# Learning to Build and Interact with 3D Rooms using Deep Neural Networks

Despoina Paschalidou

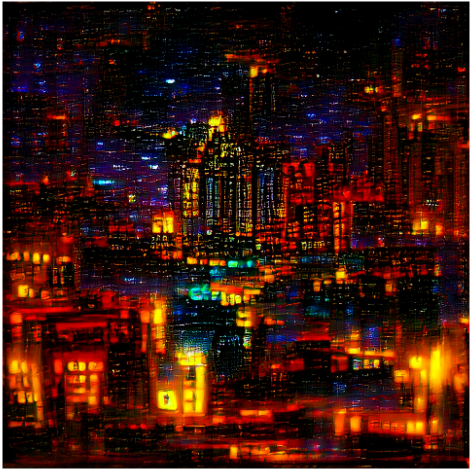
NVIDIA GTC 2022

Max Planck Institute for Intelligent Systems
Autonomous Vision Group

CVL Computer Vision Lab
ETH zürich

# Generative Models are Great!

*An abstract painting of a planet ruled by little castles*
**Image Source**:@RiversHaveWings on Twitter



*A city scape at night*
**Image Source**:@RiversHaveWings on Twitter

Image Generated with NVIDIA's Hyper-Realistic Face Generator StyleGAN

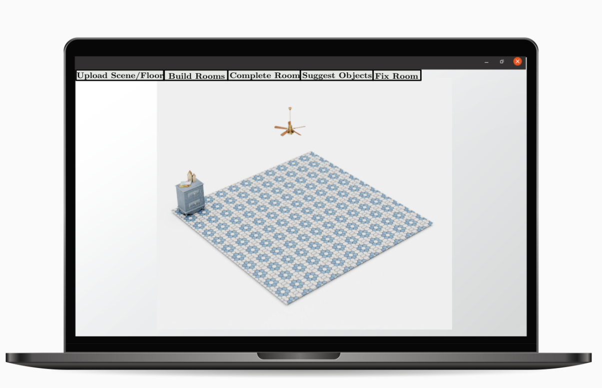

**Image Source:** NVIDIA Drive Sim

Image Source: Oculus


**Image Source:** Promethean AI

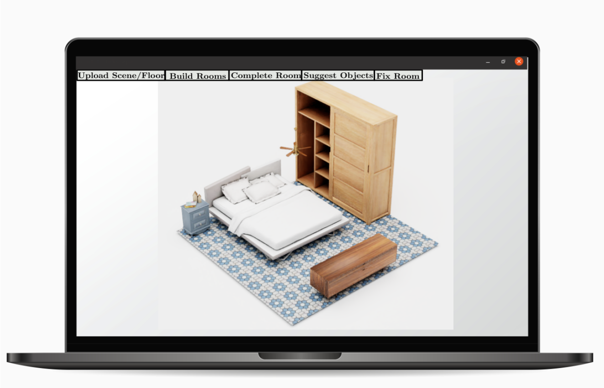
# Generative Models are Great!

Can we learn a **generative model** for **indoor scene synthesis** that allows performing a number of **interactive scenarios** with versatile user input?

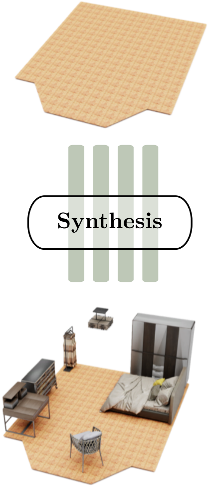


# Generative Models are Great!

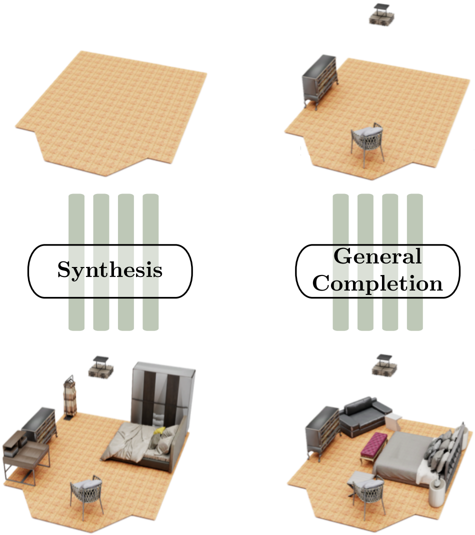
Can we learn a **generative model** for **indoor scene synthesis** that allows performing a number of **interactive scenarios** with versatile user input?



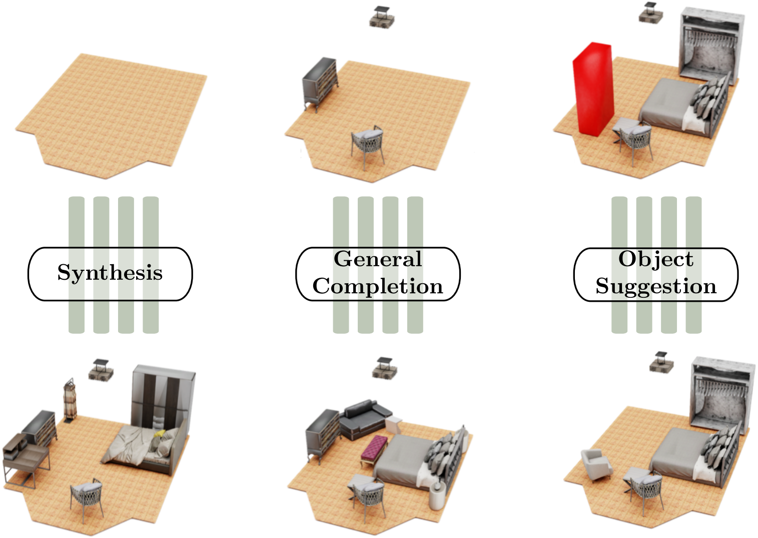
# Motivation



# Motivation

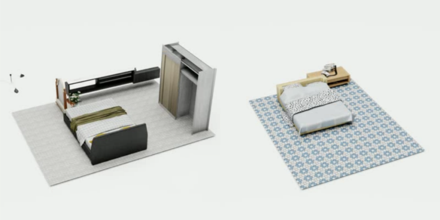


# Motivation



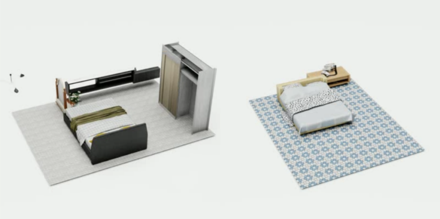
Existing scene synthesis methods **impose unnatural constraints on the scene generation process** because they represent **scenes as ordered sequences of objects**.

