# High-Resolution Image Synthesis with Latent Diffusion Models

San Zhang[1,2], Si Li[2], and Wu Wang[3]

Email: zhangsan@ia.ac.cn

[1]School of Artificial Intelligence, University of Chinese Academy of Sciences
[2,3]Institute of Automation, Chinese Academy of Sciences

Nov 20, 2024

中国科学院大学
University of Chinese Academy of Sciences

# Contents


1. Introduction
2. System Model
   - Network Model
   - Task Processing Model
3. Problem Formulation
4. Algorithm Design
5. Simulation Results
6. Conclusion

1 Introduction

2 System Model
- Network Model
- Task Processing Model

3 Problem Formulation

4 Algorithm Design

5 Simulation Results

6 Conclusion

# The Development of AI-Generated Content (AIGC)

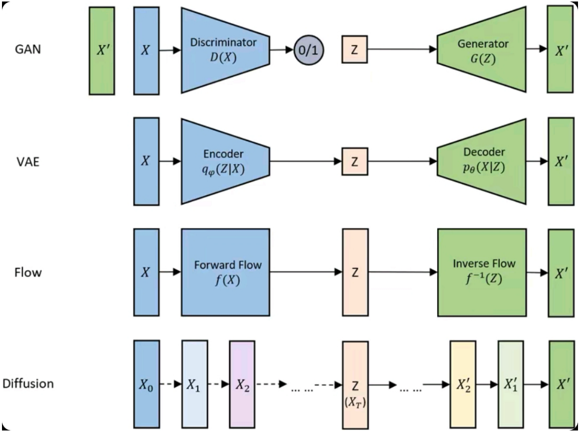


Figure 1: The development of AIGC and mobile edge computing network [1].

[1] M. Xu, et al., "Unleashing the power of edge-cloud generative AI in mobile networks: A survey of AIGC services," IEEE Commun. Surv. Tutor., Early Access, 2024.

# Diffusion Model

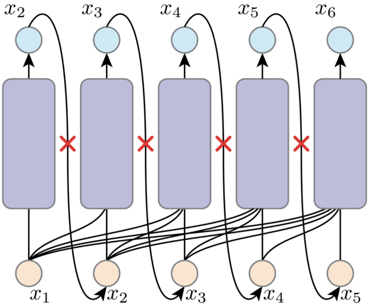


Figure 2: An illustration of diffusion model [2].

- With Gaussian noise as input
- The quality of generated images gets progressively better.

[2] H. Du, et al., "Enhancing deep reinforcement learning: A tutorial on generative diffusion models in network optimization," arXiv preprint arXiv:2308.05384, 2023.

# AIGC in Mobile Edge Computing (MEC)

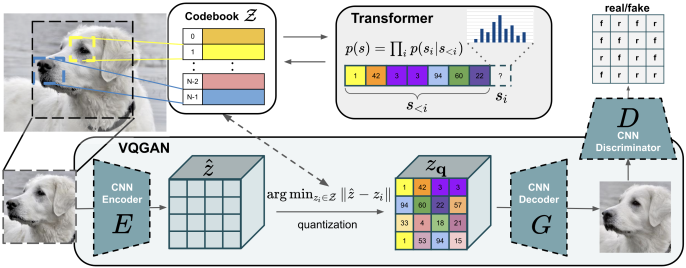


Figure 3: An overview of a mobile AIGC network [1].

- AIGC models can be deployed at edge servers.

[1] M. Xu, et al., "Unleashing the power of edge-cloud generative AI in mobile networks: A survey of AIGC services," IEEE Commun. Surv. Tutor., Early Access, 2024.

1 Introduction

2 System Model
- Network Model
- Task Processing Model

3 Problem Formulation

4 Algorithm Design

5 Simulation Results

6 Conclusion

# MEC Network

# Local Processing Model

# Home BS Processing Model

# Neighbor BS Processing Model

1 Introduction

2 System Model
- Network Model
- Task Processing Model

3 Problem Formulation

4 Algorithm Design

5 Simulation Results

6 Conclusion

# Weighted Cost

# Offloading Problem

1 Introduction

2 System Model
- Network Model
- Task Processing Model

3 Problem Formulation

4 Algorithm Design

5 Simulation Results

6 Conclusion

# Deep Reinforcement Learning based OSI Algorithm

- State:

$$\mathrm{s}_{\mathrm{n}}^{(\mathrm{l})}=\left(\mathrm{B}_{\mathrm{n}}^{(\mathrm{l})},\mathrm{q}_{\mathrm{n}}^{(\mathrm{l})},\mathrm{f}_{\mathrm{n}}^{\mathrm{U},(\mathrm{l})},\mathrm{g}^{\mathrm{B},(\mathrm{l})},\mathrm{h}_{\mathrm{n}}^{(\mathrm{l})}\right)$$

- Action:

$$\mathrm{a}_{\mathrm{n}}^{(\mathrm{l})}=\left(\mathrm{x}_{\mathrm{n}}^{(\mathrm{l})},\mathrm{y}_{\mathrm{n}}^{(\mathrm{l})},\mathrm{c}_{\mathrm{n}}^{(\mathrm{l})}\right)$$

- Reward:

$$\mathrm{r}_{\mathrm{n}}^{(\mathrm{l})}=-\sum_{\mathrm{n}\in\mathcal{N}}\left(\omega_1\mathrm{T}_{\mathrm{n}}^{(\mathrm{l})}+\omega_2\mathrm{E}_{\mathrm{n}}^{(\mathrm{l})}+\omega_3\epsilon_{\mathrm{n}}^{(\mathrm{l})}\right)-\mathrm{r}_{\mathrm{n}}^{\mathrm{P},(\mathrm{l})}$$

1 Introduction

2 System Model
- Network Model
- Task Processing Model

3 Problem Formulation

4 Algorithm Design

5 Simulation Results

6 Conclusion

1. Introduction
2. System Model
   - Network Model
   - Task Processing Model
3. Problem Formulation
4. Algorithm Design
5. Simulation Results
6. Conclusion

- Conclusion 1.
- Conclusion 2.
- Conclusion 3.
- Conclusion 4.

# Acknowledgements

Thanks for your listening!

Please feel free to contact us:

zhangsan@ia.ac.cn

University of Chinese Academy of Sciences