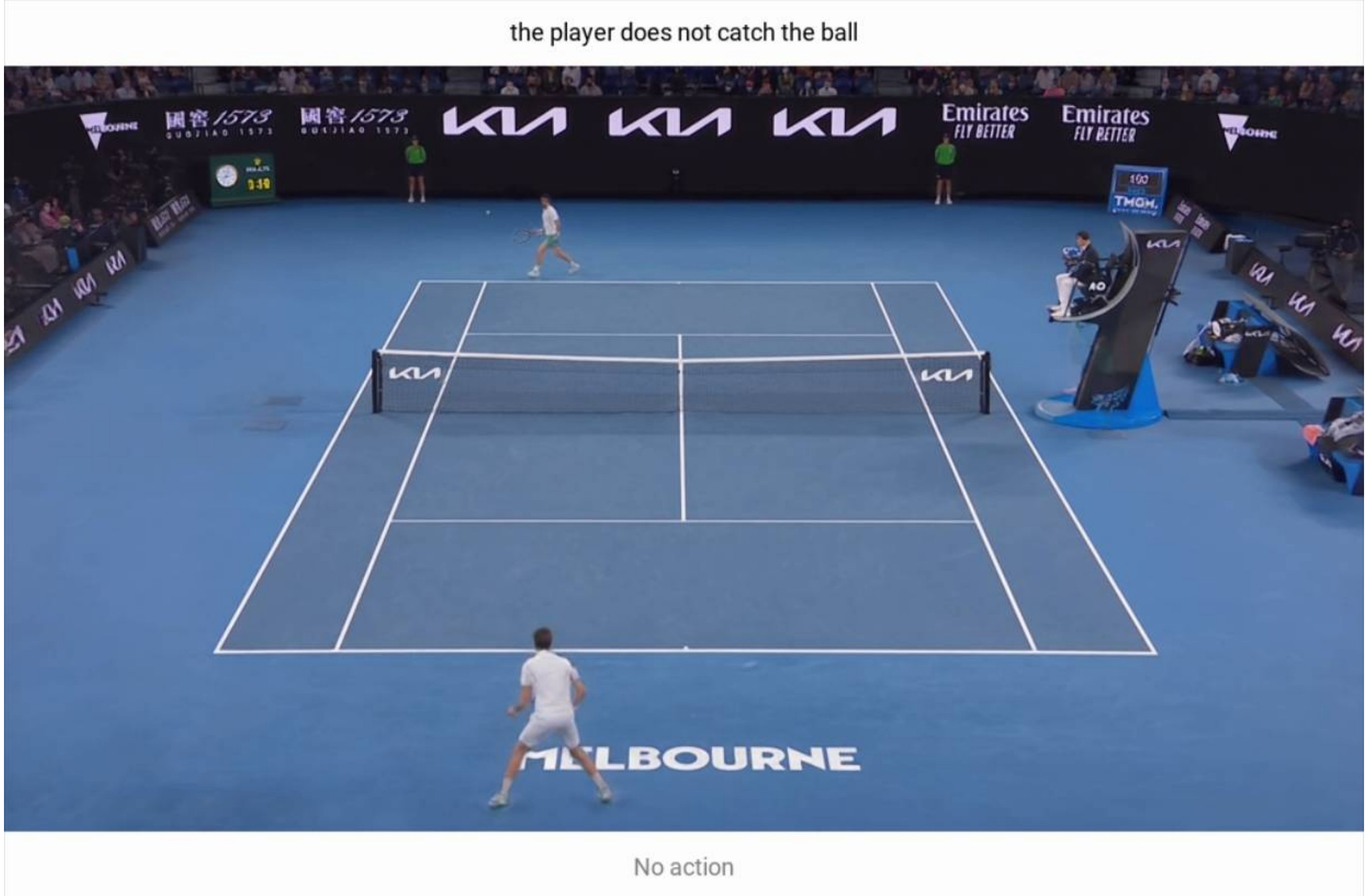
Vladislav Golyanik

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

max planck institut
informatik

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

Example 1



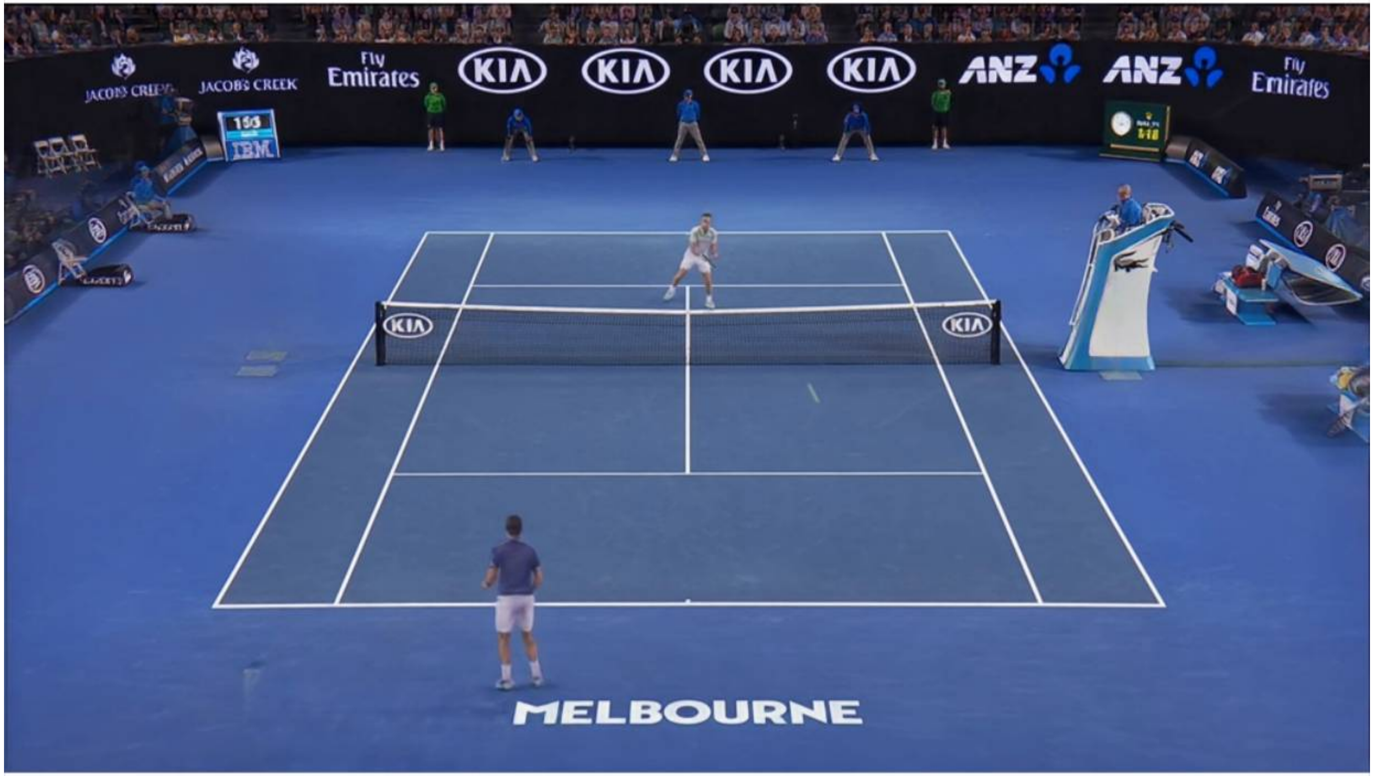
“The [top] player does not catch the ball”