FastSynth, Ritchie et al. CVPR 2019



SceneFormer, Wang et al. ARXIV 2020

Existing scene synthesis methods **impose unnatural constraints on the scene generation process** because they represent **scenes as ordered sequences of objects**.



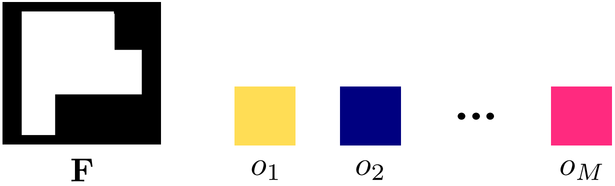
FastSynth, Ritchie et al. CVPR 2019

SceneFormer, Wang et al. ARXIV 2020

We pose scene synthesis as an **unordered set generation problem**.

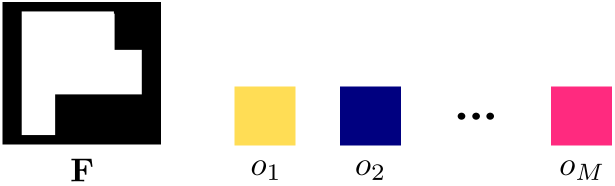
# Scene Parametrization

A scene comprises an **unordered set of** $M$ **objects** $\mathcal{O} = \{o_j\}_{j=1}^{M}$ and its **floor shape** $\mathbf{F}$.



# Scene Parametrization

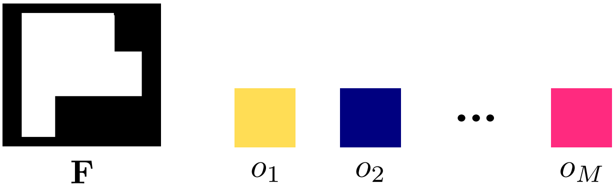
A scene comprises an **unordered set of** $M$ **objects** $\mathcal{O} = \{o_j\}_{j=1}^{M}$ and its **floor shape** $\mathbf{F}$.




Each object $o_j = \{\mathbf{c}_j, \mathrm{s}_j, \mathbf{r}_j, \mathbf{t}_j\}$ is modelled with four random variables that describe their **category, size, orientation and location**.

# Scene Parametrization

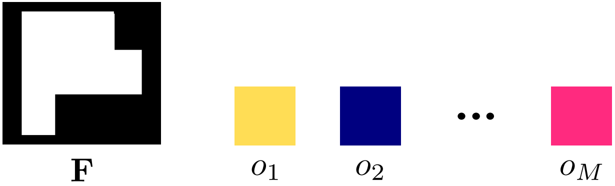
A scene comprises an **unordered set of $M$ objects** $\mathcal{O} = \{o_j\}_{j=1}^{M}$ and its **floor shape $\mathbf{F}$**.




Each object $o_j = \{\mathbf{c}_j, \mathrm{s}_j, \mathbf{r}_j, \mathbf{t}_j\}$ is modelled with four random variables that describe their **category, size, orientation and location**.

# Scene Parametrization

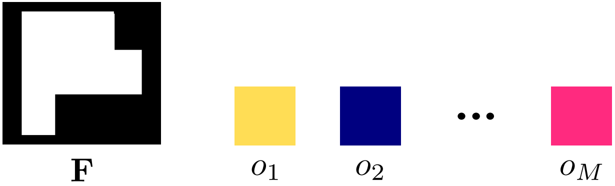
A scene comprises an **unordered set of** $M$ **objects** $\mathcal{O} = \{o_j\}_{j=1}^{M}$ and its **floor shape** $\mathbf{F}$.




Each object $o_j = \{\mathbf{c}_j, \mathbf{s}_j, \mathbf{r}_j, \mathbf{t}_j\}$ is modelled with four random variables that describe their **category, size, orientation and location**.

# Scene Parametrization

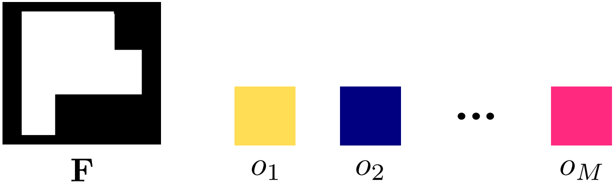
A scene comprises an **unordered set of** $M$ **objects** $\mathcal{O} = \{o_j\}_{j=1}^{M}$ and its **floor shape** $\mathbf{F}$.




Each object $o_j = \{\mathbf{c}_j, \mathrm{s}_j, \mathbf{r}_j, \mathbf{t}_j\}$ is modelled with four random variables that describe their **category, size, orientation and location**.

# Scene Parametrization

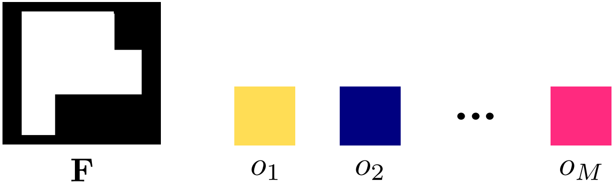
A scene comprises an **unordered set of** $M$ **objects** $\mathcal{O} = \{o_j\}_{j=1}^{M}$ and its **floor shape** $\mathbf{F}$.




Each object $o_j = \{\mathbf{c}_j, \mathrm{s}_j, \mathbf{r}_j, \mathrm{t}_j\}$ is modelled with four random variables that describe their **category, size, orientation and location**.

# Scene Parametrization

A scene comprises an **unordered set of $M$ objects** $\mathcal{O} = \{o_j\}_{j=1}^{M}$ and its **floor shape $\mathbf{F}$**.

