NMSL@NTHU

Networking and Multimedia Systems Lab

A Blind Streaming System for Multi-client Online 6-DoF View Touring

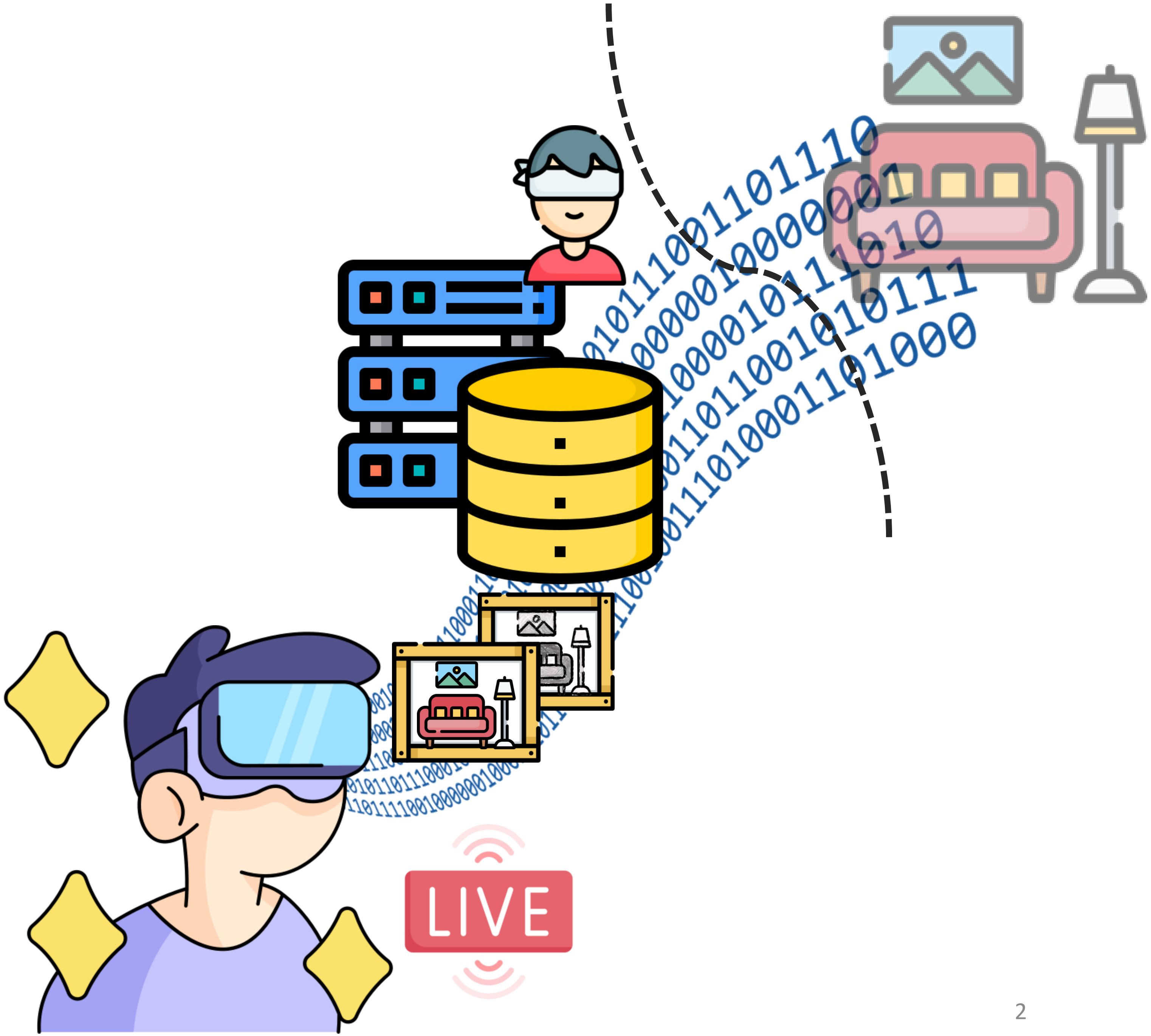
ShengMing (shengming0308@gapp.nthu.edu.tw)

Advisor: Cheng-Hsin Hsu

Networking and Multimedia Systems Lab, CS, National Tsing Hua University

Outline

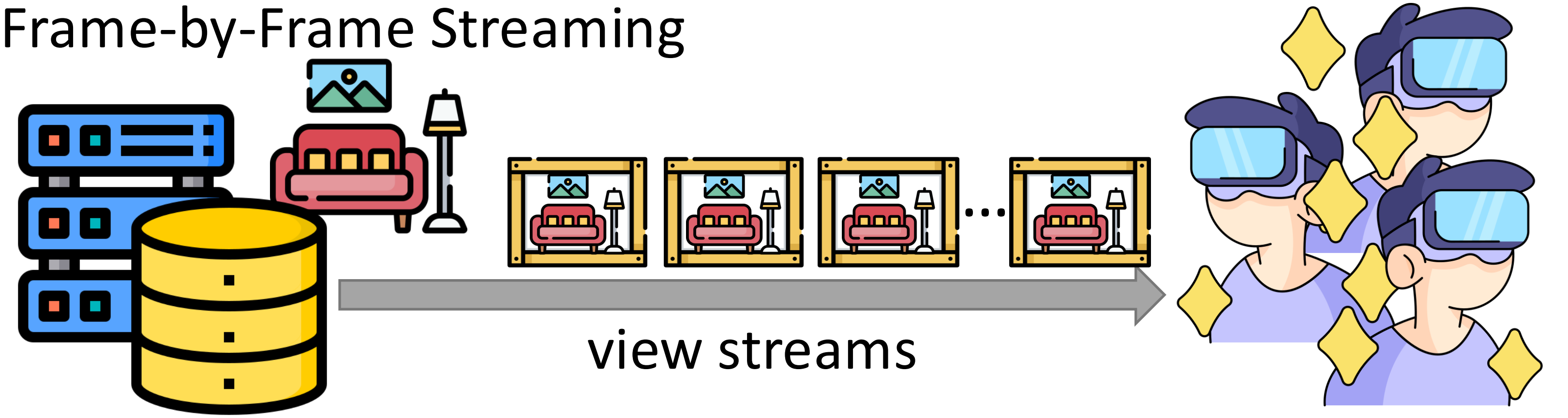
- Inspiration
- Goal & Challenges
- Related Work
- System Design
- Novel View Optimization
- Cloud Service Provider
  - Pose Predictor
  - Candidate Generator
  - Coverage Estimator
  - Solver & Algorithms
- Implementation
- Evaluations
- Conclusion & Future Work



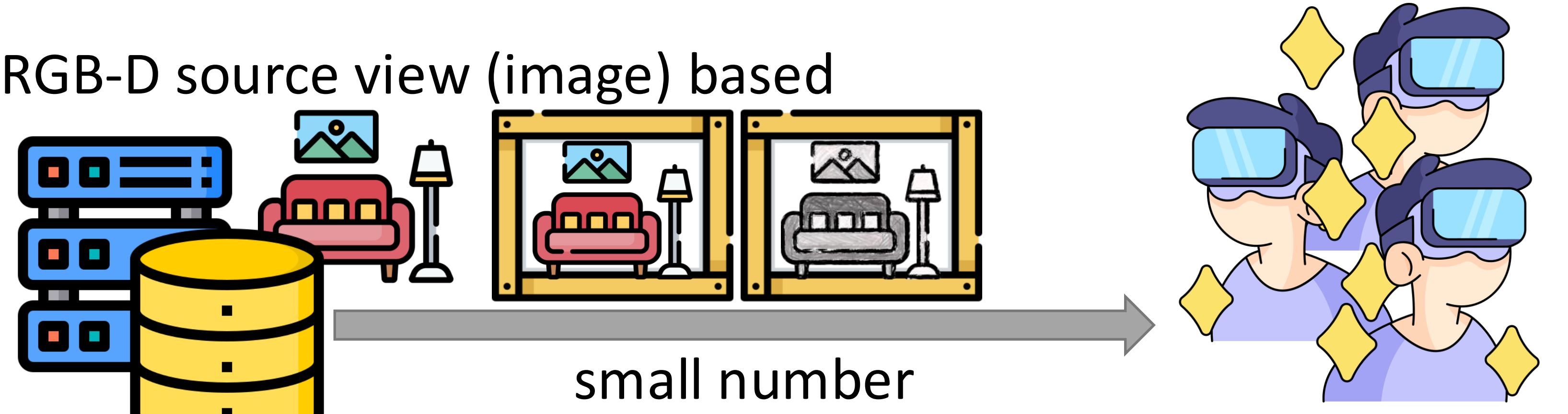
Inspiration

Bandwidth Saving and No Mesh Streaming

- Frame-by-Frame Streaming



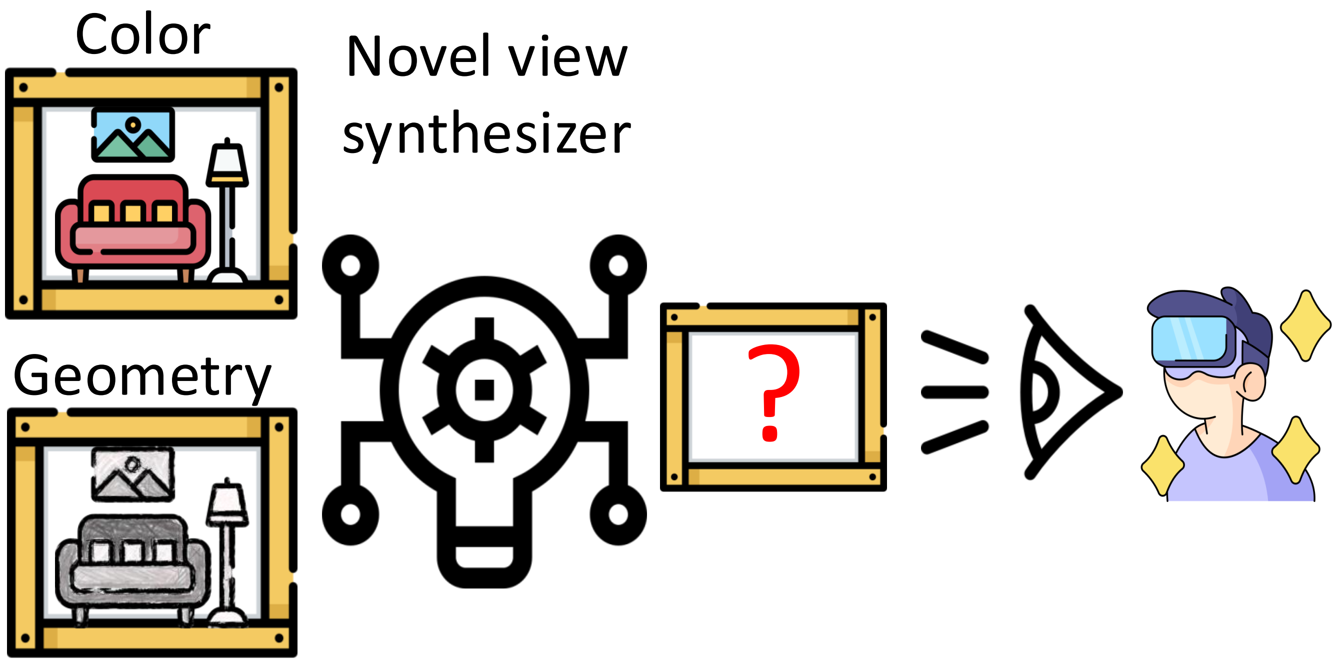
- RGB-D source view (image) based



Only sends a small number of source views to serve many clients

Novel View Synthesis

- RGB source views
  - Describe color information
- D source views
  - Describe partial content geometry