max planck institut
informatik

original video =
bottom player loses

Example 2

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



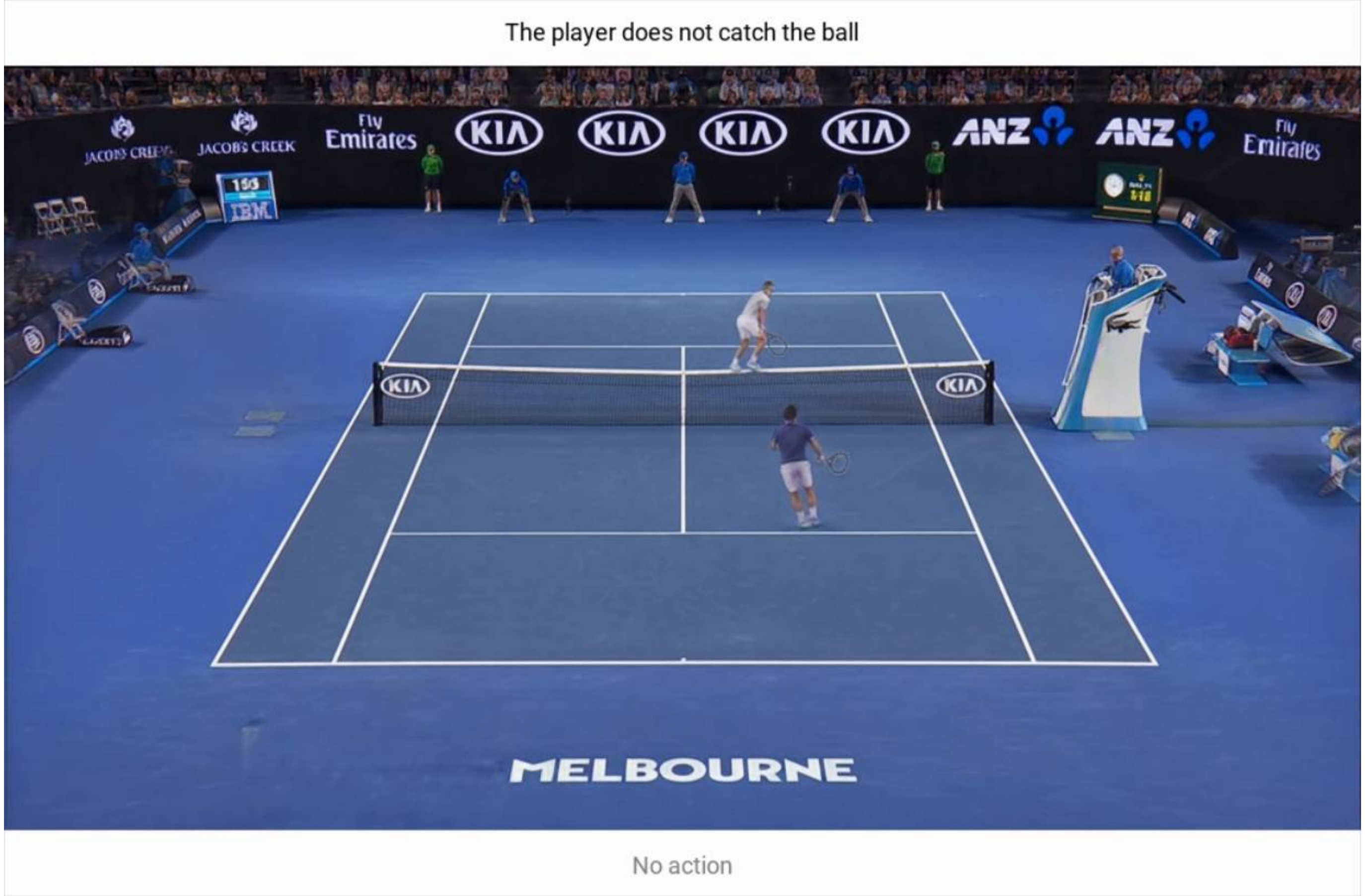
The player steps to the right and stops unable to save the ball