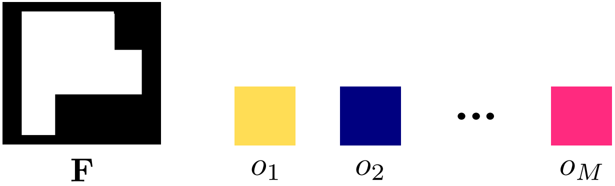



Each object $o_j = \{\mathbf{c}_j, \mathrm{s}_j, \mathbf{r}_j, \mathbf{t}_j\}$ is modelled with four random variables that describe their **category, size, orientation and location**.

$$\underbrace{p_\theta(o_j \mid o_{<j}, \mathbf{F})}_{\text{Probability of generating j-th object}} = p_\theta(\mathbf{c}_j | o_{<j}, \mathbf{F}) p_\theta(\mathbf{t}_j | \mathbf{c}_j, o_{<j}, \mathbf{F}) p_\theta(\mathbf{r}_j | \mathbf{c}_j, \mathbf{t}_j, o_{<j}, \mathbf{F}) p_\theta(\mathrm{s}_j | \mathbf{c}_j, \mathbf{t}_j, \mathbf{r}_j, o_{<j}, \mathbf{F})$$

# Scene Parametrization

A scene comprises an **unordered set of $M$ objects** $\mathcal{O} = \{o_j\}_{j=1}^{M}$ and its **floor shape** $\mathbf{F}$.



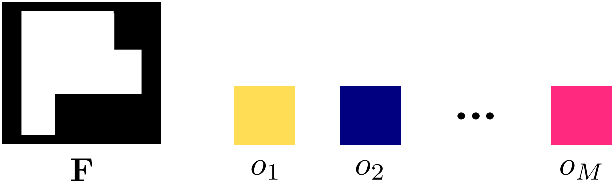

The **likelihood** of generating a scene **with any order** is:

$$\underbrace{p_\theta(\mathcal{O}|\mathbf{F})}_{\substack{\text{Probability of generating } \mathcal{O} \\ \text{with any order}}} = \sum_{\hat{\mathcal{O}} \in \pi(\mathcal{O})} \underbrace{\prod_{j \in \hat{\mathcal{O}}} p_\theta(o_j \mid o_{<j}, \mathbf{F})}_{\substack{\text{Probability of generating } \mathcal{O} \\ \text{with order } \hat{\mathcal{O}}}}$$

where $\pi(\mathcal{O})$ is a a permutation function that computes the set of permutations of all objects $\mathcal{O}$ in the scene.

# Scene Parametrization

A scene comprises an **unordered set of** $M$ **objects** $\mathcal{O} = \{o_j\}_{j=1}^{M}$ and its **floor shape** $\mathbf{F}$.



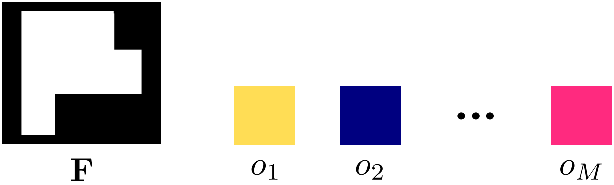

The **likelihood** of generating a scene **with any order** is:

$$\underbrace{p_\theta(\mathcal{O}|\mathbf{F})}_{\substack{\text{Probability of generating } \mathcal{O} \\ \text{with any order}}} = \sum_{\hat{\mathcal{O}} \in \pi(\mathcal{O})} \underbrace{\prod_{j \in \hat{\mathcal{O}}} p_\theta(o_j \mid o_{<j}, \mathbf{F})}_{\substack{\text{Probability of generating } \mathcal{O} \\ \text{with order } \hat{\mathcal{O}}}}$$

where $\pi(\mathcal{O})$ is a a permutation function that computes the set of permutations of all objects $\mathcal{O}$ in the scene.

# Scene Parametrization

A scene comprises an **unordered set of $M$ objects** $\mathcal{O} = \{o_j\}_{j=1}^{M}$ and its **floor shape** $\mathbf{F}$.

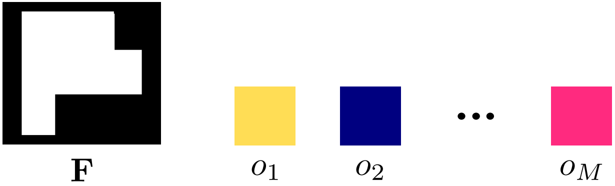



The **likelihood** of generating a scene **with any order** is:

$$\underbrace{p_\theta(\mathcal{O}|\mathbf{F})}_{\substack{\text{Probability of generating } \mathcal{O} \\ \text{with any order}}} = \sum_{\hat{\mathcal{O}} \in \pi(\mathcal{O})} \underbrace{\prod_{j \in \hat{\mathcal{O}}} p_\theta(o_j \mid o_{<j}, \mathbf{F})}_{\substack{\text{Probability of generating } \mathcal{O} \\ \text{with order } \hat{\mathcal{O}}}}$$

# Scene Parametrization

A scene comprises an **unordered set of $M$ objects** $\mathcal{O} = \{o_j\}_{j=1}^{M}$ and its **floor shape** $\mathbf{F}$.



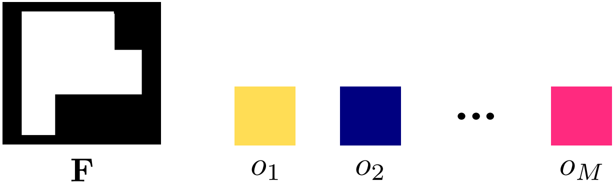

The **likelihood** of generating a scene **with all orders** is:

$$\underbrace{\hat{p}_\theta(\mathcal{O}|\mathbf{F})}_{\substack{\text{Probability of generating } \mathcal{O} \\ \text{with all orders}}} = \prod_{\hat{\mathcal{O}} \in \pi(\mathcal{O})} \underbrace{\prod_{j \in \hat{\mathcal{O}}} p_\theta(o_j \mid o_{<j}, \mathbf{F})}_{\substack{\text{Probability of generating } \mathcal{O} \\ \text{with order } \hat{\mathcal{O}}}}$$

ATISS is trained to **maximize the log-likelihood of all possible permutations of object arrangements** in a collection of scenes.

## Scene Parametrization

A scene comprises an **unordered set of $M$ objects** $\mathcal{O} = \{o_j\}_{j=1}^{M}$ and its **floor shape $\mathbf{F}$**.