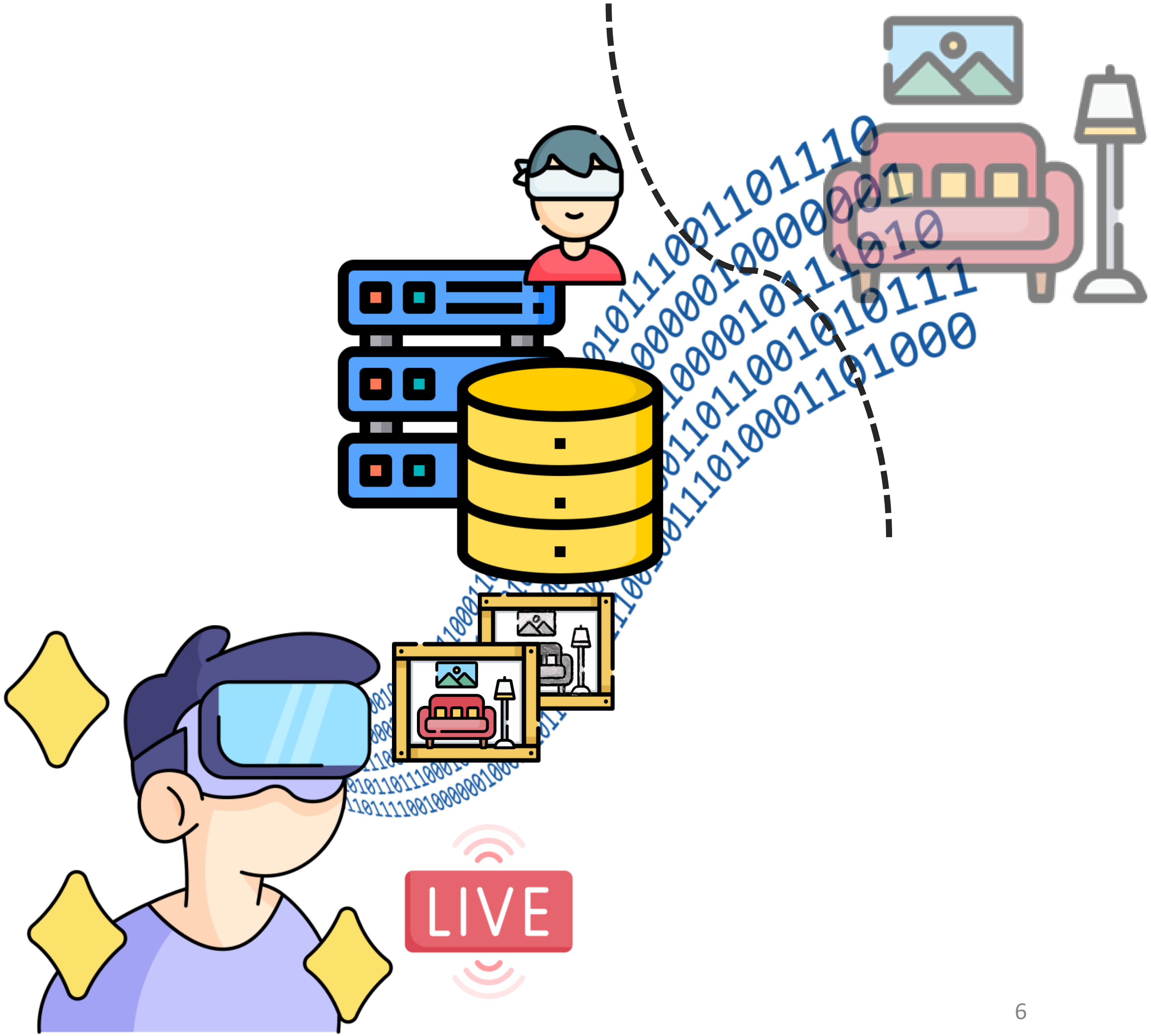


- Limitations
  - Not light enough to run on Head Mounted Displays (HMDs) in real-time
- Reference View Synthesizer (RVS)[RVS]

[RVS] Bart Kroon and Gauthier Lafruit. 2018. Reference View Synthesizer (RVS) 2.0 manual. Taipa, Macao

Outline

- Inspiration
- Goal & Challenges
- Related Work
- System Design
- Novel View Optimization
- Cloud Service Provider
  - Pose Predictor
  - Candidate Generator
  - Coverage Estimator
  - Solver & Algorithms
- Implementation
- Evaluations
- Conclusion & Future Work



Goal & Challenges

Goal:

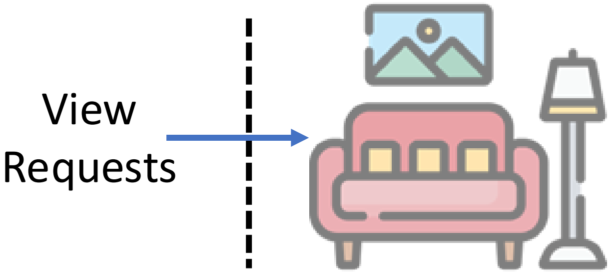
Synthesize high quality views for all clients

Constraints:

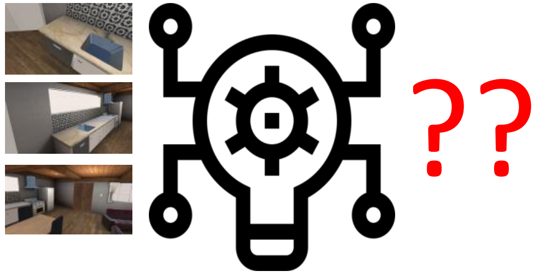
- Source view budgets (no. of source views allowed)
- Content observation budgets

Challenges:

No direct access to 3D content



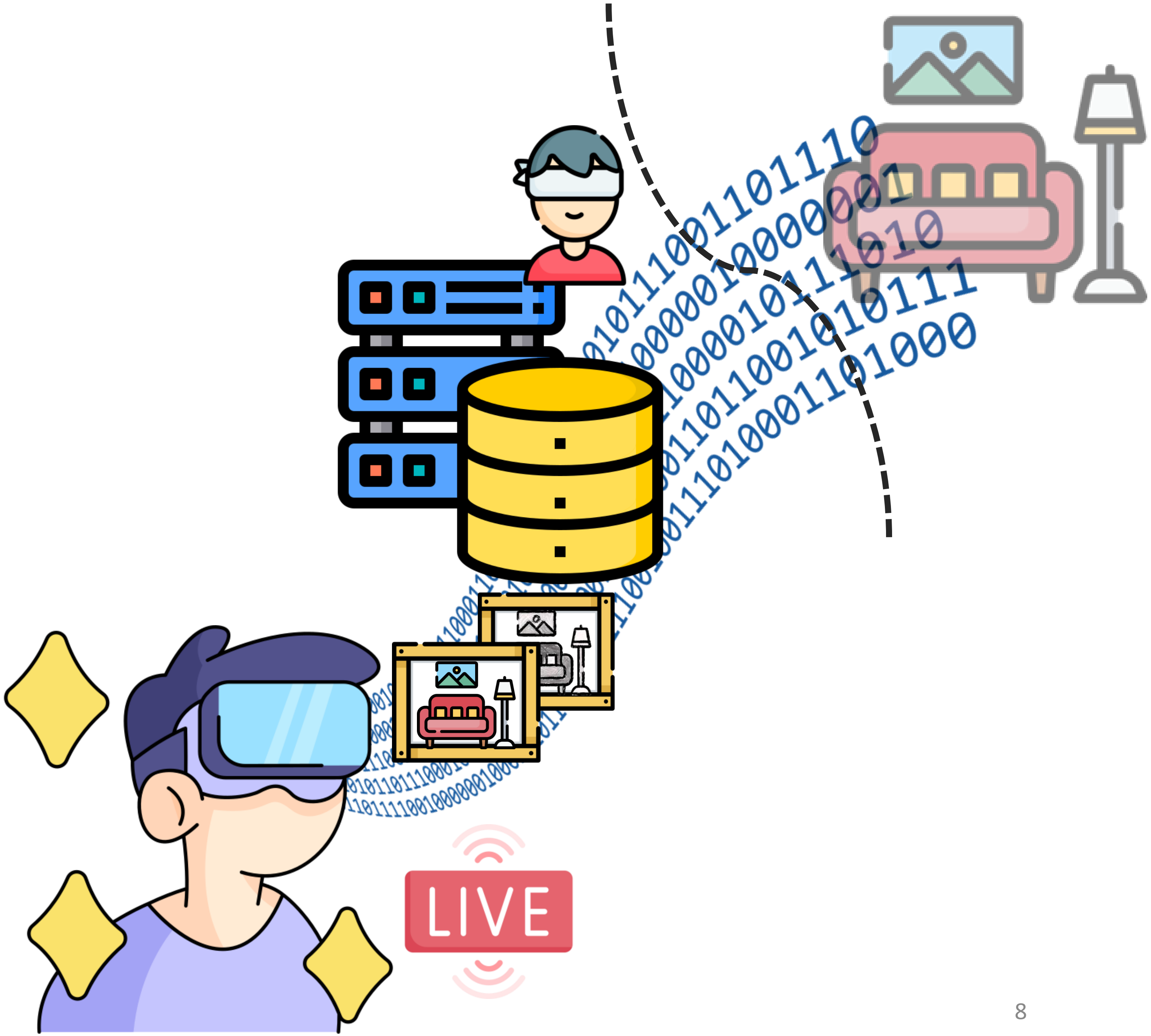
No close form representation of quality prediction



Defense against Structure-from-Motion

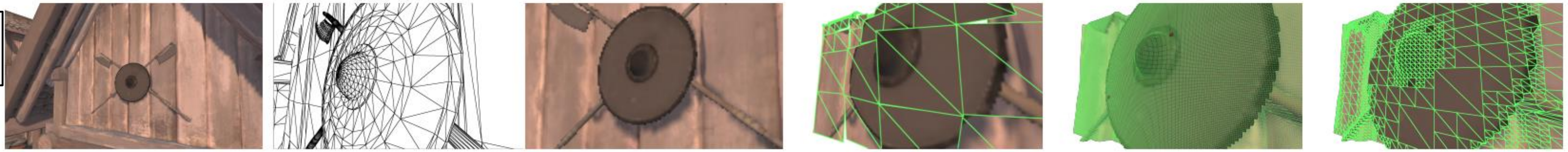
Outline

- Inspiration
- Goal & Challenges
- Related Work
- System Design
- Novel View Optimization
- Cloud Service Provider
  - Pose Predictor
  - Candidate Generator
  - Coverage Estimator
  - Solver & Algorithms
- Implementation
- Evaluations
- Conclusion & Future Work



Novel View Synthesis

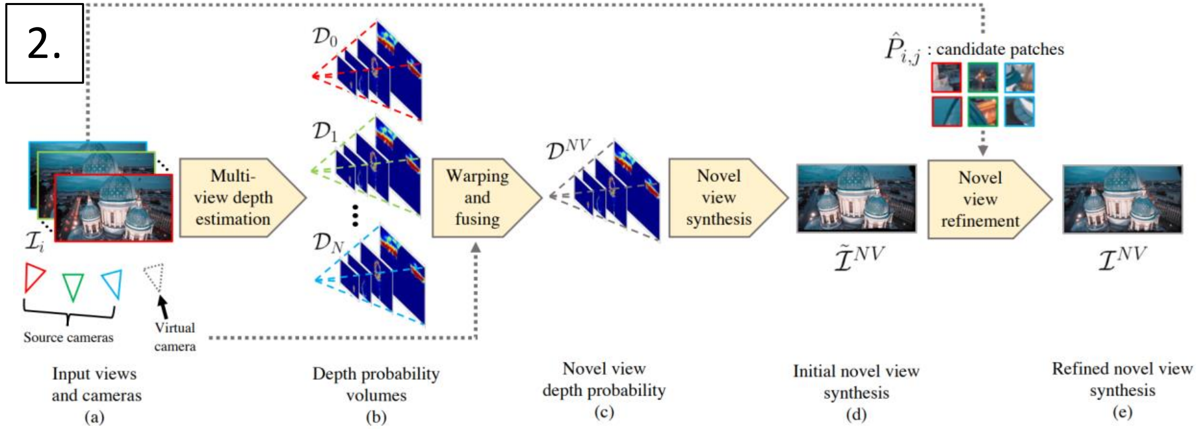
1.



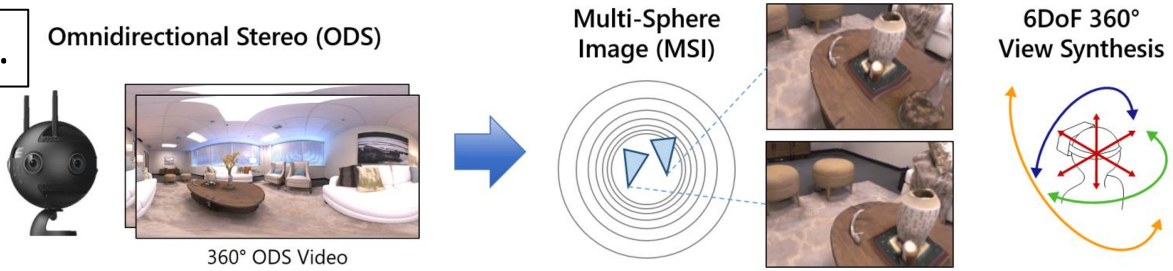
(a) Original scene (b) Scene geometry (c) Low-res client view (d) Proxy planes, initial (e) Tessellated (f) Merged and flattened

1. Hladky et al. invented QuadStream to synthesize view within a pre-defined view cell
   - Requires 3D content
2. Choi et al. generalized scalar depth prediction from multiple cameras to refine synthesized views
3. Attal et al. transform 360° stereo videos to multi-sphere images to synthesize 6-DoF views

2.



3.