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

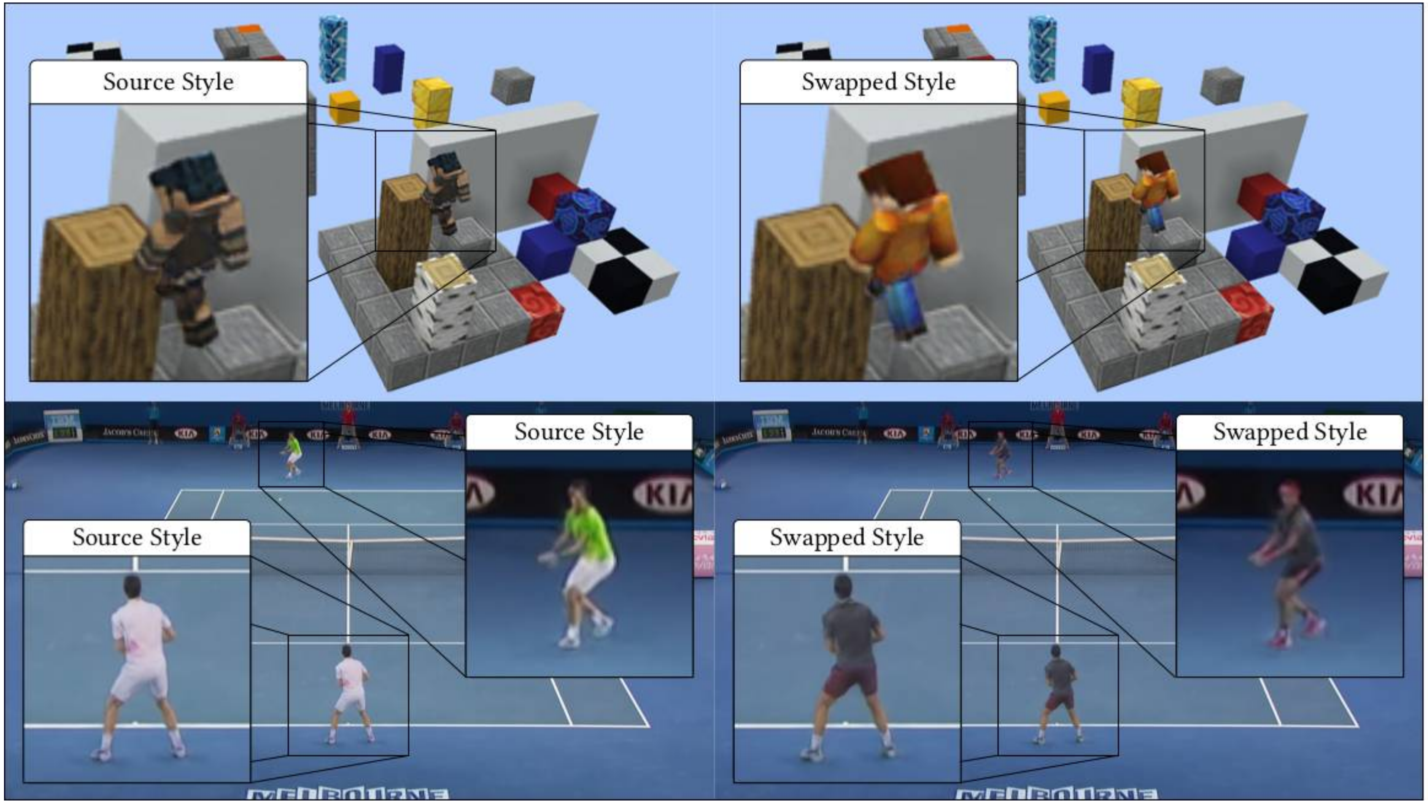
max planck institut
informatik

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

Example 2



Style Swap


Source Style
Swapped Style
Source Style
Source Style
Swapped Style
Swapped Style

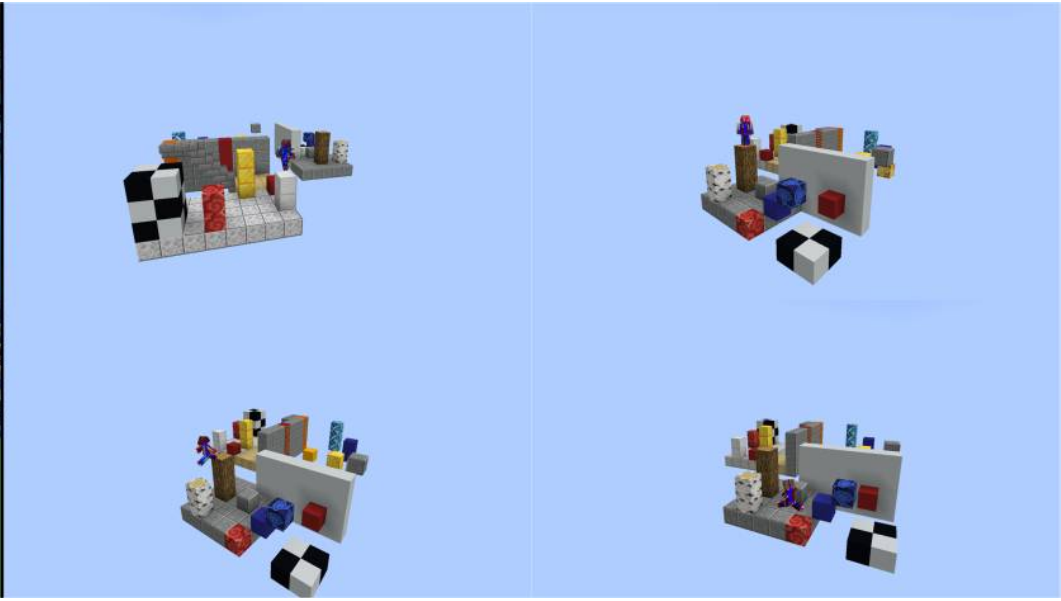

Target Style


Target Style

Tennis and Minecraft Datasets

max planck institut
informatik

Tennis and Minecraft Dataset



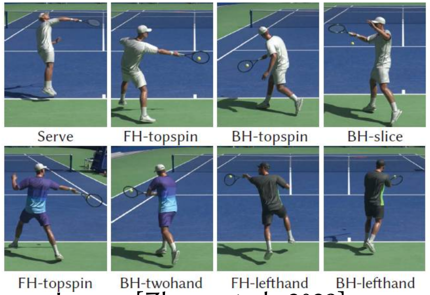
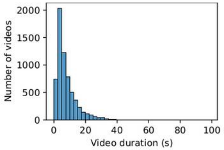
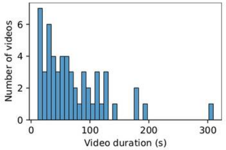


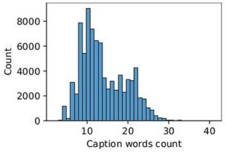
Image: [Zhang et al., 2023]



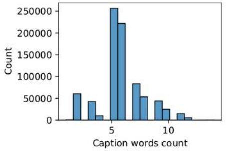
(a) Distribution of video durations in the Tennis dataset.



(b) Distribution of video durations in the Minecraft dataset.



(c) Distribution of words per caption in the Tennis dataset.



(d) Distribution of words per caption in the Minecraft dataset.

Tennis dataset (broadcast tennis matches):

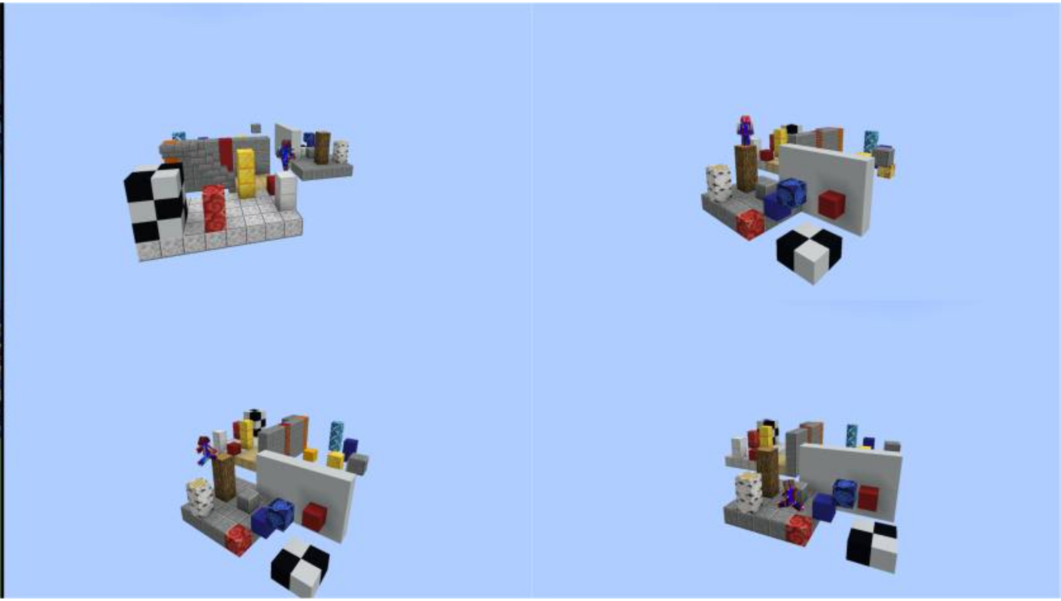
- 7.1k video sequences (1920x1080px at 25 fps)
- 15.5h
- 1.12M fully-annotated frames
- 25.5k unique captions and 915 unique words