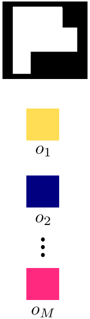



The **log-likelihood** of generating a scene **with all orders** is:

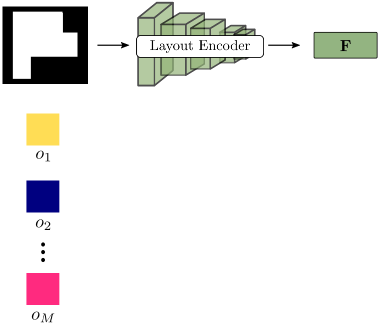
$$\underbrace{\log \hat{p}_\theta(\mathcal{O}|\mathbf{F})}_{\substack{\text{Log-likelihood of generating } \mathcal{O} \\ \text{with all orders}}} = \sum_{\hat{\mathcal{O}} \in \pi(\mathcal{O})} \underbrace{\sum_{j \in \hat{\mathcal{O}}} \log p_\theta(o_j \mid o_{<j}, \mathbf{F})}_{\substack{\text{Probability of generating } \mathcal{O} \\ \text{with order } \hat{\mathcal{O}}}}$$

ATISS is trained to **maximize the log-likelihood of all possible permutations of object arrangements** in a collection of scenes.

# Scene Generation

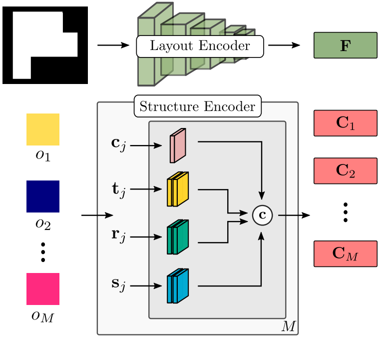


# Scene Generation



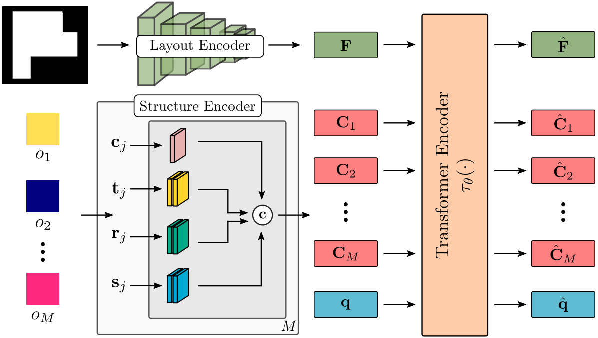

- **Layout encoder**: Computes a global feature representation for the floor.

# Scene Generation



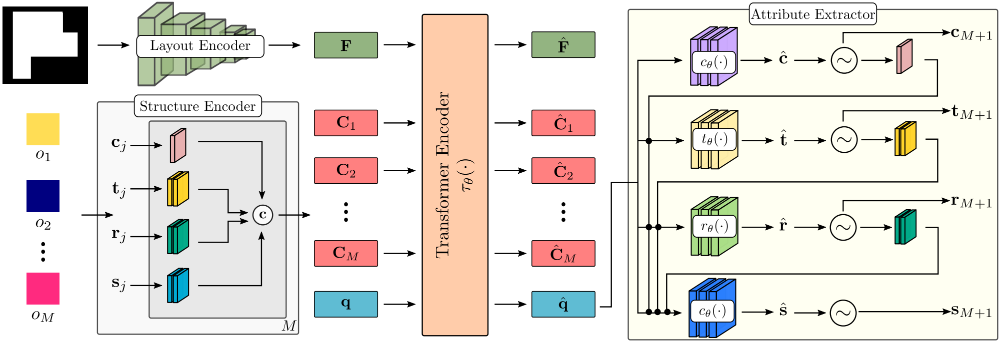

- **Layout encoder**: Computes a global feature representation for the floor.
- **Structure encoder**: Maps the j-th object to a per-object context embedding $\mathbf{C}_j$.

# Scene Generation



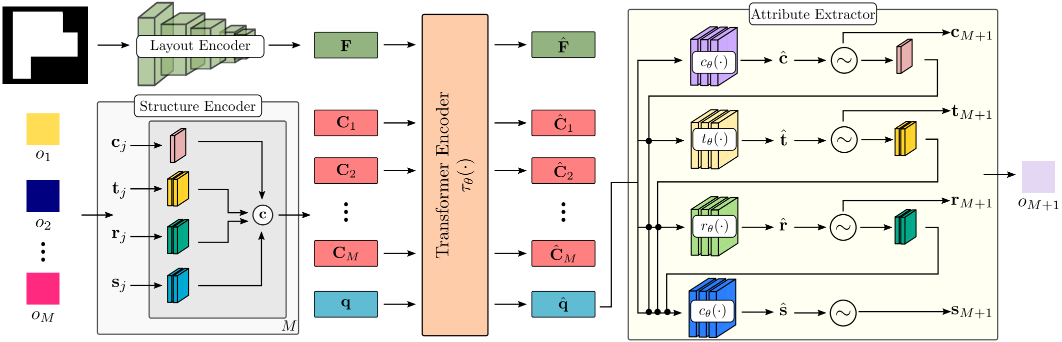

- **Layout encoder**: Computes a global feature representation for the floor.
- **Structure encoder**: Maps the j-th object to a per-object context embedding $\mathbf{C}_j$.
- **Transformer encoder**: Takes $\mathbf{F}, \{\mathbf{C}_j\}_{j=1}^{M}, \mathbf{q}$ and predicts the features $\hat{\mathbf{q}}$ of the next object to be added in the scene.

# Scene Generation



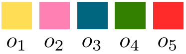

- **Layout encoder**: Computes a global feature representation for the floor.
- **Structure encoder**: Maps the j-th object to a per-object context embedding $\mathbf{C}_j$.
- **Transformer encoder**: Takes $\mathbf{F}, \{\mathbf{C}_j\}_{j=1}^{M}, \mathbf{q}$ and predicts the features $\hat{\mathbf{q}}$ of the next object to be added in the scene.
- **Attribute extractor**: Predicts the object attributes of the next object.

# Scene Generation

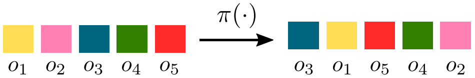



- **Layout encoder**: Computes a global feature representation for the floor.
- **Structure encoder**: Maps the j-th object to a per-object context embedding $\mathbf{C}_j$.
- **Transformer encoder**: Takes $\mathbf{F}, \{\mathbf{C}_j\}_{j=1}^{M}, \mathbf{q}$ and predicts the features $\hat{\mathbf{q}}$ of the next object to be added in the scene.
- **Attribute extractor**: Predicts the object attributes of the next object.