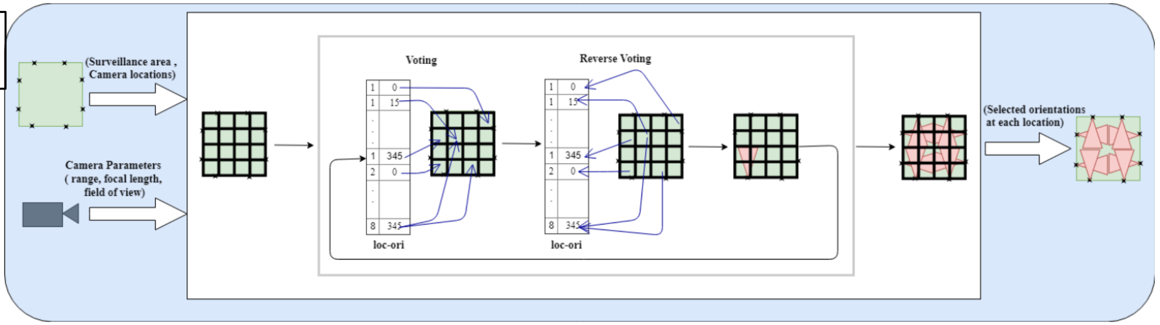
1. Jozef Hladky, Michael Stengel, Nicholas Vining, Bernhard Kerbl, Hans-Peter Seidel, and Markus Steinberger. 2022. QuadStream: A Quad-Based Scene Streaming Architecture for Novel Viewpoint Reconstruction. ACM Transactions on Graphics 41, 6 (November 2022), 1–13.
2. Inchang Choi, Orazio Gallo, Alejandro Troccoli, Min Kim, and Jan Kautz. 2019. Extreme View Synthesis. In Proc. of IEEE/CVF International Conference on Computer Vision (ICCV'19). Seoul, Korea.
3. Benjamin Attal, Selena Ling, Aaron Gokaslan, Christian Richardt, and James Tompkin. 2020. MatryODShka: Real-time 6DoF video view synthesis using multi-sphere images. In Proceedings of European Conference on Computer Vision (ECCV'20). Glasgow, United Kingdom, 441–459

Related Work

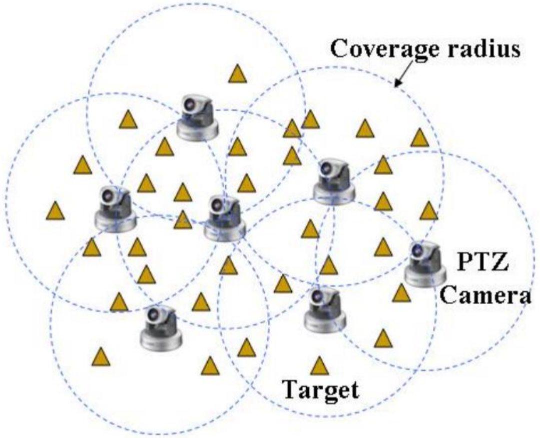
Coverage optimization

1. Suresh et al. solved 2D terrain coverage problem
   - Greedy based, discretized pose
2. Abu-Ghazaleh maximized the number of covered targets given a fixed number of cameras
3. Peng and Isler computed optimal flying paths for aerial 3D reconstruction

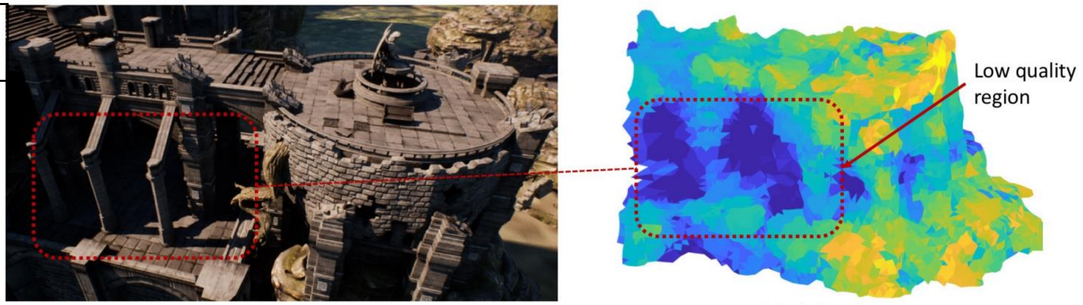
1.



2.



3.



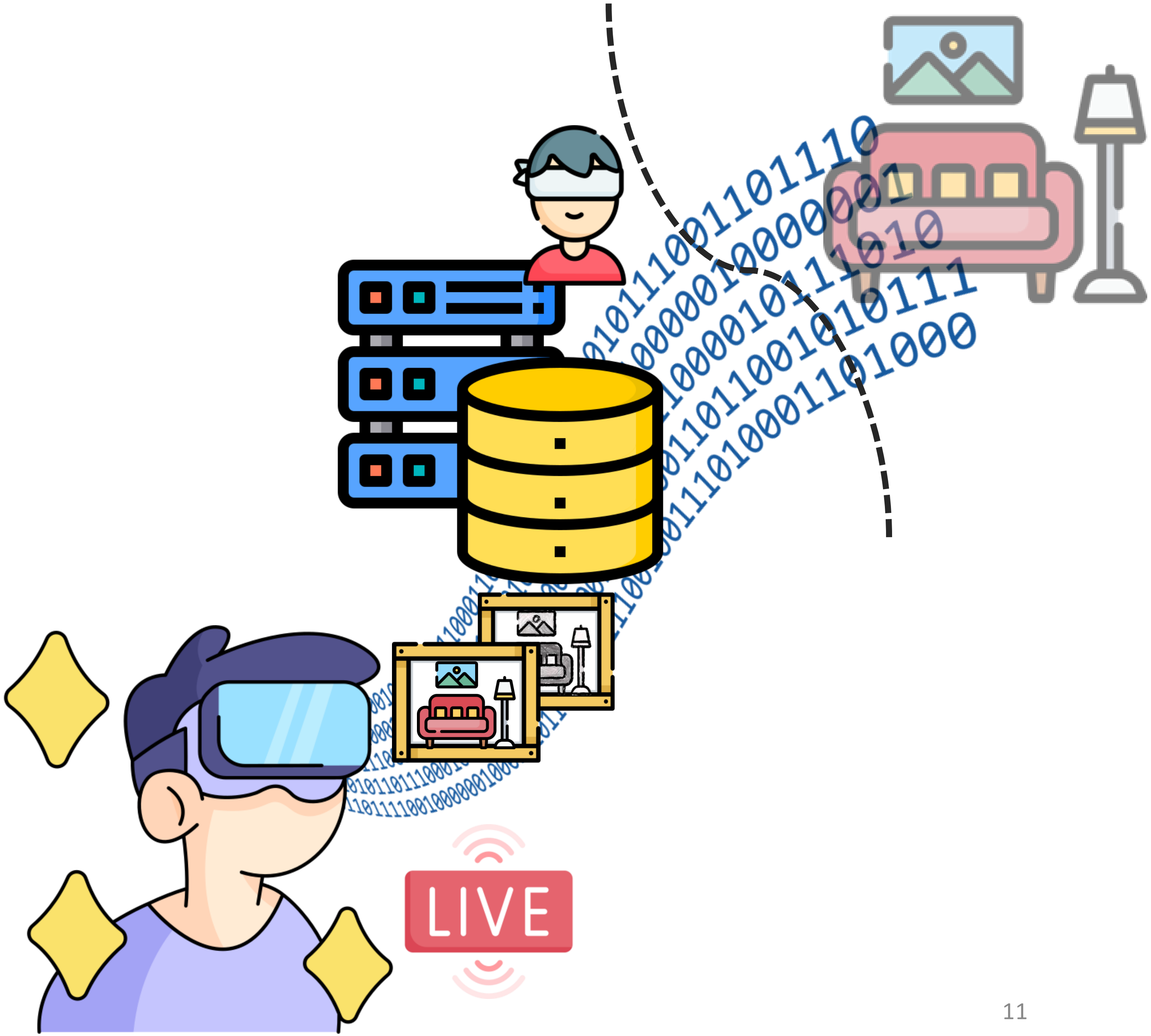
Ground Truth (scene 3)

Mesh Quality

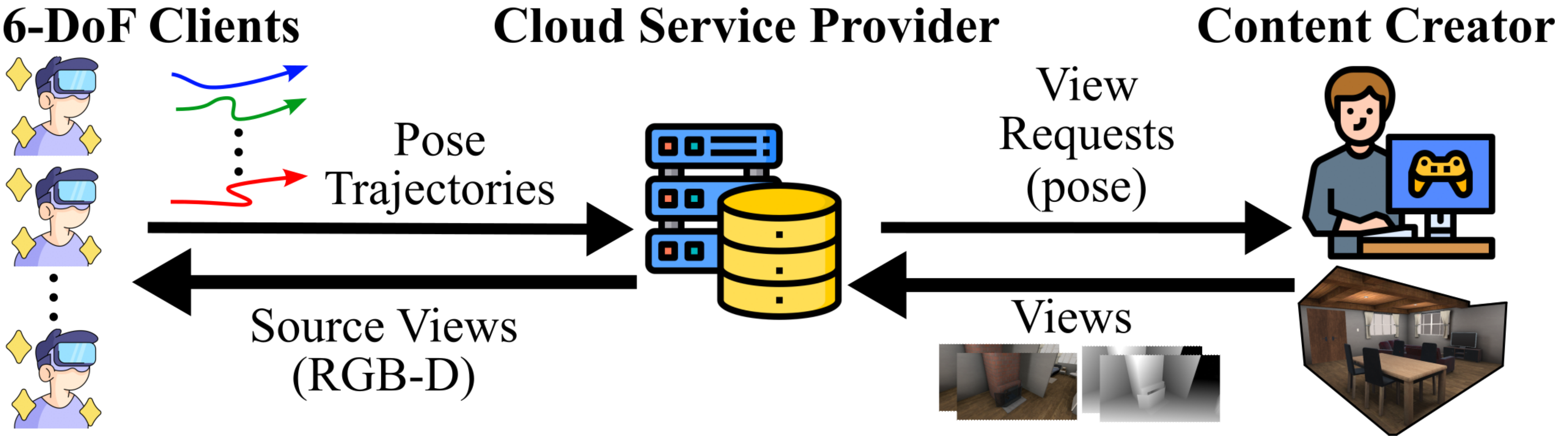
1. Sumi Suresh, Athi Narayanan, and Vivek Menon. 2020. Maximizing Camera Coverage in Multicamera Surveillance Networks. IEEE Sensors Journal 20, 17 (September 2020), 10170–10178
2. Vikram Munishwar and Nael Abu-Ghazaleh. 2010. Scalable Target Coverage in Smart Camera Networks. In Proc. of ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC'10). Atlanta, GA, 206–213.
3. heng Peng and Volkanr Isler. 2019. Adaptive View Planning for Aerial 3D Reconstruction. In Proc. of IEEE International Conference on Robotics and Automation (ICRA'19). Montreal, Canada, 2981–2987.

Outline

- Inspiration
- Goal & Challenges
- Related Work
- System Design
- Novel View Optimization
- Cloud Service Provider
  - Pose Predictor
  - Candidate Generator
  - Coverage Estimator
  - Solver & Algorithms
- Implementation
- Evaluations
- Conclusion & Future Work



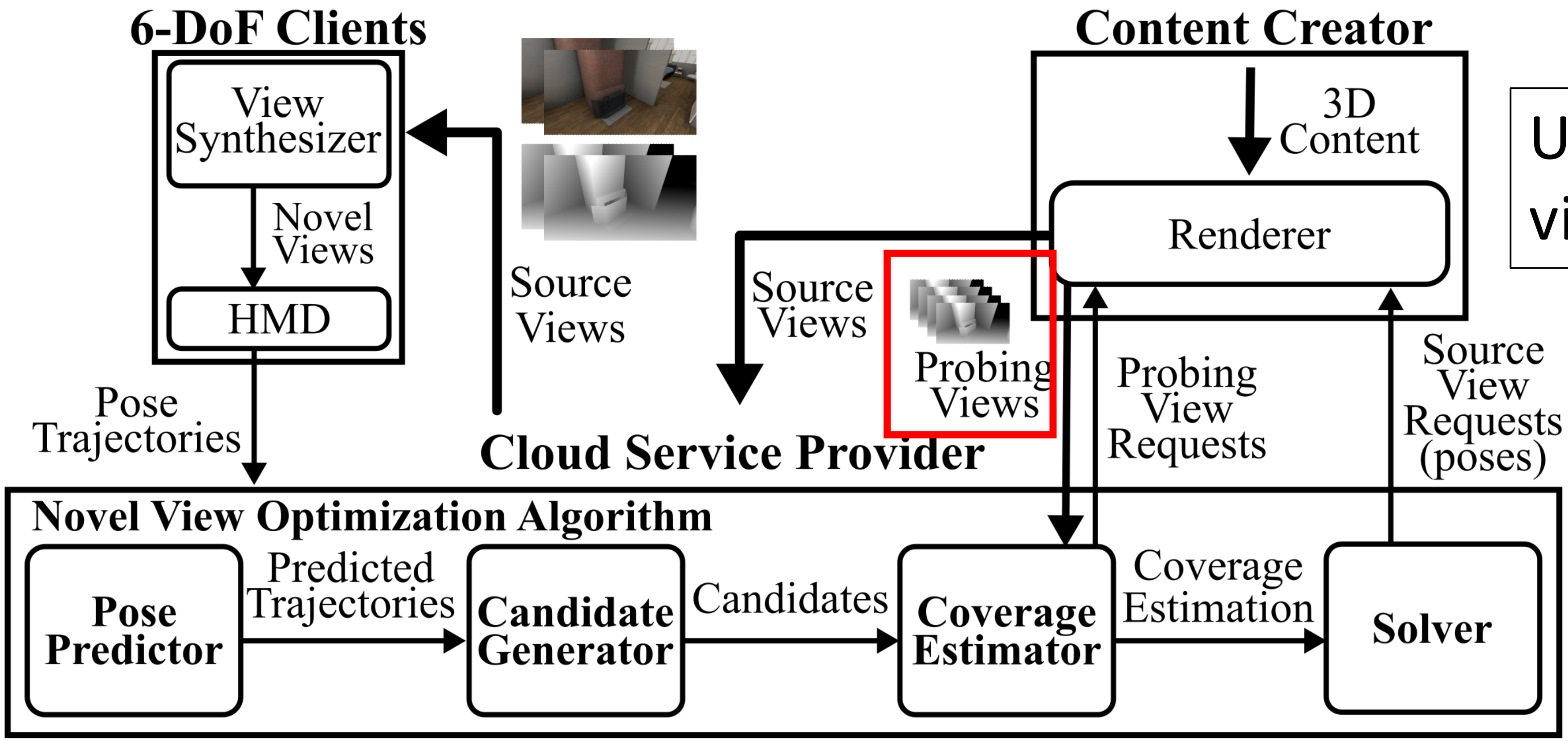
Blind Streaming



- 6-DoF clients
  - Transmit pose trajectories
    - Pairs of position & orientation (p, q)
  - Novel view synthesis
- Cloud service provider
  - Collect pose trajectories
  - On behalf of 6-DoF clients
  - Novel view optimization algorithms
- Content creator
  - Serve view requests