4D and Quantum Vision Group

Vladislav Golyanik

Tennis and Minecraft Dataset

max planck institut
informatik

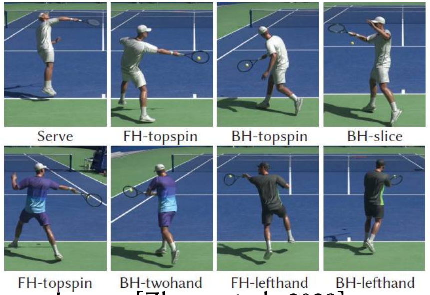
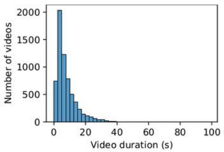
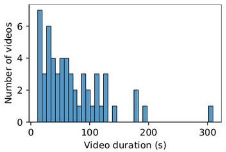


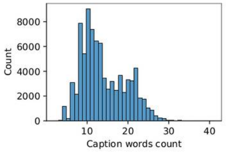
Image: [Zhang et al., 2023]



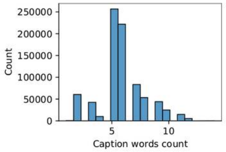
(a) Distribution of video durations in the Tennis dataset.



(b) Distribution of video durations in the Minecraft dataset.



(c) Distribution of words per caption in the Tennis dataset.



(d) Distribution of words per caption in the Minecraft dataset.

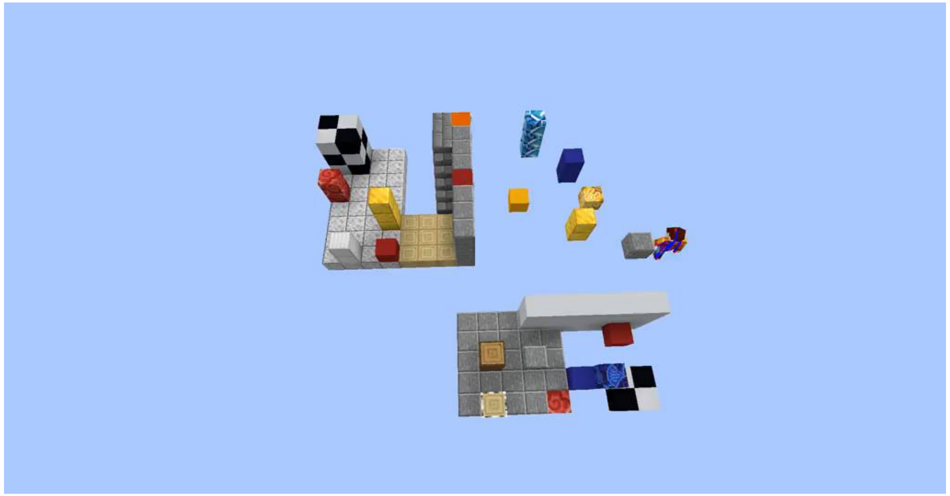
Minecraft dataset (from the video game):

- 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

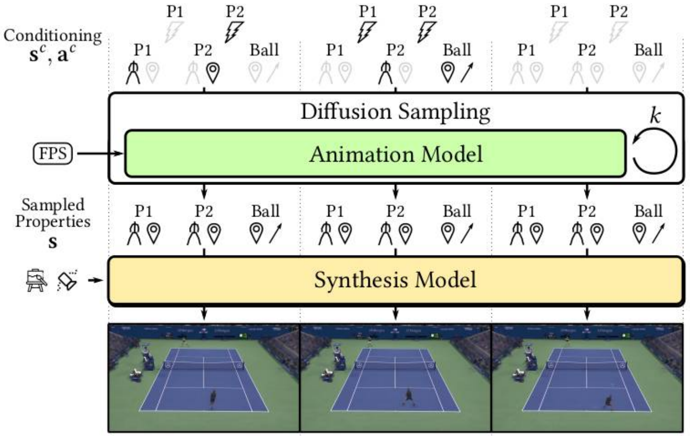


The player sprints and jumps on the first block of the second area

Annotation and Training Costs

	Tennis	Minecraft
Sequences:	7112	61
train	5690	51
validation	711	5
test	711	5
Duration:	15.5h	1.21h
train	12.4h	0.952h
validation	1.59h	0.16h
test	1.52h	0.101
Annotated frames:	1.12M	68.5k
train	1.05M	64.5k
validation	135k	11.2k
test	130k	7.06k
Resolution	1920x1080px	1024x576px
Framerate	25fps	20fps
Captions	84.1k	818k
of which unique	25.5k	1.24k
Unique words	915	117
Avg. words	13.8	5.85
Avg. span	1.32s	0.500s
Parts of sentence:		
Nouns	32.3%	36.2%
Verbs	11.9%	17.4%
Adjectives	3.08%	6.48%
Adverbs	2.70%	11.7%
Pronouns	0.18%	0.00%
Articles	26.4%	8.03%
Prepositions	7.89%	6.98%
Numerals	0.11%	0.03%
Particles	9.28%	1.50%
Punctuation	1.76%	1.12%
Others	0.00%	0.00%

dataset statistics



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

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$)
- Minecraft dataset: Two days (211$)

Experimental Results

Quantitative Results (AM and SM)

max planck institut informatik

Tennis	Position L2↓	Position FD↓	Root angle L2↓	Root angle FD↓	Joints 3D L2↓	Joints 3D FD↓
PE	3.291	229.112	1.126	15.953	0.303	53.242
Rec. LSTM	1.597	7.253	0.907	7.051	0.193	16.735
Rec. Transf.	1.074	4.402	0.767	6.838	0.175	14.845
Ours Small	1.380	1.443	1.014	0.560	0.148	1.253
Ours	1.099	0.929	0.844	0.356	0.129	0.836
Minecraft	Position L2↓	Position FD↓	Root angle L2↓	Root angle FD↓	Joints 3D L2↓	Joints 3D FD↓
PE	2.739	105.973	1.620	31.232	0.311	39.572
Rec. LSTM	2.292	47.296	1.702	49.971	0.489	99.843
Rec. Transf.	2.154	53.198	1.430	36.123	0.385	69.977
Ours Small	1.084	4.461	1.077	6.016	0.140	3.590
Ours	1.065	4.815	0.956	4.083	0.132	3.360