# Training Overview

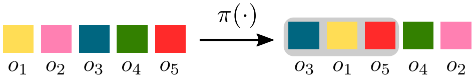


# Training Overview



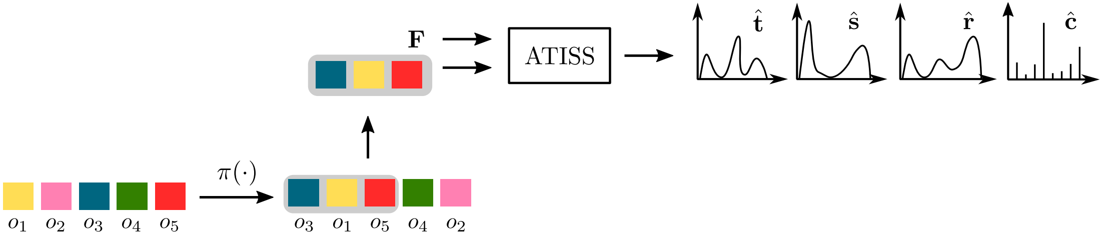

- Randomly permute the $M$ objects of a scene.

# Training Overview



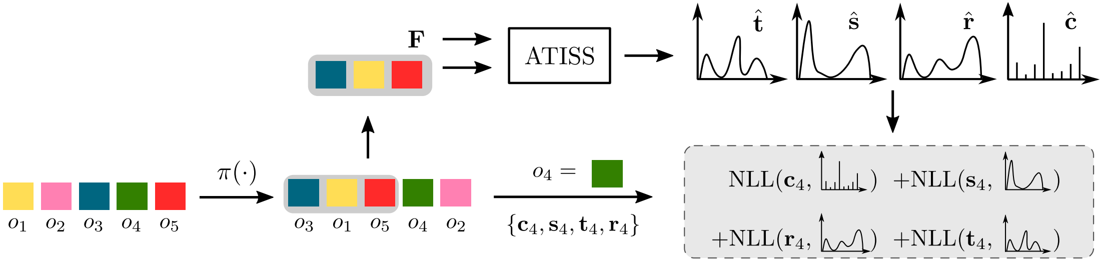

- Randomly permute the $M$ objects of a scene.
- Randomly select the first $T$ objects to compute the context embedding $\mathbf{C}$.

# Training Overview




- Randomly permute the $M$ objects of a scene.
- Randomly select the first $T$ objects to compute the context embedding $\mathbf{C}$.
- Conditioned on the $\mathbf{C}$ and $\mathbf{F}$, ATISS **predicts the attribute distributions of the next object**.

# Training Overview



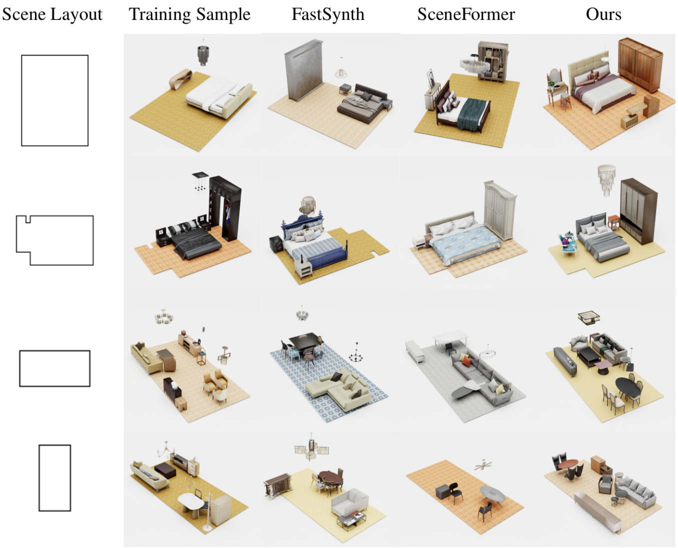

- Randomly permute the $M$ objects of a scene.
- Randomly select the first $T$ objects to compute the context embedding $\mathbf{C}$.
- Conditioned on the $\mathbf{C}$ and $\mathbf{F}$, ATISS **predicts the attribute distributions of the next object**.
- ATISS is trained to maximize the log likelihood of the $T + 1$ object from the permuted set of objects.

How well does it work?

# Scene Synthesis Results

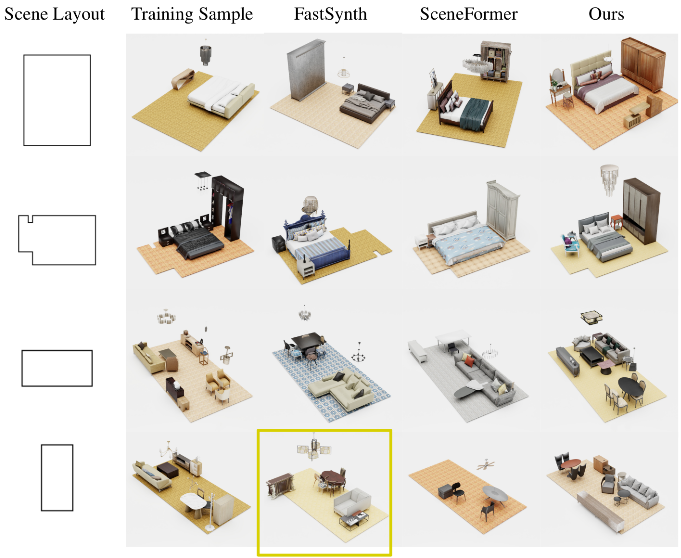
The scenes were rendered using NVIDIA OMNIVERSE.

# Scene Synthesis Results



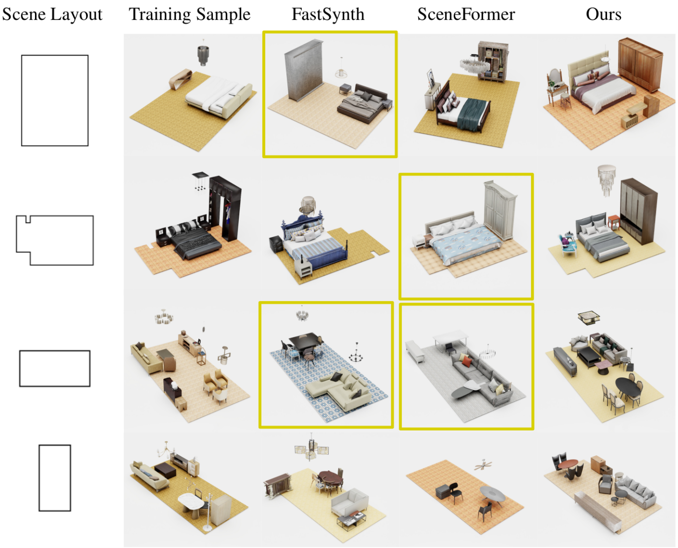

The scenes were rendered using NVIDIA OMNIVERSE.