Component Diagram of Each Party

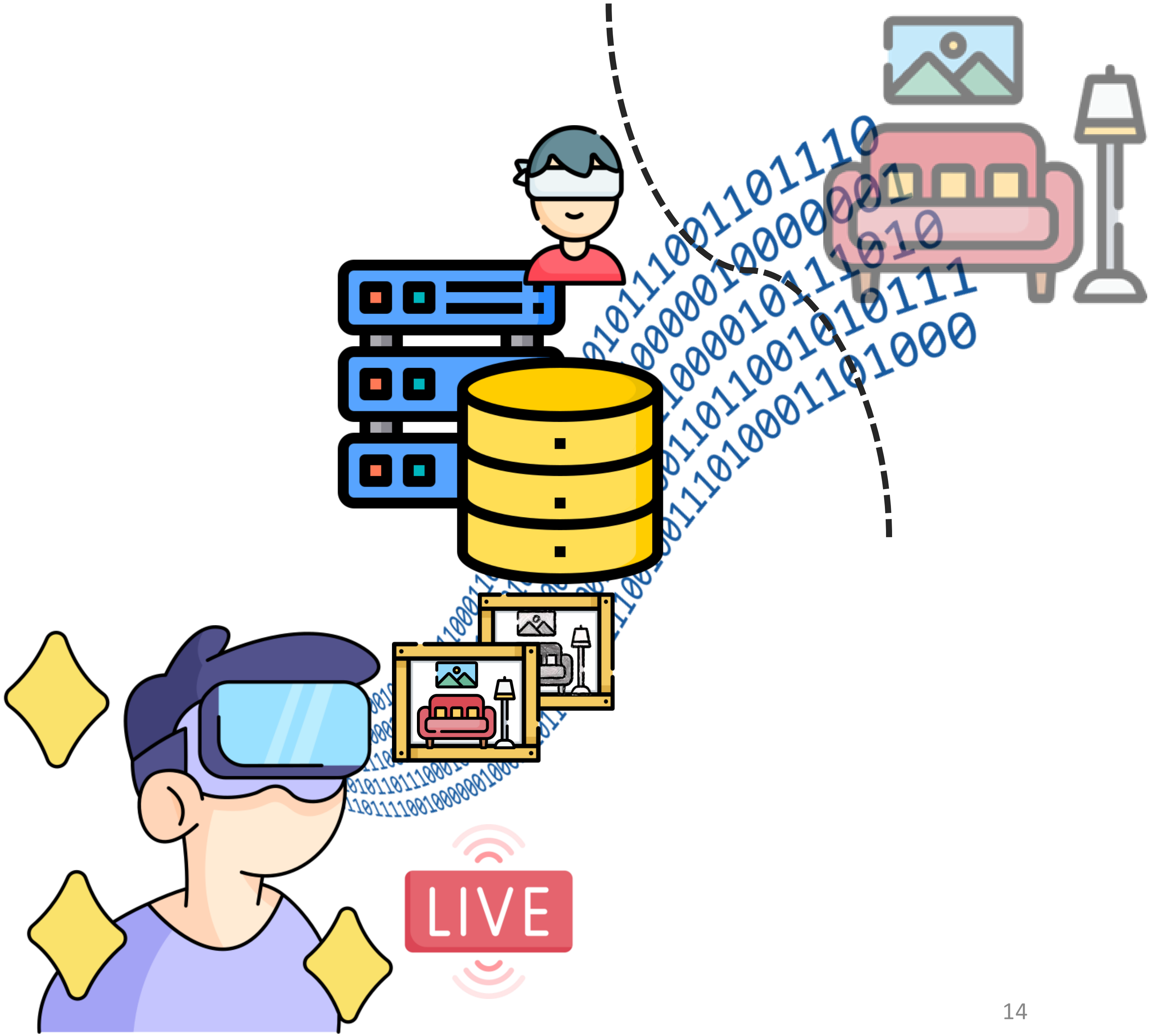
- Probing view: Low resolution depth image (1/16 of original)



Update source views every second

Outline

- Inspiration
- Goal & Challenges
- Related Work
- System Design
- Novel View Optimization
- Cloud Service Provider
    - Pose Predictor
    - Candidate Generator
    - Coverage Estimator
    - Solver & Algorithms
- Implementation
- Evaluations
- Conclusion & Future Work



Summation of Expected Quality over Novel Views and 6-DoF clients

expected quality of a novel view *v*

$$\underset{\boxed{\mathcal{S}_{\mathcal{T}}}}{\text{maximize}} \quad \underbrace{\sum\sum}_{\boxed{u\in\mathcal{U}\ \ t\in\mathcal{T}}} \boxed{\text{ql}(v_{u,t}, \mathcal{S}_{\mathcal{T}}, \mathcal{P}_{\mathcal{T}})}$$

choose the optimal set of source views

for all clients and all novel views

$$\text{subject to :} \quad \boxed{|\mathcal{S}_{\mathcal{T}}| \leq N};$$

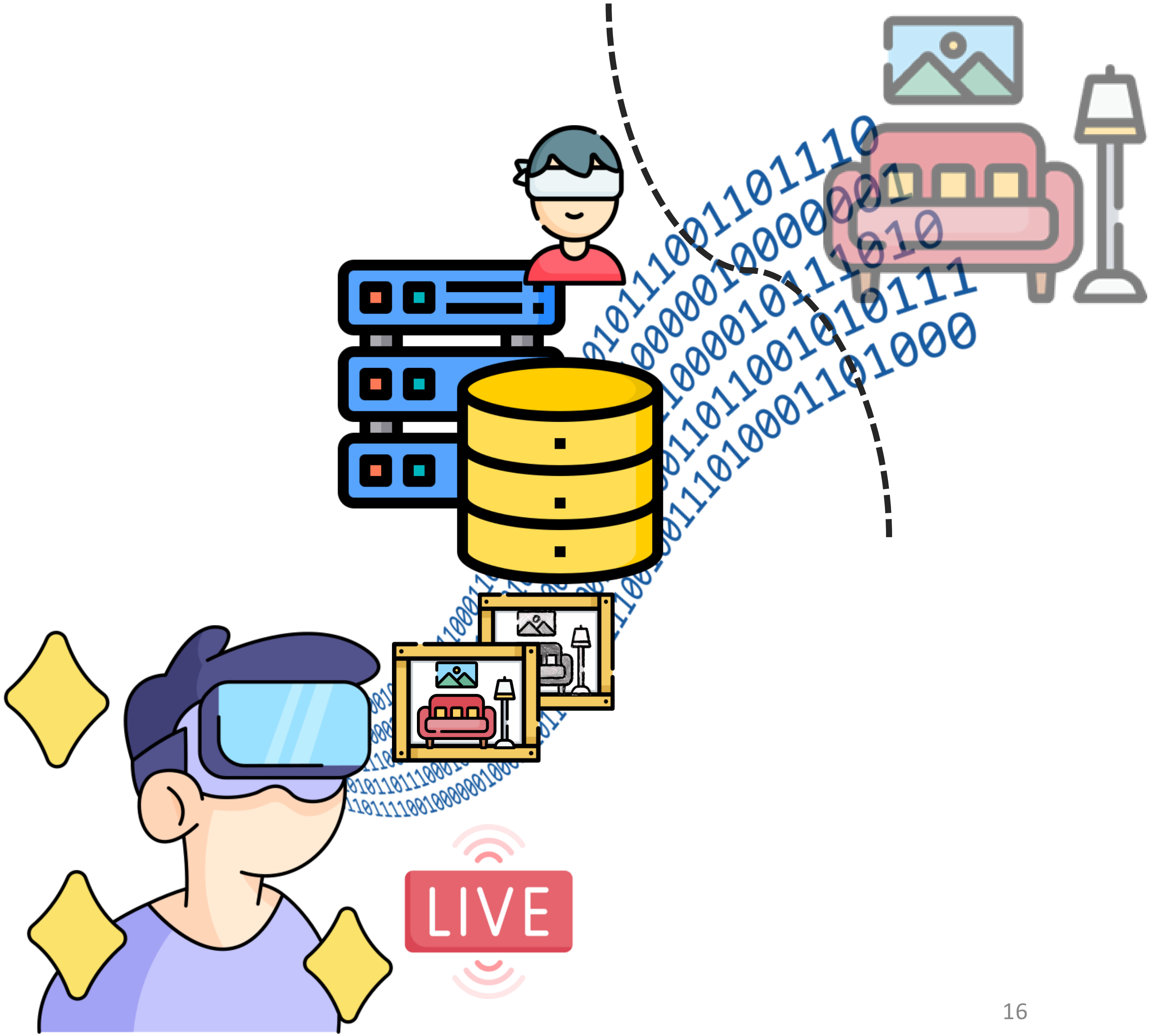
source view budgets

$$\boxed{|\mathcal{P}_{\mathcal{T}}| \leq M},$$

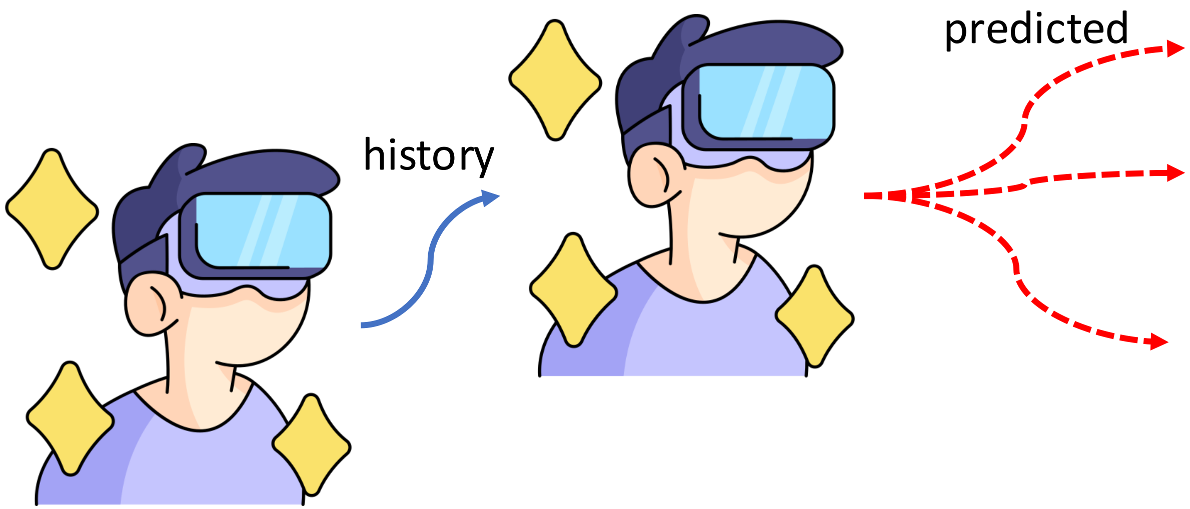
probing view budgets

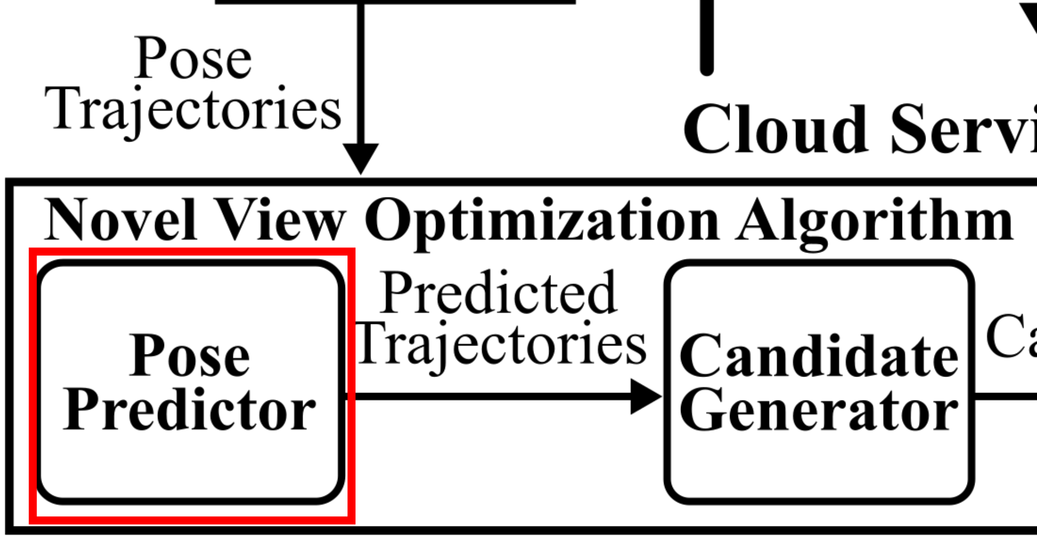
Outline

- Inspiration
- Goal & Challenges
- Related Work
- System Design
- Novel View Optimization
- Cloud Service Provider
  - Pose Predictor
  - Candidate Generator
  - Coverage Estimator
  - Solver & Algorithms
- Implementation
- Evaluations
- Conclusion & Future Work