* results averaged over all tasks

Animation Module

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

Synthesis Module

Reconstruction tasks for AM evaluation:

- Video prediction conditioned on actions
- Unconditioned video prediction
- Opponent modeling
- Sequence completion

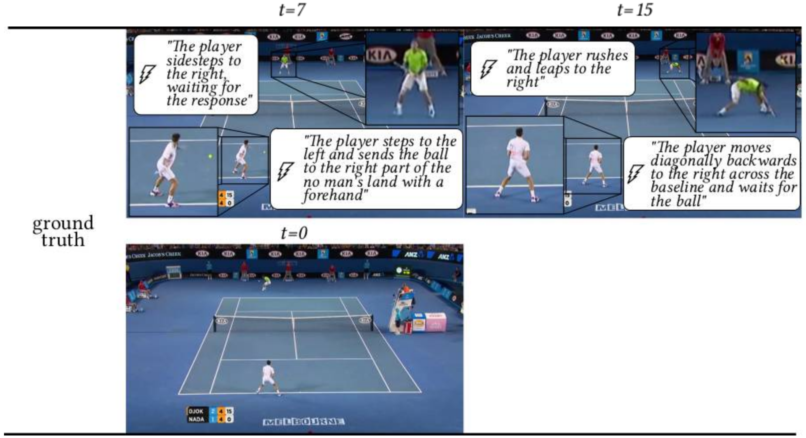
Comparison to PE and Ablation Study

max planck institut informatik

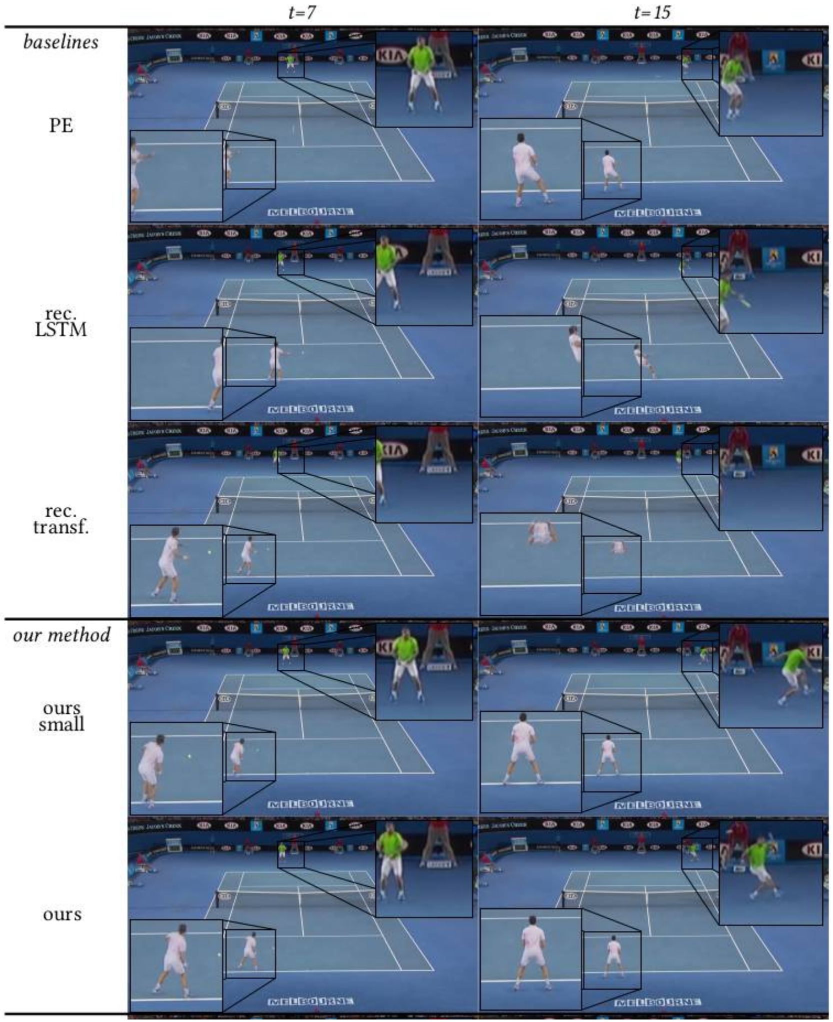
PGM generates sharper players and static scene elements.
PE and PE+ produce checkerboard artifacts

Quantitative Results (SM and AM)

max planck institut informatik



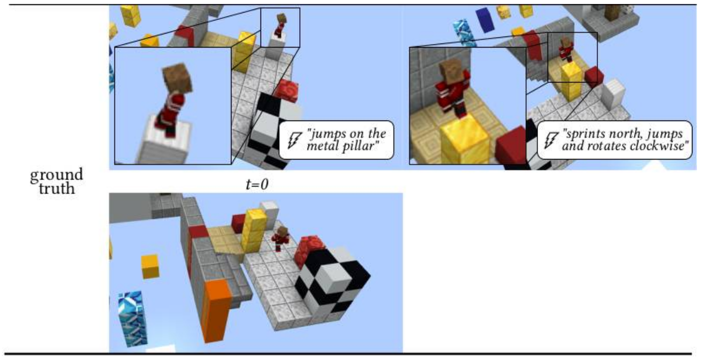
The model uses the first-frame object properties and all actions as conditioning.