# Scene Synthesis Results



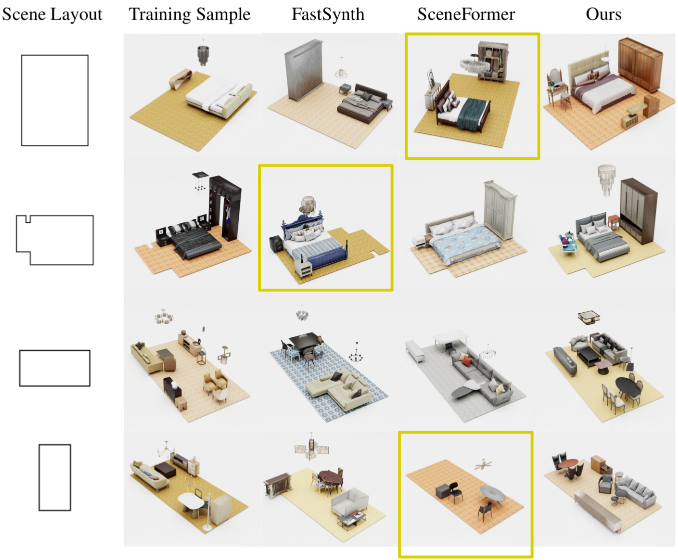

The scenes were rendered using NVIDIA OMNIVERSE.

# Scene Synthesis Results



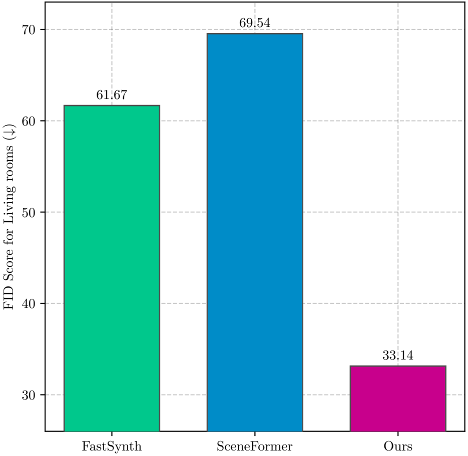

The scenes were rendered using NVIDIA OMNIVERSE.

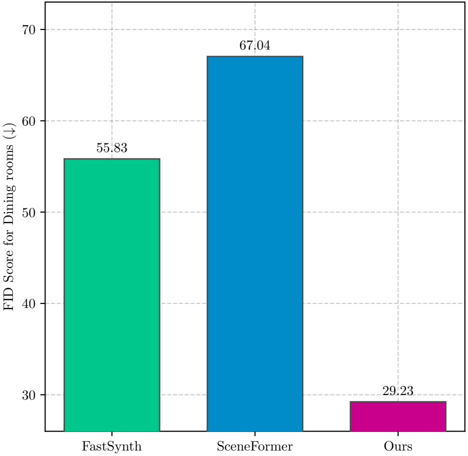
# Scene Synthesis Results



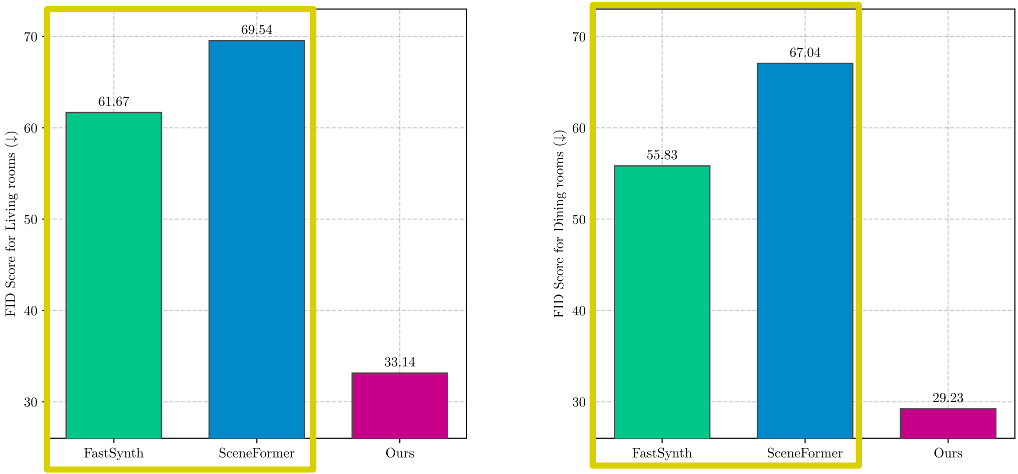

The scenes were rendered using NVIDIA OMNIVERSE.