Compensate for Latency





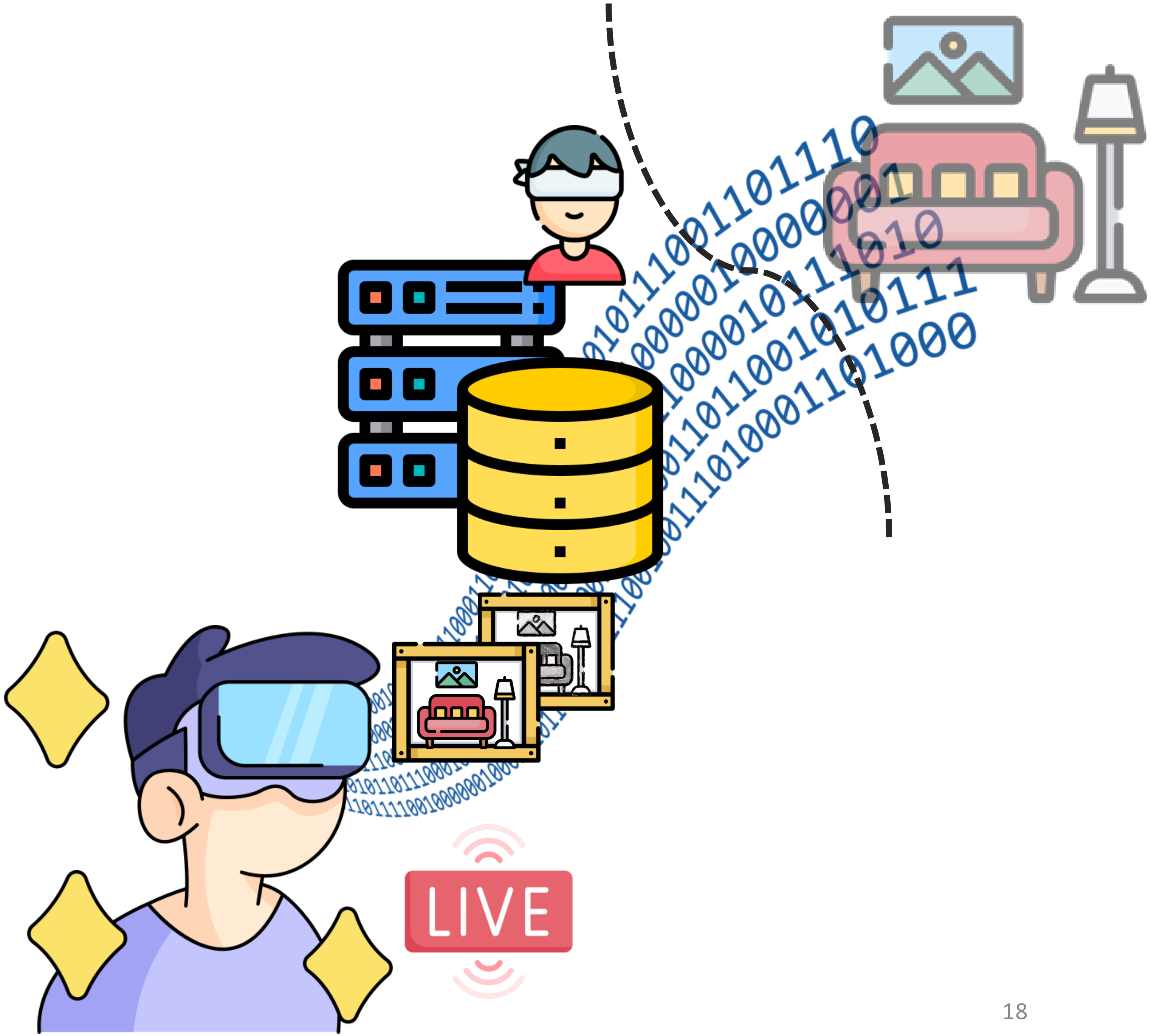
- Take history pose trajectory to predict future ones
- Compute source views beforehand

- Mature work
  - Kalman filter based (Serhan et al.)
  - LSTM based (Hou et al.)
- Assume perfect prediction

S. Gul, S. Bosse, D. Podborski, T. Schierl, and C. Hellge. Kalman filter-based head motion prediction for cloud-based mixed reality. In Proc. of ACM International Conference on Multimedia (MM'20) page 3632–3641, Seattle, WA, October 2020
X. Hou and S. Dey. Motion prediction and pre-rendering at the edge to enable ultralow latency mobile 6dof experiences. IEEE Open Journal of the Communications Society, 1:1674–1690, 2020

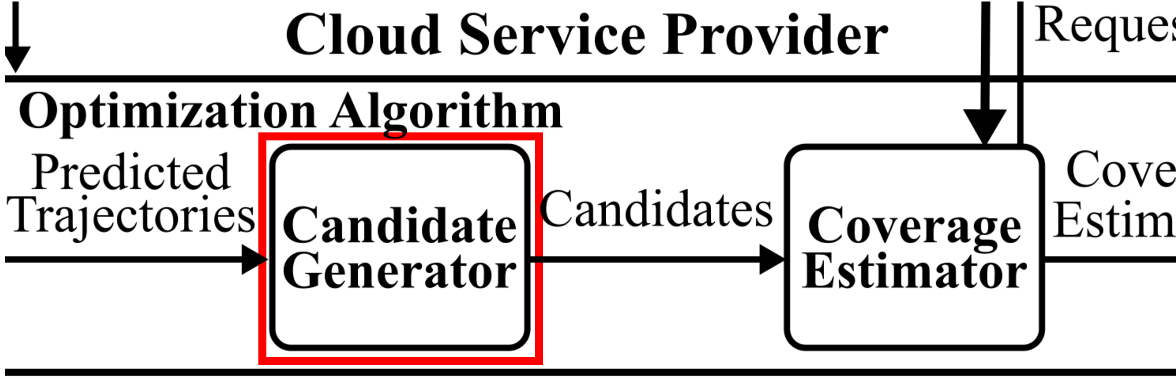
Outline

- Inspiration
- Goal & Challenges
- Related Work
- System Design
- Novel View Optimization
- Cloud Service Provider
  - Pose Predictor
  - Candidate Generator
  - Coverage Estimator
  - Solver & Algorithms
- Implementation
- Evaluations
- Conclusion & Future Work



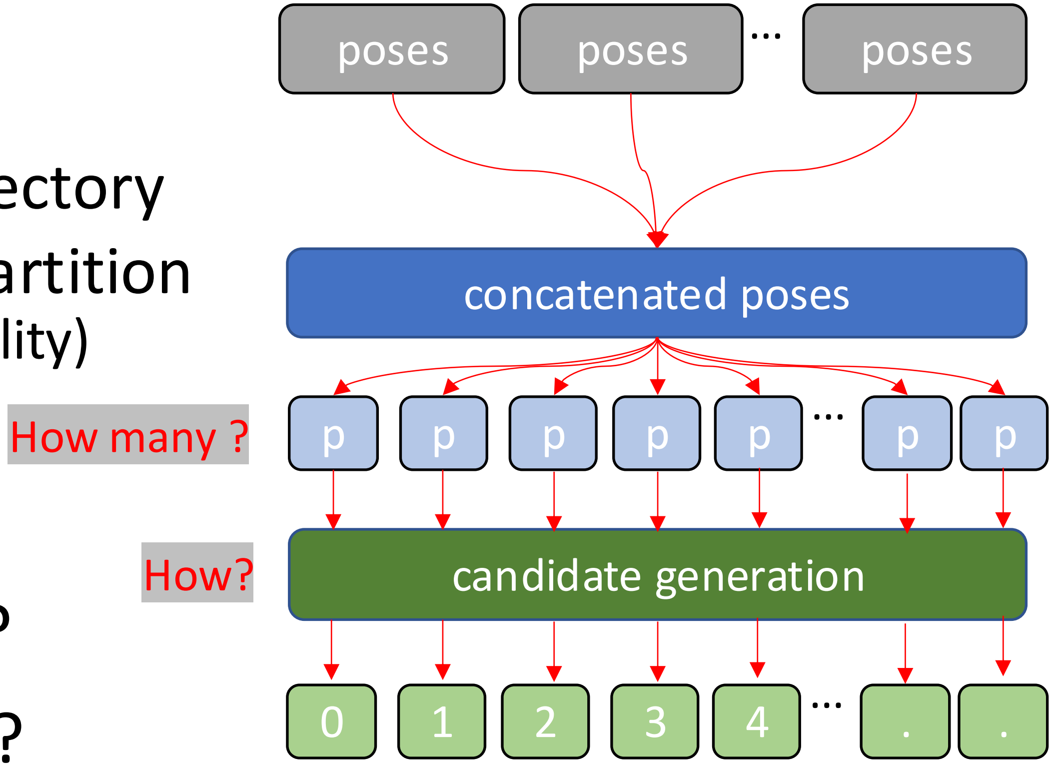
Candidate Generator

Tradeoff between Runtime and Optimality



Procedure:

1. Concatenate all poses from all 6-DoF clients
2. Cut partitions from the concatenated pose trajectory
3. Generate a source view candidate from each partition
   - Candidates as representatives (leverage temporal locality)
   - The pose at least covers nearby poses

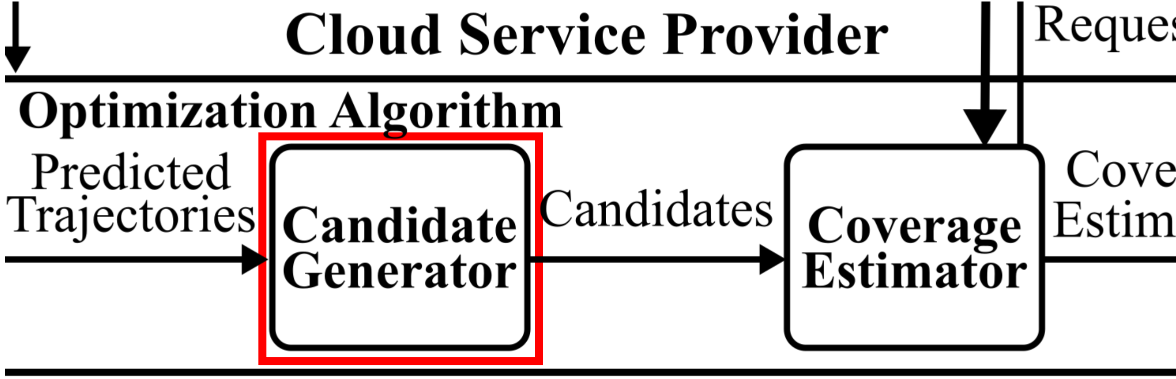


To be determined:

- How many partitions (poses) should we have?
- How to generate a candidate from a partition?