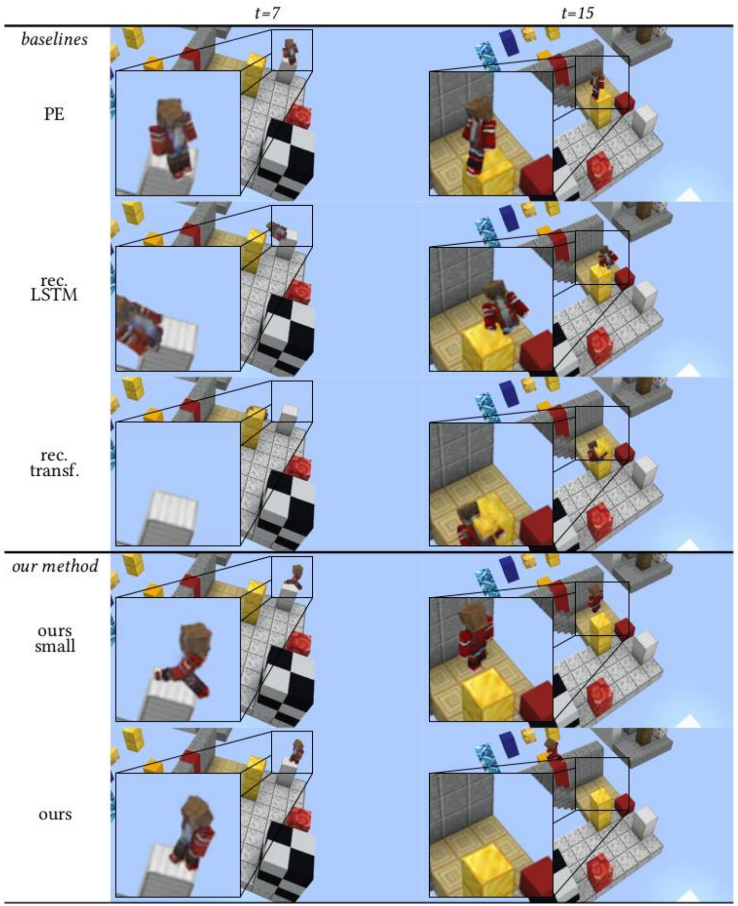


Evaluation Breakdown and Other Tests

max planck institut informatik

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



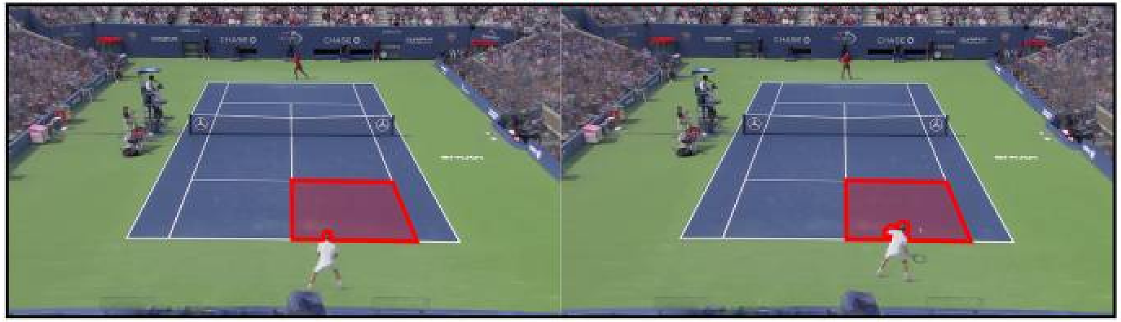


	Position L2↓	*Position* FD↓	*Root angle* L2↓	*Root angle* FD↓	*Joints 3D* L2↓	*Joints 3D* FD↓
Action conditioned video prediction						
PE	3.117	87.688	1.182	12.627	0.277	30.711
Rec. LSTM	1.753	7.413	1.100	8.416	0.234	18.455
Rec. Transf.	1.183	2.996	0.913	7.566	0.212	15.976
Ours Small	1.244	1.071	1.187	0.601	0.178	1.570
Ours	1.064	0.846	0.961	0.421	0.153	1.049
Unconditional video prediction						
PE	3.973	146.019	1.604	30.448	0.437	78.835
Rec. LSTM	2.064	11.283	1.224	14.860	0.264	28.736
Rec. Transf.	1.649	10.514	1.123	15.648	0.251	27.258
Ours Small	2.352	2.271	1.455	0.781	0.213	1.827
Ours	1.925	1.377	1.277	0.518	0.192	1.261
Opponent modeling						
PE	4.353	641.976	0.903	13.955	0.251	62.981
Rec. LSTM	1.581	5.507	0.697	2.517	0.143	10.443
Rec. Transf.	1.169	3.735	0.631	2.514	0.138	10.519
Ours Small	1.578	2.243	0.832	0.560	0.114	0.851
Ours	1.153	1.349	0.703	0.288	0.101	0.558
Sequence completion						
PE	1.720	40.766	0.814	6.783	0.246	40.441
Rec. LSTM	0.990	4.809	0.606	2.411	0.132	9.305
Rec. Transf.	0.294	0.364	0.403	1.623	0.100	5.628
Ours Small	0.344	0.187	0.581	0.301	0.088	0.765
Ours	0.252	0.143	0.437	0.198	0.069	0.478
Average						
PE	3.291	229.112	1.126	15.953	0.303	53.242
Rec. LSTM	1.597	7.253	0.907	7.051	0.193	16.735
Rec. Transf.	1.074	4.402	0.767	6.838	0.175	14.845
Ours Small	1.380	1.443	1.014	0.560	0.148	1.253
Ours	1.099	0.929	0.844	0.356	0.129	0.836

Implausible Actions

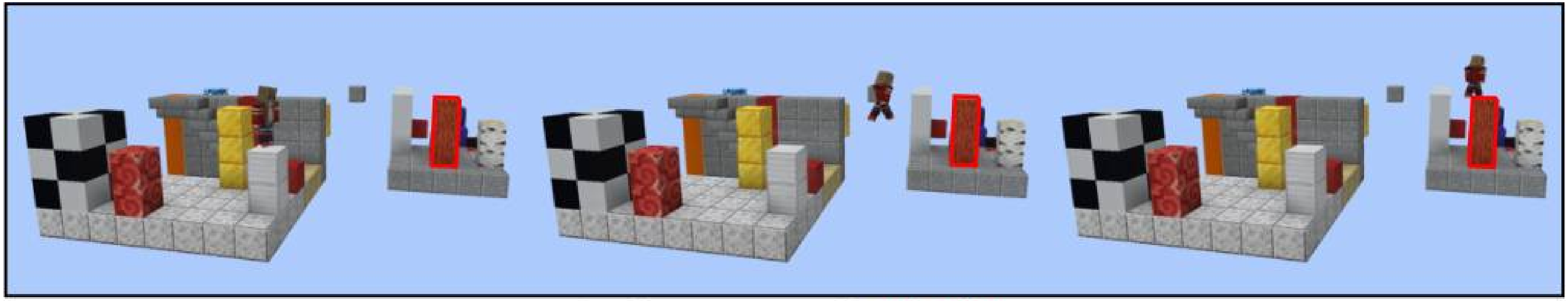
max planck institut
informatik

"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"