# Scene Synthesis Quantitative Results



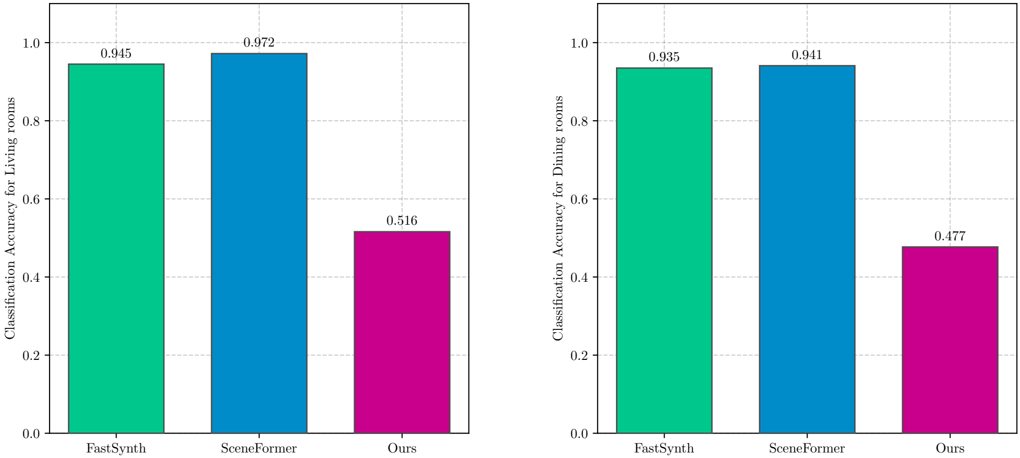



# Scene Synthesis Quantitative Results

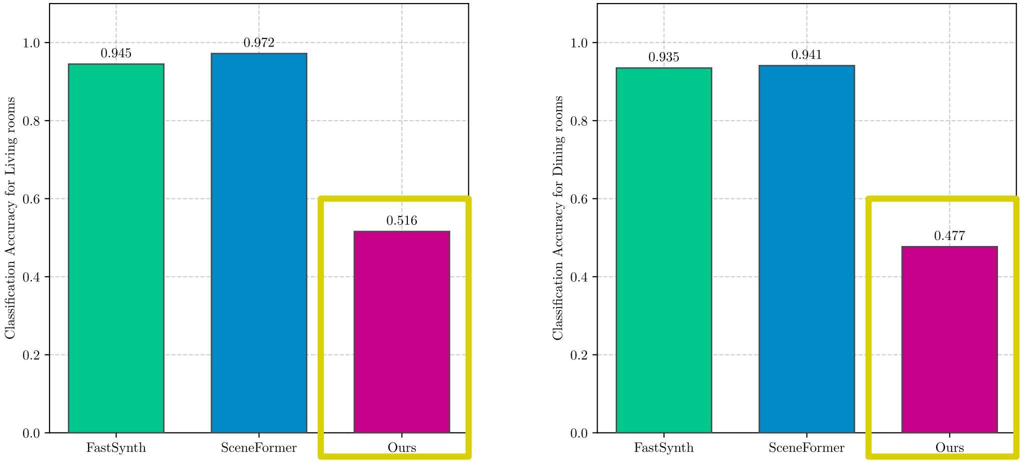



Our model achieves a **lower FID score** for all room types.

# Scene Synthesis Quantitative Results

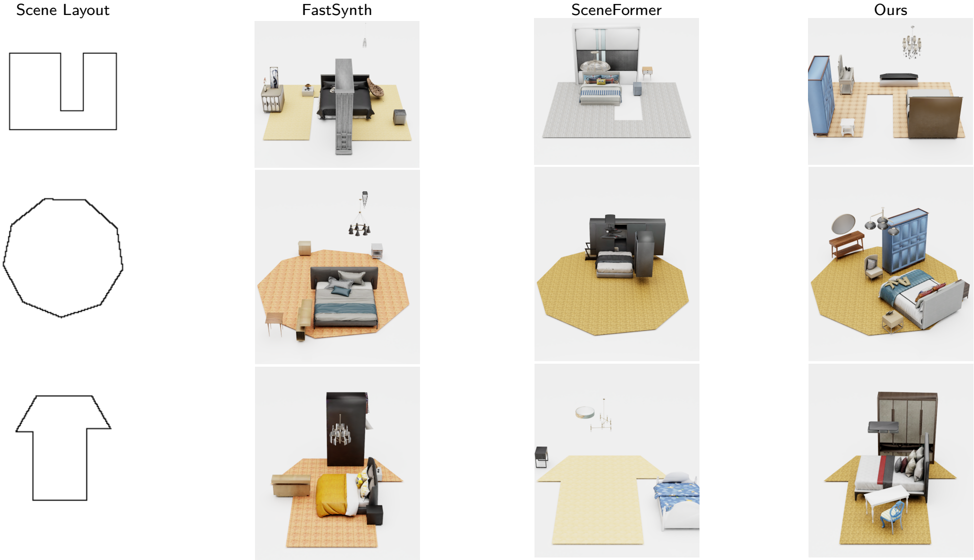


# Scene Synthesis Quantitative Results



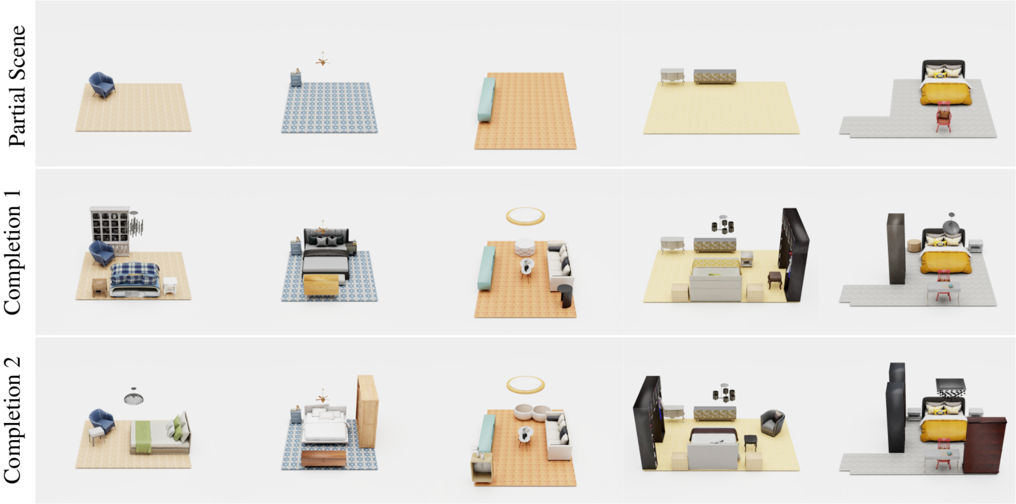

Our model achieves a **classification accouracy closer to 0.5** for all room types.

# Generalization Beyond Training Data



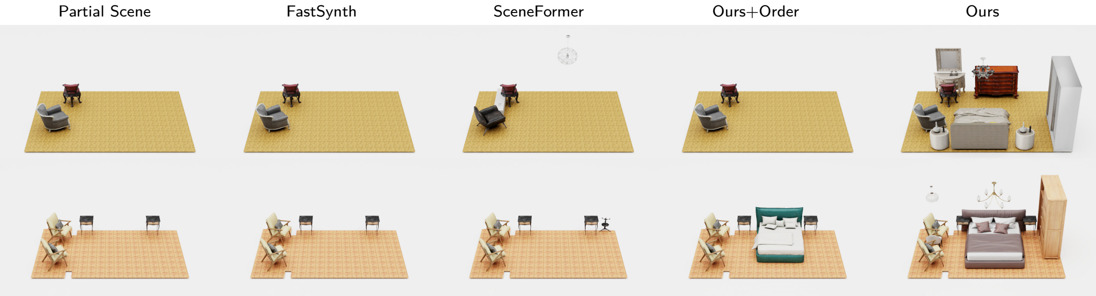

The scenes were rendered using NVIDIA OMNIVERSE.