19

Strike Optimal Number of Partitions

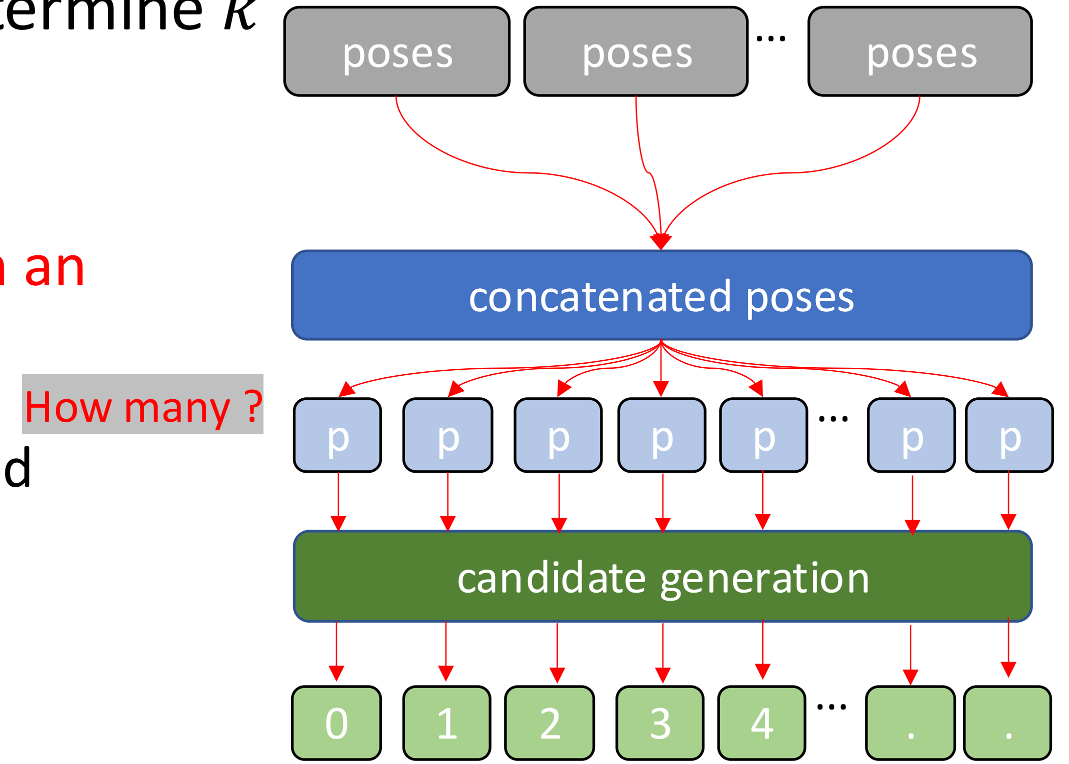


Random arbitrary-number selection analysis to determine k

(Relaxed to selection of $\geq N$ source views)

- N, M are no. of source views and partitions
- k = M/N, redundant factor
- $m = N/P$, source view budget
- $l = rm$, computational load
- $h = \frac{M}{P} + rm$, candidate overhead + computational load

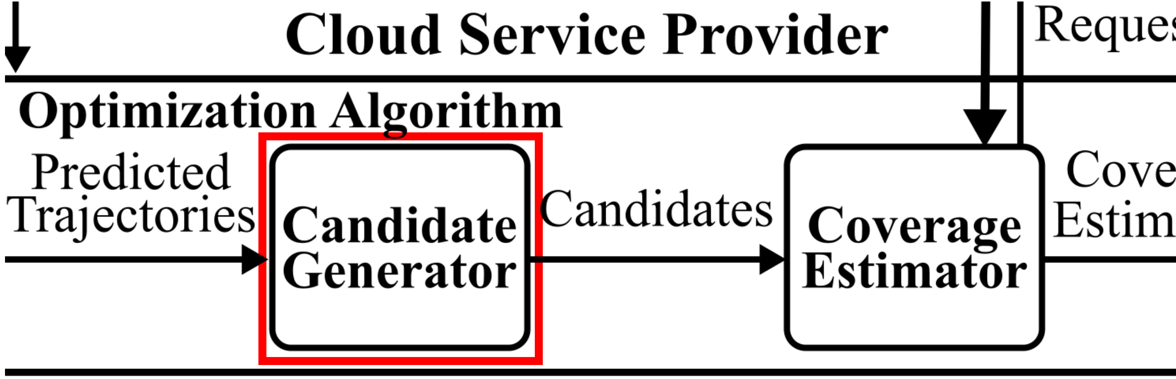
m, h are constant in an experiment



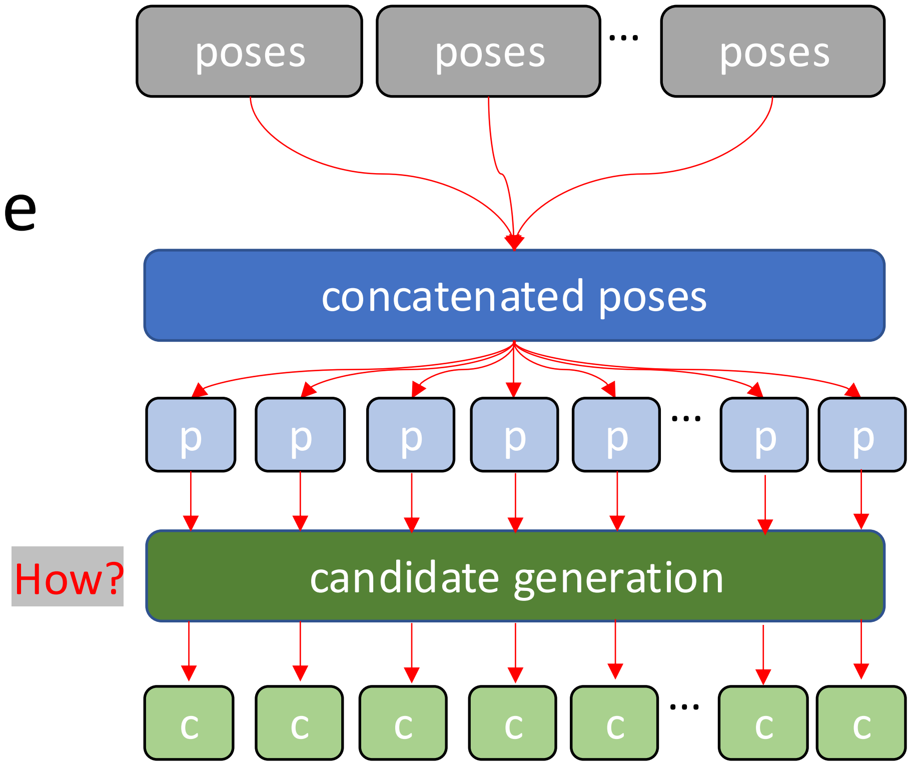
1. Select M candidates out of P poses at random
2. Select $\geq N$ source views from M candidates
3. $k = \sqrt{\frac{(m+h)(mP-1)}{m^2P}} \approx \frac{\sqrt{m+h}}{\sqrt{m}}$ as $P \to \infty$

Candidate Generator

Generate a Candidate from a Partition



- Consider position and orientation separately
- Average pose of a partition of size L as a candidate
- $\bar{p}$: average position = $(\bar{x}, \bar{y}, \bar{z})$
  - Vector mean
- $\bar{q}$: average orientation = $\overline{q_w} + \overline{q_x}\hat{\imath} + \overline{q_y}\hat{\jmath} + \overline{q_z}\hat{k}$
  - Unit quaternion to avoid rotation order ambiguity
  - Solve a maximum eigenvalue problem of a 4x4 matrix

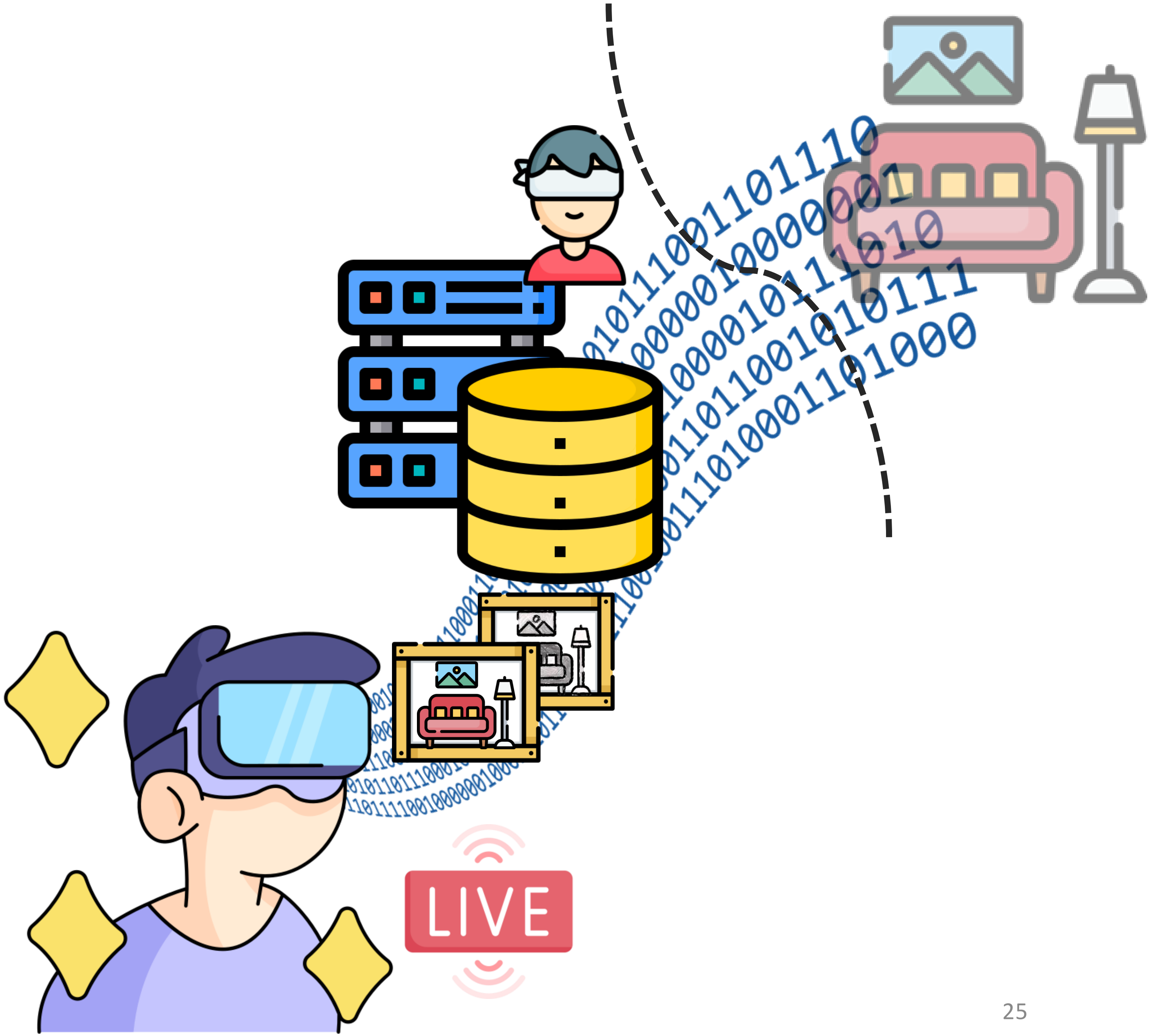


$$\text{cdd} = (\overline{p}, \overline{q}) = \left(\frac{1}{L}\sum_{i=1}^{L} p_i, \operatorname*{argmax}_{q \in \mathbb{S}^3}\{q^T(\sum_{i=1}^{L} q_i q_i^T)q\}\right)$$

L is the length of a partition

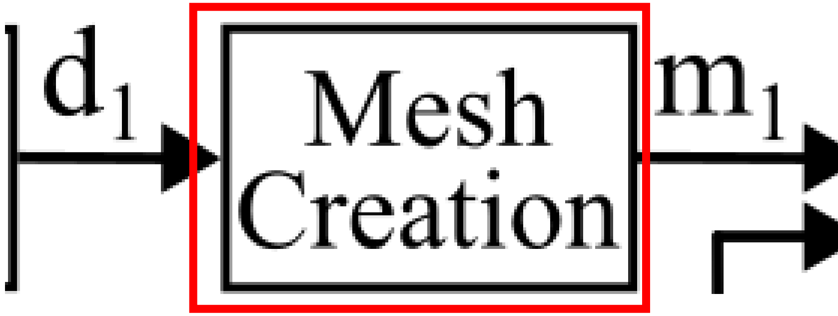
Outline

- Inspiration
- Goal & Challenges
- Related Work
- System Design
- Novel View Optimization
- Cloud Service Provider
  - Pose Predictor
  - Candidate Generator
  - Coverage Estimator
  - Solver & Algorithms
- Implementation
- Evaluations
- Conclusion & Future Work