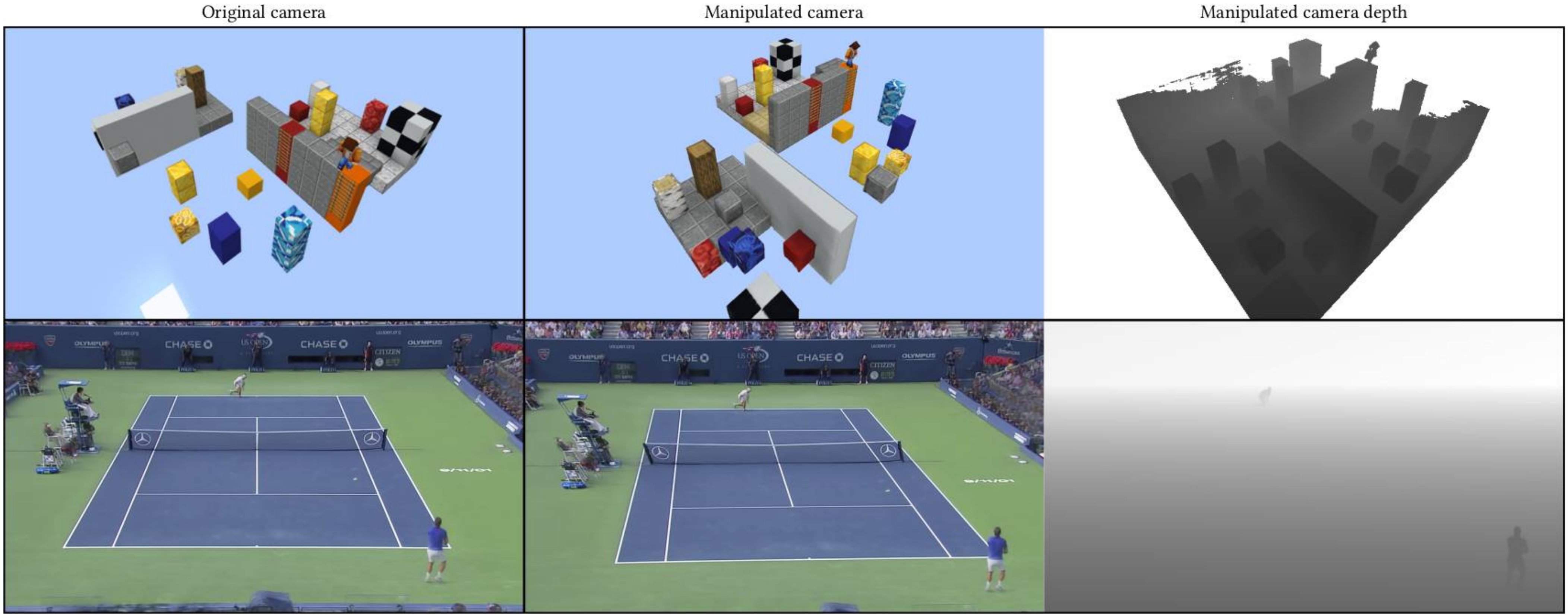
the left movement command is ignored



"The player jumps on the oak pillar"

Camera Manipulation

max planck institut informatik



Limitations

Novel scene views

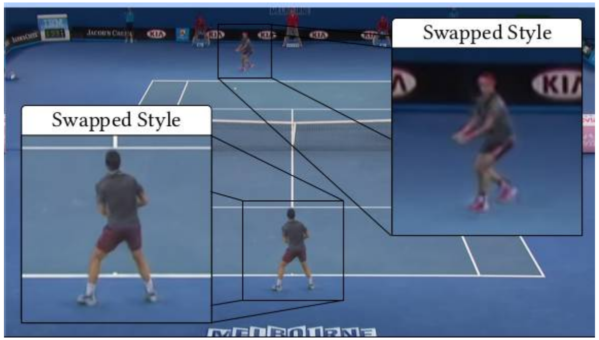
Foot sliding, slight jitter

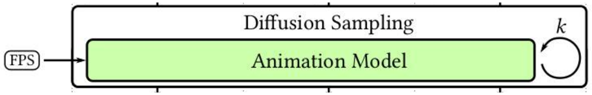
- No AM conditioning on scene geometry
- Tennis scenario: Overfitting with less than 60% of the data
- Foot sliding artefacts

- No explicit physics modelling (everything is learnt from data)
- Not real-time (AM: 1.08fps)

Conclusion and Take-home Messages

max planck institut
informatik

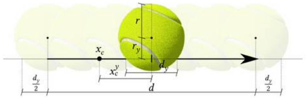


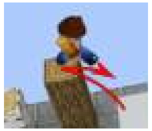


$$\epsilon_k^p = \mathcal{A}(\mathbf{s}_k^p | \mathbf{s}^c, \mathbf{a}^{\text{emb}}, \mathbf{m}^{\text{s}}, \mathbf{m}^{\text{a}}, k).$$

Style $\boldsymbol{\omega}$

Camera





- 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

max planck institut
informatik

Menapace *et al.*, arXiv:2303.13472

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

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

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

SIGGRAPH 2024
DENVER+ 28 JUL — 1 AUG

arXiv:2303.13472v3 [cs.CV] 21 Jan 2024

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

WILLI MENAPACE*, University of Trento, Italy
ALIAKSANDR SIAROHIN, Snap Inc., USA
STÉPHANE LATHUILIÈRE, LTCI, Télécom Paris, Institut Polytechnique de Paris, France
PANOS ACHLIOPTAS, Snap Inc., USA
VLADISLAV GOLYANIK, MPI for Informatics, SIC, 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 learned from annotated videos. Our PGM enables the generation of videos using prompts, a wide spectrum of conditioning signals such as player poses, object locations, and detailed textual actions (see) indicating what each player should do. Our *Animation Model* uses this information to generate future, past, or interpolated environment states according to the learned game dynamics. At this stage, the model is able to perform complex action reasoning such as generating a winning shot if the action "the [other] player does not catch the ball" is specified, as shown in the figure. To accomplish this goal, the model decides that the bottom player should hit the ball with a "lob" shot, sending the ball high above the opponent, who is unable to catch it. Our model renders the scene from a user-defined viewpoint (see) using a *Synthesis Model* where the style of the scene (see) can be controlled explicitly.

*Work performed while the author was an intern at Snap Inc.

Authors' addresses: Willi Menapace, University of Trento, Italy, willi.menapace@unitn.it; Aliaksandr Siarohin, Snap Inc., USA, asiarohin@snapchat.com; Stéphane Lathuilière, LTCI, Télécom Paris, Institut Polytechnique de Paris, France, stephane.lathuiliere@telecom-paris.fr; Panos Achlioptas, Snap Inc., USA, pachlioptas@gmail.com; Vladislav Golyanik, MPI for Informatics, SIC, Germany, golyanik@mpi-inf.mpg.de; Sergey Tulyakov, Snap Inc., USA, stulyakov@snapchat.com; Elisa Ricci, University of Trento, Fondazione Bruno Kessler, Italy, e.ricci@unitn.it.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
© 2023 Copyright held by the owner/author(s). Publication rights licensed to ACM.
0730-0301/2023/1-ART1 $15.00
https://doi.org/10.1145/3635705

Neural video game simulators emerged as powerful tools to generate and edit videos. Their idea is to represent games as the evolution of an environment's state driven by the actions of its agents. While such a paradigm enables users to *play* a game action-by-action, its rigidity precludes more semantic forms of

ACM Trans. Graph., Vol. 1, No. 1, Article 1. Publication date: January 2023.