# Scene Completion Results



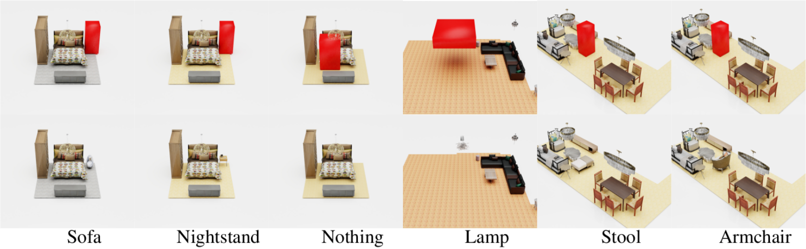

The scenes were rendered using NVIDIA OMNIVERSE.

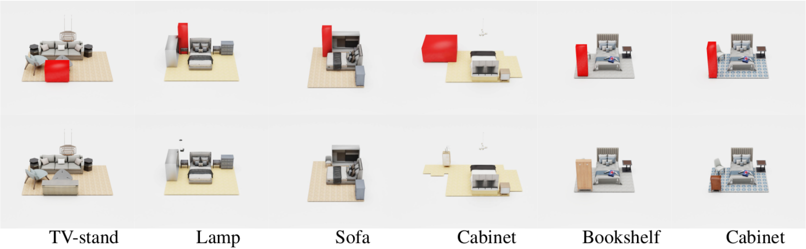
# Scene Completion Results




Since FastSynth, SceneFormer, and Ours+Order were trained with ordered sequences of objects, **they can only generate objects in the order they were trained with.**

# Objects Suggestion Results



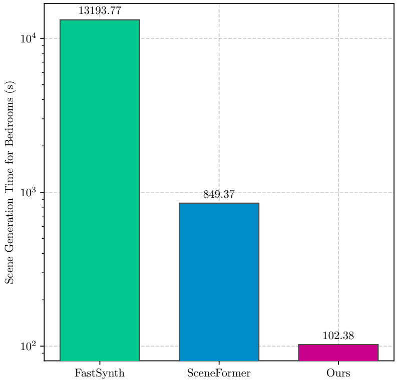



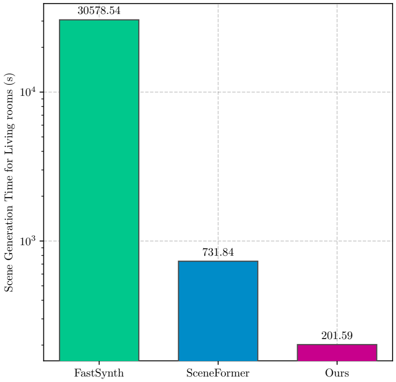

The scenes were rendered using NVIDIA OMNIVERSE.

# Failure Cases Correction Results

The scenes were rendered using NVIDIA OMNIVERSE.

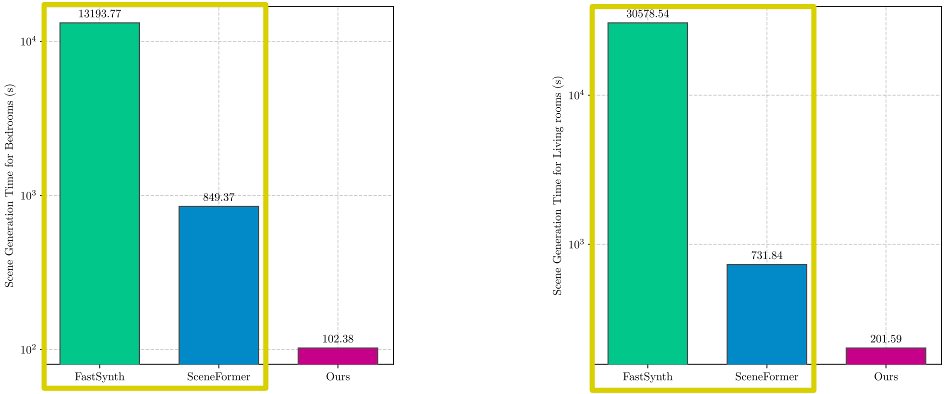
# Generation Time




- At least $100\times$ faster than the CNN-based FastSynth for all room types.
- At least $4\times$ faster than the Transformer-based SceneFormer for all room types.

# Generation Time




- At least $100\times$ faster than the CNN-based FastSynth for all room types.
- At least $4\times$ faster than the Transformer-based SceneFormer for all room types.

# Conclusions

- We propose ATISS **a novel autoregressive model for unordered set generation**.

# Conclusions

- We propose ATISS **a novel autoregressive model for unordered set generation**.
- We demonstrate that our unordered set formulation **opens up multiple interactive applications**.

# Conclusions

- We propose ATISS **a novel autoregressive model for unordered set generation**.
- We demonstrate that our unordered set formulation **opens up multiple interactive applications**.
- ATISS has fewer parameters, **is simpler to implement and train and runs up to 8x faster** than existing methods.

# Conclusions

- We propose ATISS **a novel autoregressive model for unordered set generation**.
- We demonstrate that our unordered set formulation **opens up multiple interactive applications**.
- ATISS has fewer parameters, **is simpler to implement and train and runs up to 8x faster** than existing methods.
- Limitations:

# Conclusions

- We propose ATISS **a novel autoregressive model for unordered set generation**.
- We demonstrate that our unordered set formulation **opens up multiple interactive applications**.
- ATISS has fewer parameters, **is simpler to implement and train and runs up to 8x faster** than existing methods.
- Limitations:
  - The autoregressive generation of attributes need to follow a specific ordering.

# Conclusions

- We propose ATISS **a novel autoregressive model for unordered set generation**.
- We demonstrate that our unordered set formulation **opens up multiple interactive applications**.
- ATISS has fewer parameters, **is simpler to implement and train and runs up to 8x faster** than existing methods.
- Limitations:
  - The autoregressive generation of attributes need to follow a specific ordering.
  - Separate object retrieval module.

# Thank you!

https://nv-tlabs.github.io/ATISS