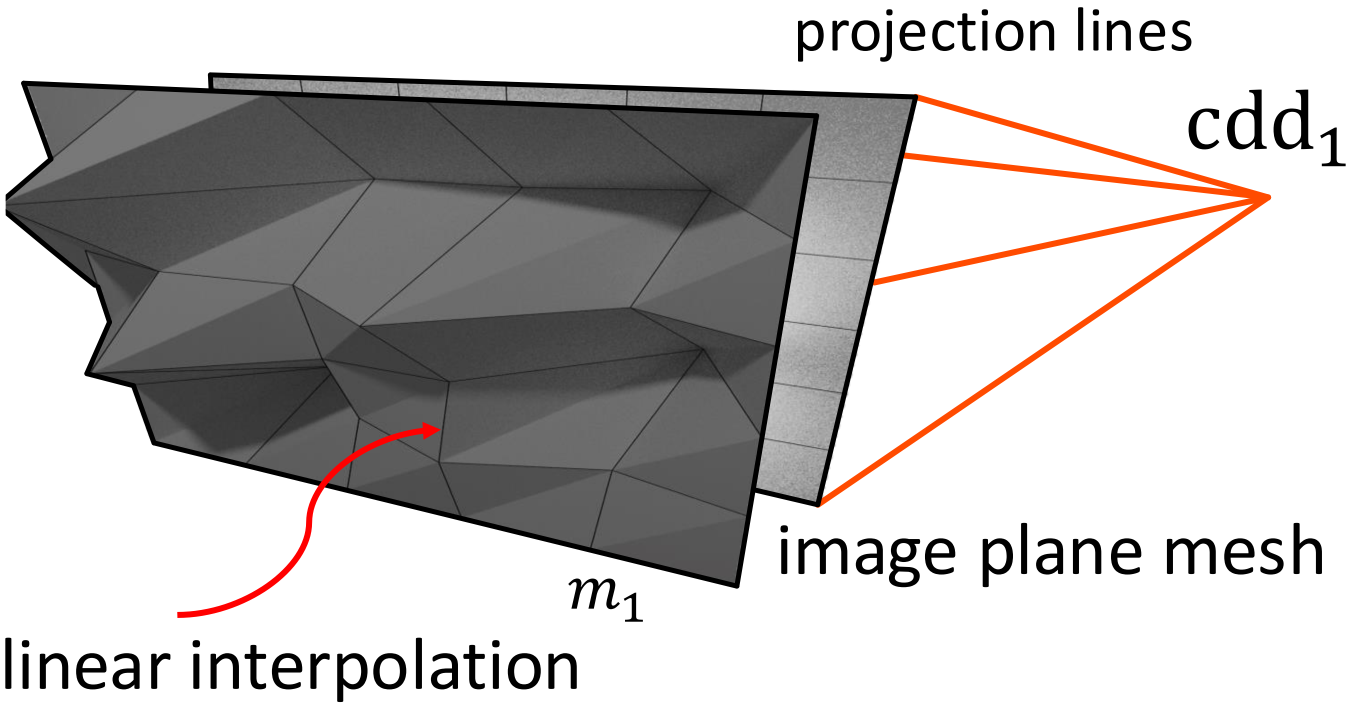


Coverage Estimator

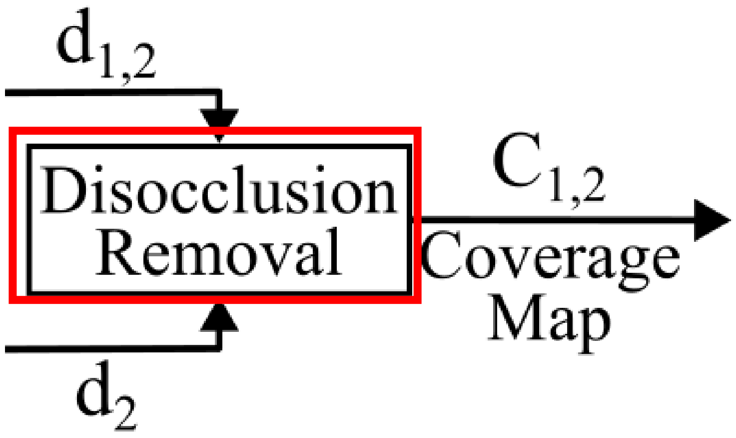
Mesh Creation



1. Create a image plane mesh of WxH vertices seen from cdd_1
2. Move the vertices along the projection lines according to their depth
3. Vertex connections are kept
   - Linear interpolation of depth between vertices
4. Transform the mesh to cdd_2



Disocclusion Removal



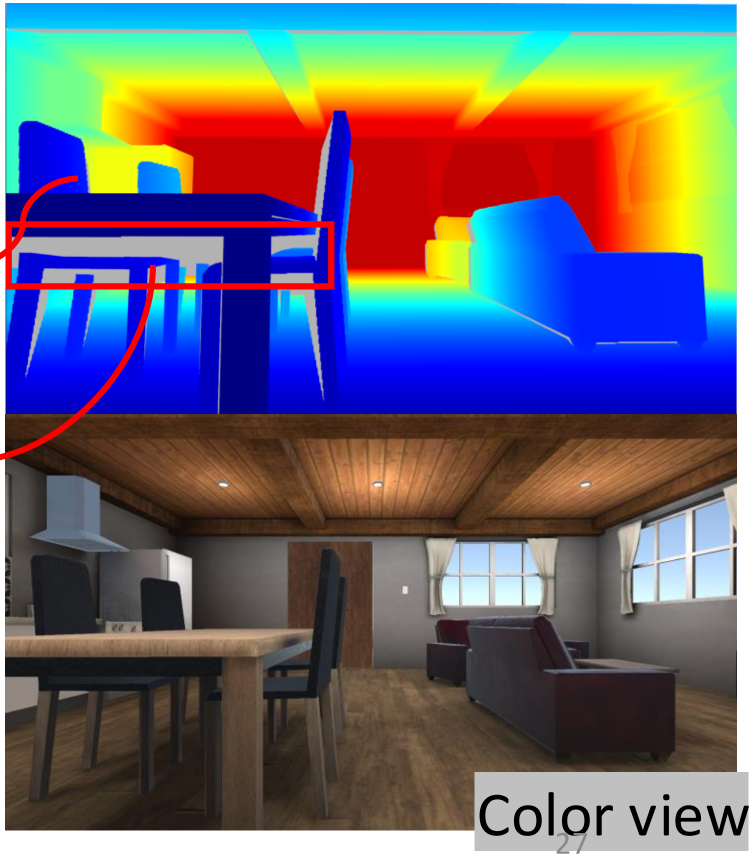
Analyze how cdd_1 covers cdd_2

Values in $d_{1,2}$ should be consistent with d_2 unless:

1. cdd_1 does not cover that pixel →Infinity depth
2. That part is disoccluded

Colored parts from cdd_1

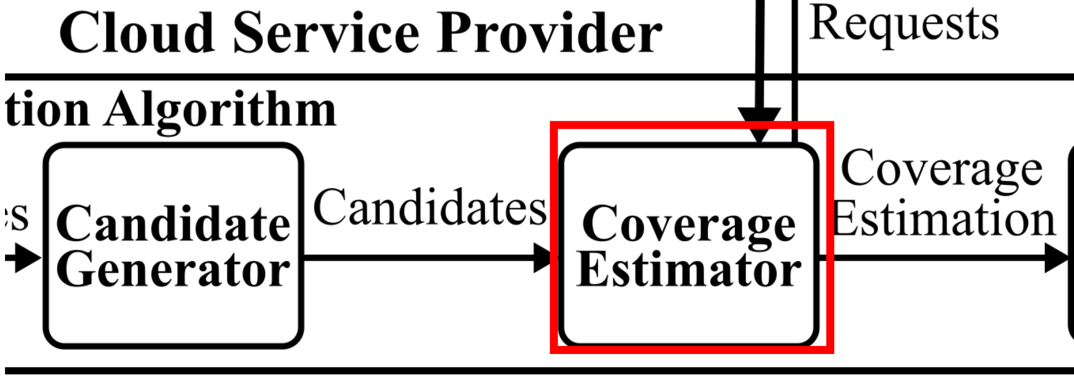
Gray parts are disocclusion



Disocclusion removal:

1. Compute $\mathrm{d}_{\mathrm{abs}} = |\, d_{1,2} - d_2|$
2. Remove those ≥ threshold in d_{abs}

Compute Coverage Map of cdd_1 on cdd_2, $C_{1,2}$

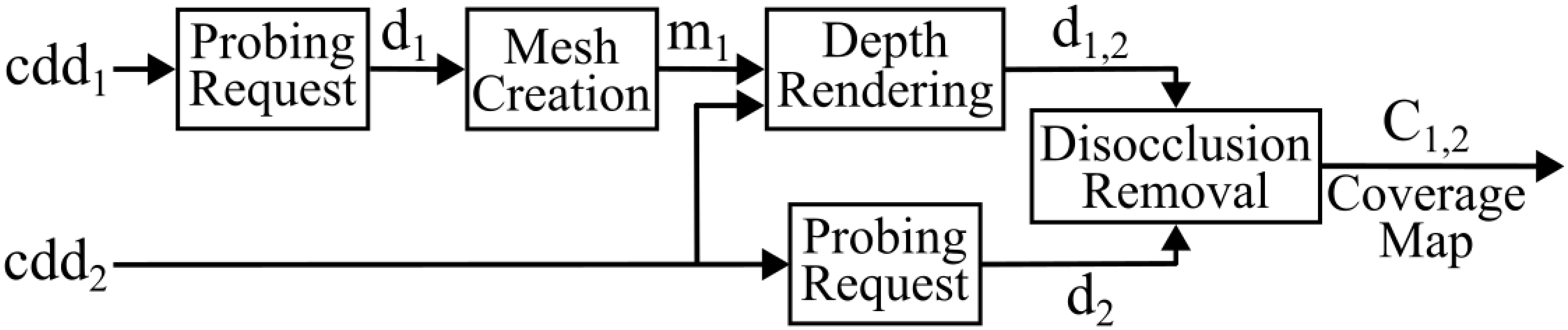


For a pair of candidates:

1. Request depth images, d_1 and d_2
2. Create mesh m_1 from d_1
3. Re-project m_1 to cdd_2 as $d_{1,2}$
4. Remove disocculusion of $d_{1,2}$ by comparing with d_2

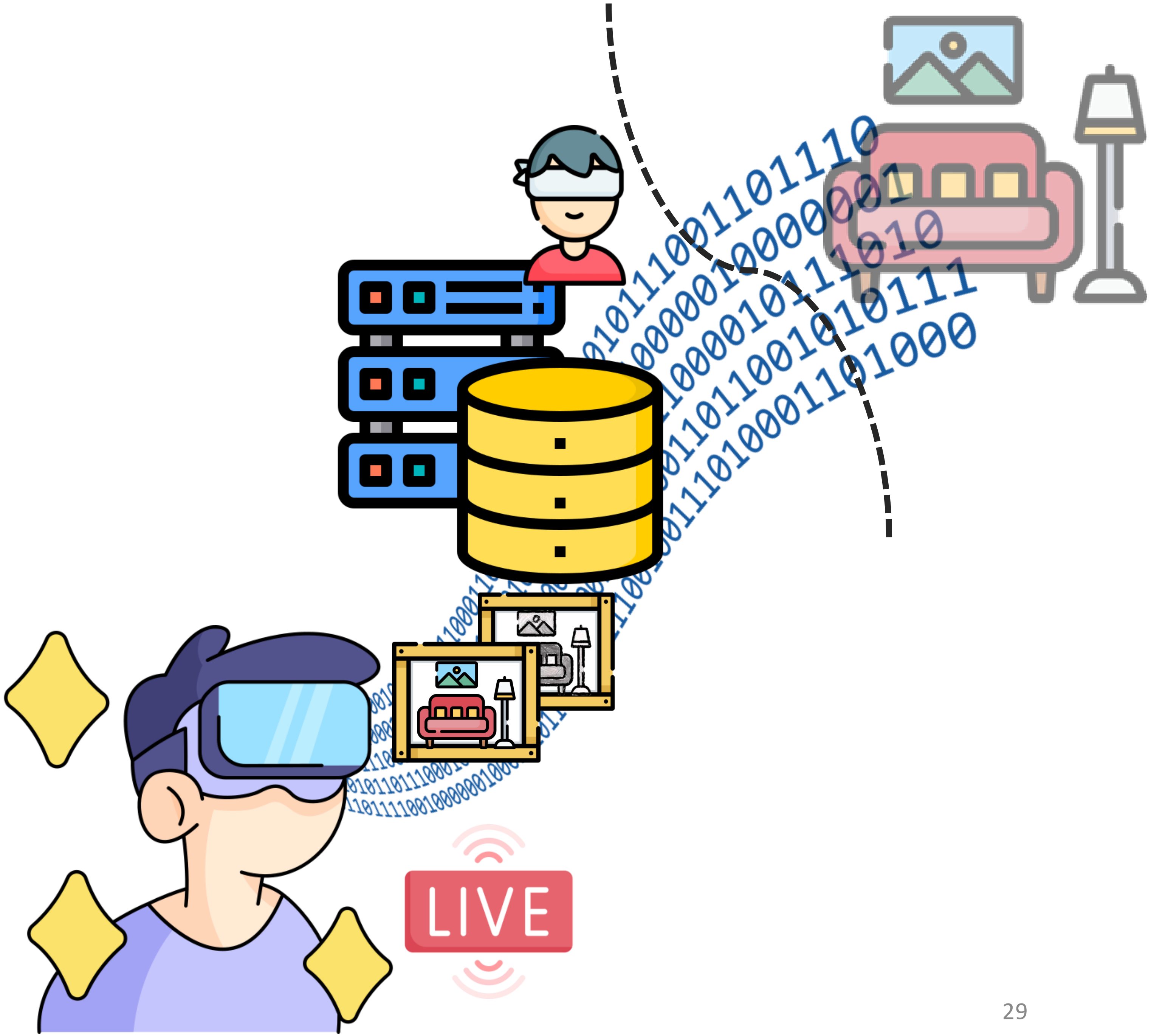
For all pairs of candidates:

1. Repeat the procedure of computing $C_{j,i}$ for all candidates
2. Result in M probing views (low resolution depth images)

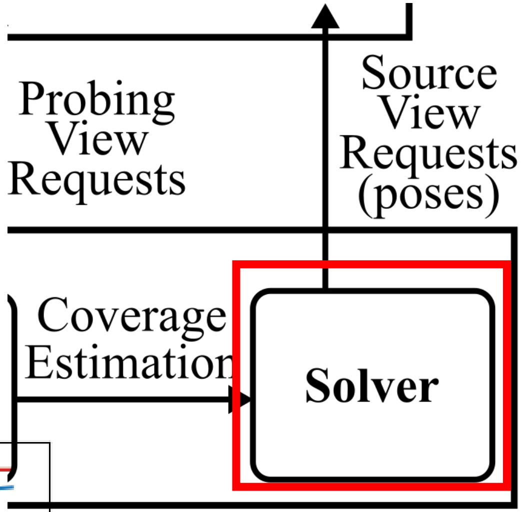


Outline

- Inspiration
- Goal & Challenges
- Related Work
- System Design
- Novel View Optimization
- Cloud Service Provider
  - Pose Predictor
  - Candidate Generator
  - Coverage Estimator
  - Solver & Algorithms
- Implementation
- Evaluations
- Conclusion & Future Work