Diffusion Models in Visual Computing

max planck institut informatik

State of the Art on Diffusion Models for Visual Computing

R. Po[1*] W. Yifan[1*] V. Golyanik[2*] K. Aberman[3] J. T. Barron[4] A. Bermano[5] E. Chan[1] T. Dekel[6] A. Holynski[4,7] A. Kanazawa[7] C. K. Liu[1] L. Liu[8] B. Mildenhall[4] M. Nießner[9] B. Ommer[10] C. Theobalt[2] P. Wonka[11] G. Wetzstein[1]

[1]Stanford University [2]MPI for Informatics and VIA Center [3]Snap Inc. [4]Google Research [5]Tel Aviv University [6]Weizmann Institute of Science [7]UC Berkley [8]University of Pennsylvania [9]TU Munich [10]LMU Munich [11]KAUST *Equal contribution

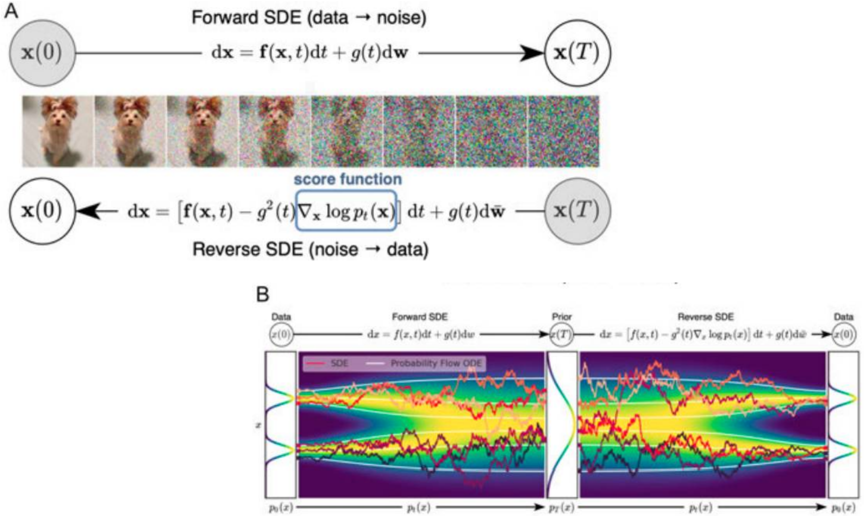
Figure 1: *This state-of-the-art report discusses the theory and practice of diffusion models for visual computing. These models have recently become the de-facto standard for image, video, 3D, and 4D generation and editing. Images adapted from [PJBM22, DMGT23, SSP*23b, MSP*23, BTOAF*22, HTE*23, Lab23, PW23, RLJ*22, MPE*23, Arn23] ©2023 IEEE.*

Abstract

The field of visual computing is rapidly advancing due to the emergence of generative artificial intelligence (AI), which unlocks unprecedented capabilities for the generation, editing, and reconstruction of images, videos, and 3D scenes. In these domains, diffusion models are the generative AI architecture of choice. Within the last year alone, the literature on diffusion-based tools and applications has seen exponential growth and relevant papers are published across the computer graphics, computer vision, and AI communities with new works appearing daily on arXiv. This rapid growth of the field makes it difficult to keep up with all recent developments. The goal of this state-of-the-art report (STAR) is to introduce the basic mathematical concepts of diffusion models, implementation details and design choices of the popular Stable Diffusion model, as well as overview important aspects of these generative AI tools, including personalization, conditioning, inversion, among others. Moreover, we give a comprehensive overview of the rapidly growing literature on diffusion-based generation and editing, categorized by the type of generated medium, including 2D images, videos, 3D objects, locomotion, and 4D scenes. Finally, we discuss available datasets, metrics, open challenges, and social implications. This STAR provides an intuitive starting point to explore this exciting topic for researchers, artists, and practitioners alike.

CCS Concepts

• *Computing methodologies → Computer graphics; Neural networks;*



Sec. 3 Fundamentals of Diffusion Models

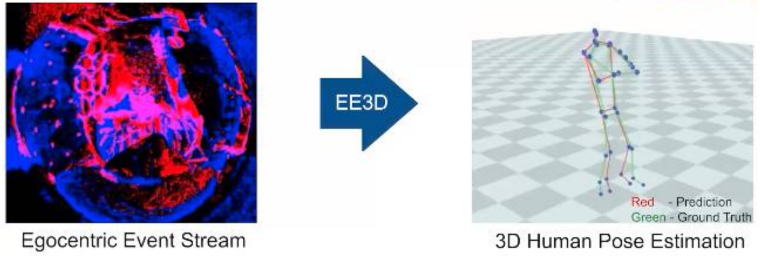
Sec. 7 Towards 4D Spatio-temporal Diffusion

7.3 4D Scene Generation and Editing

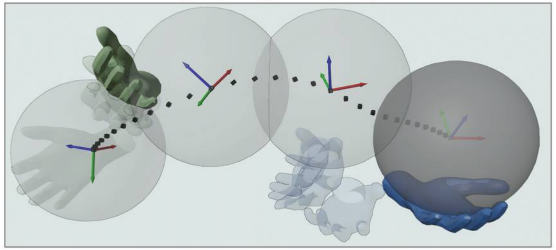
Po, Wang, Golyanik *et al.* EUROGRAPHICS, 2024.

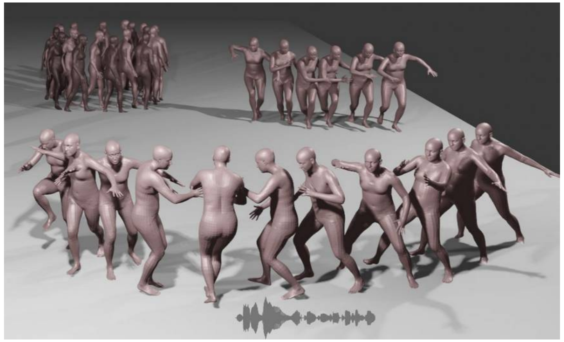
4DQV: Research Interests

max planck institut informatik

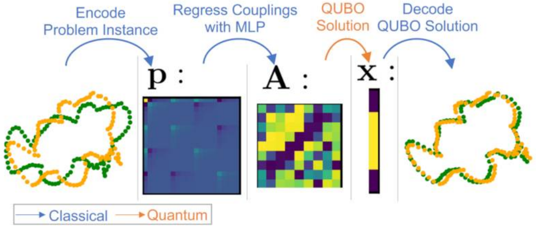


3D/4D Reconstruction and Neural Rendering





4D Generative Models



$$\frac{\partial L_{\text{gap}}(\mathbf{A})}{\partial \mathbf{A}_{i,j}} = 2\hat{\mathbf{x}}_i\hat{\mathbf{x}}_j - 2\mathbf{x}_i^*\mathbf{x}_j^* - 2\cancel{\frac{\partial \mathbf{x}^*(\mathbf{A})}{\partial \mathbf{A}_{i,j}}}\mathbf{A}\mathbf{x}^*$$

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?