Pixel Contribution Function $f(c)$



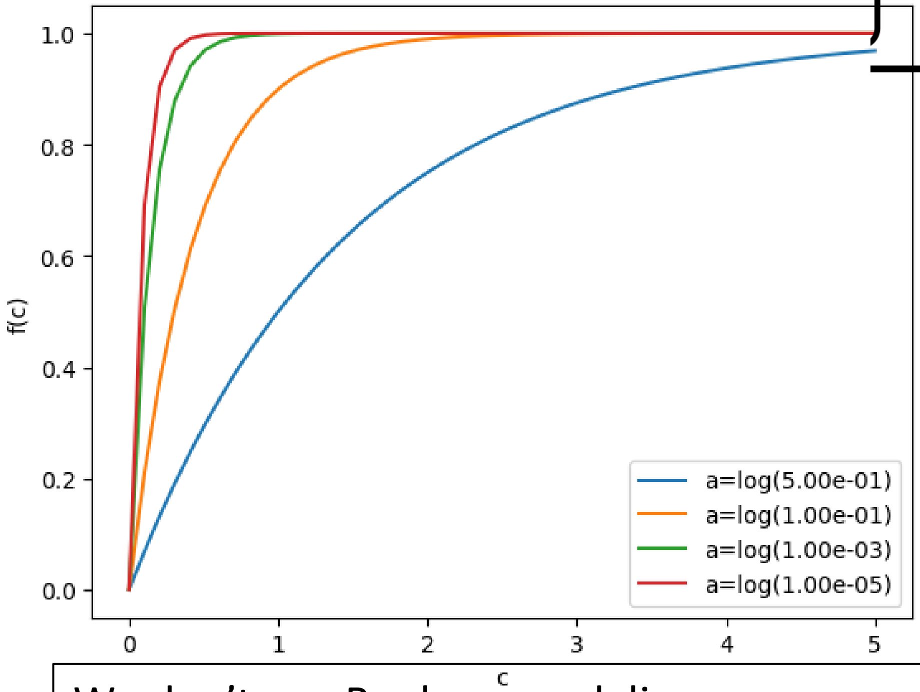
Describe contribution of a pixel:

$f(c) = 1 - e^{ac}$ for $a < 0, c \in Z$

- c : coverage count
- $a = \log(10^{-5})$

- Zero coverage: $f(c) = 0, c = 0$
- Bounded quality: $f(c) \to 1, c \to \infty$
- Monotonic increase: $f(c_1) \geq f(c_2)$ for $c_1 \geq c_2$
- Quality saturation: $f'(c_1) \leq f'(c_2)$ for $c_1 \geq c_2$

Saturates fast and bounded



We don't use Boolean modeling:

- $b(c) = 1$ for $c > 0$
- $b(c) = 0$ for $c \leq 0$

because we seek for improvement from multiple coverage

Solver

Optimization Objective

average over all candidates

$$\underset{\{s_j\}}{\text{maximize}} \quad g(\{s_j\}) = \mathcal{W} \boxed{\odot} \sum_{i}^{M} \boxed{1 - e^{a\left(\boxed{\sum_j^M s_j C_{j,i}}\right)}}$$

coverage count of $\mathrm{cdd_i}$

matrix version of $f(c)$

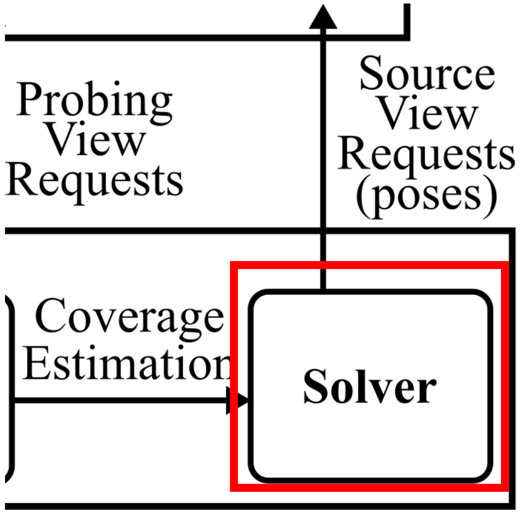
$$\text{subject to :} \quad \boxed{s_j = \{0,1\}} \text{ for } 1 \leq j \leq M$$

select or not

$$\boxed{\sum_j^M s_j = N;}$$

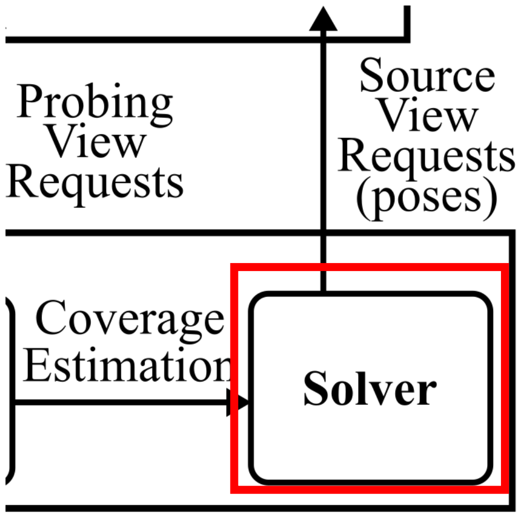
source view budget

We will call it ***g* value** in the following discussion



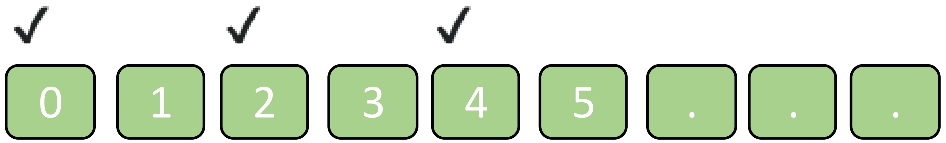
- $C_{j,i}$: coverage map of how $\mathrm{cdd_j}$ covers $\mathrm{cdd_i}$
- $\{s_j\}$: Boolean decision variables
  - $s_j = 1$ indicates the j^{th} candidate is selected
- W: weighting mask (averaging mask)
- $\odot$: element-wise multiplication and summation

Uniform Solver (Uni)



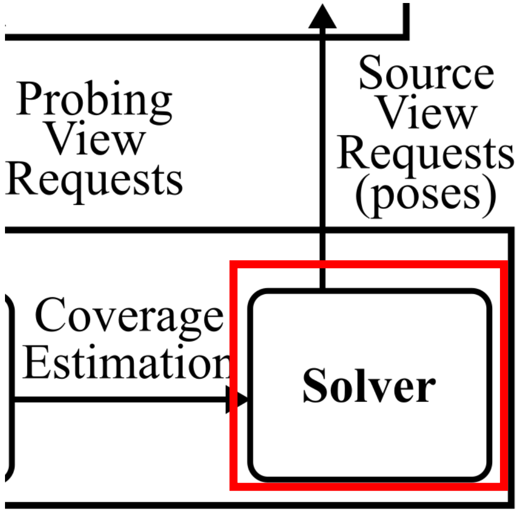
Pick candidates every fixed skips

- Guarantees uniform source view distribution across temporal axis and 6-DoF clients
- No need for coverage estimation
- Runs fast

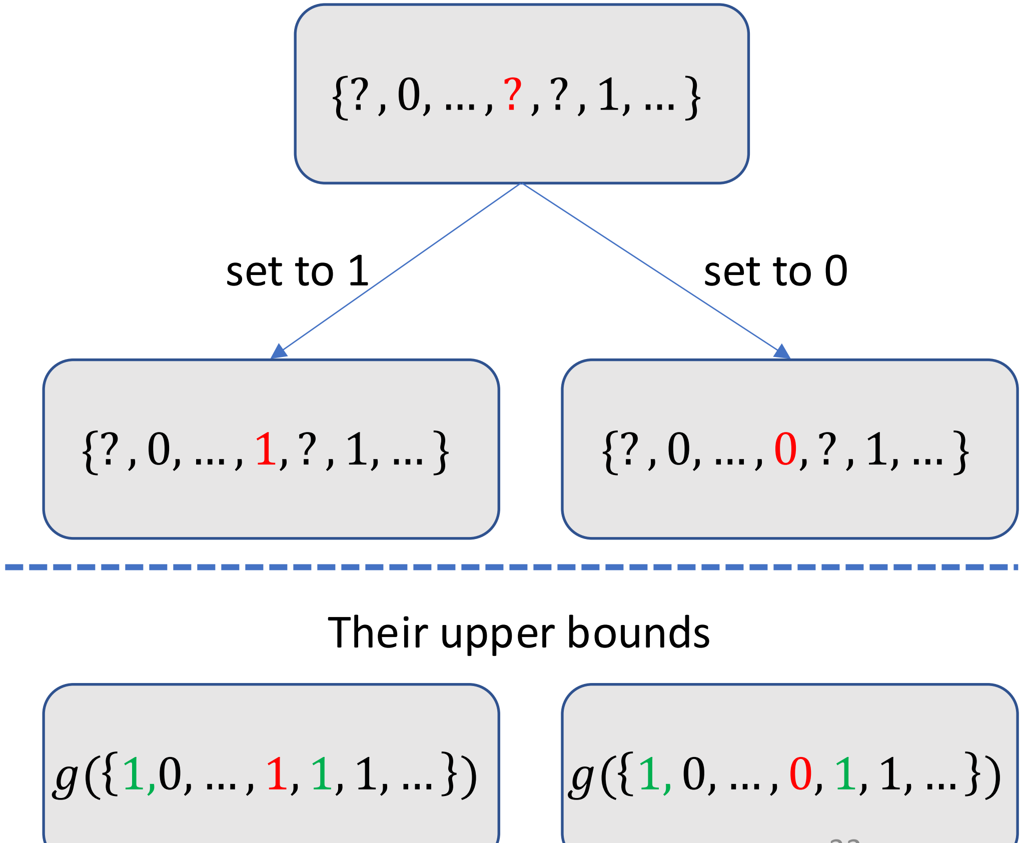


Source view candidates

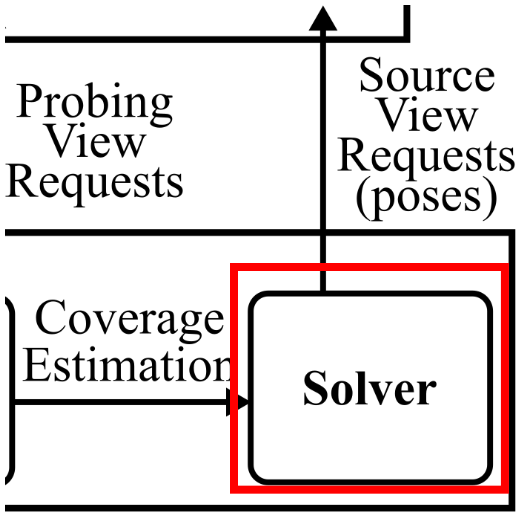
Branch & Bound Solver (BB)



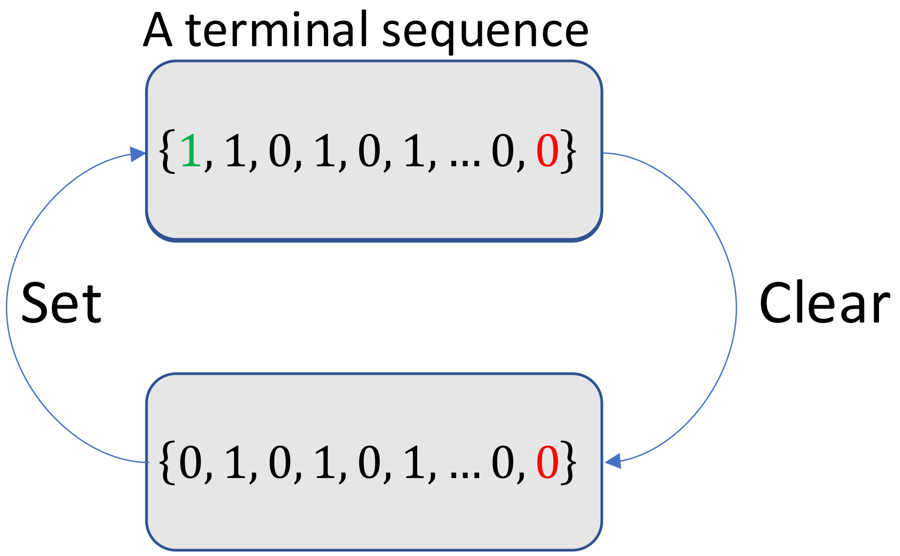
- Start from $\{s_j\} = 0$, mark all s_j as "undetermined"
- $ub(\{s_j\}) = g$ value of setting all "undetermined" s_j to 1
- **Branch**
  1. Set one of the 0s to 1 such that g value increases the most
  2. Mark the corresponding 0 in **Branch 1.** as "determined"
- **Bound**
  - lb: g value of the best sequence
  - Remove from list if $\mathrm{ub}(\{s_j\}) \leq \mathrm{lb}$



Uniform & Modify Solver (UM)

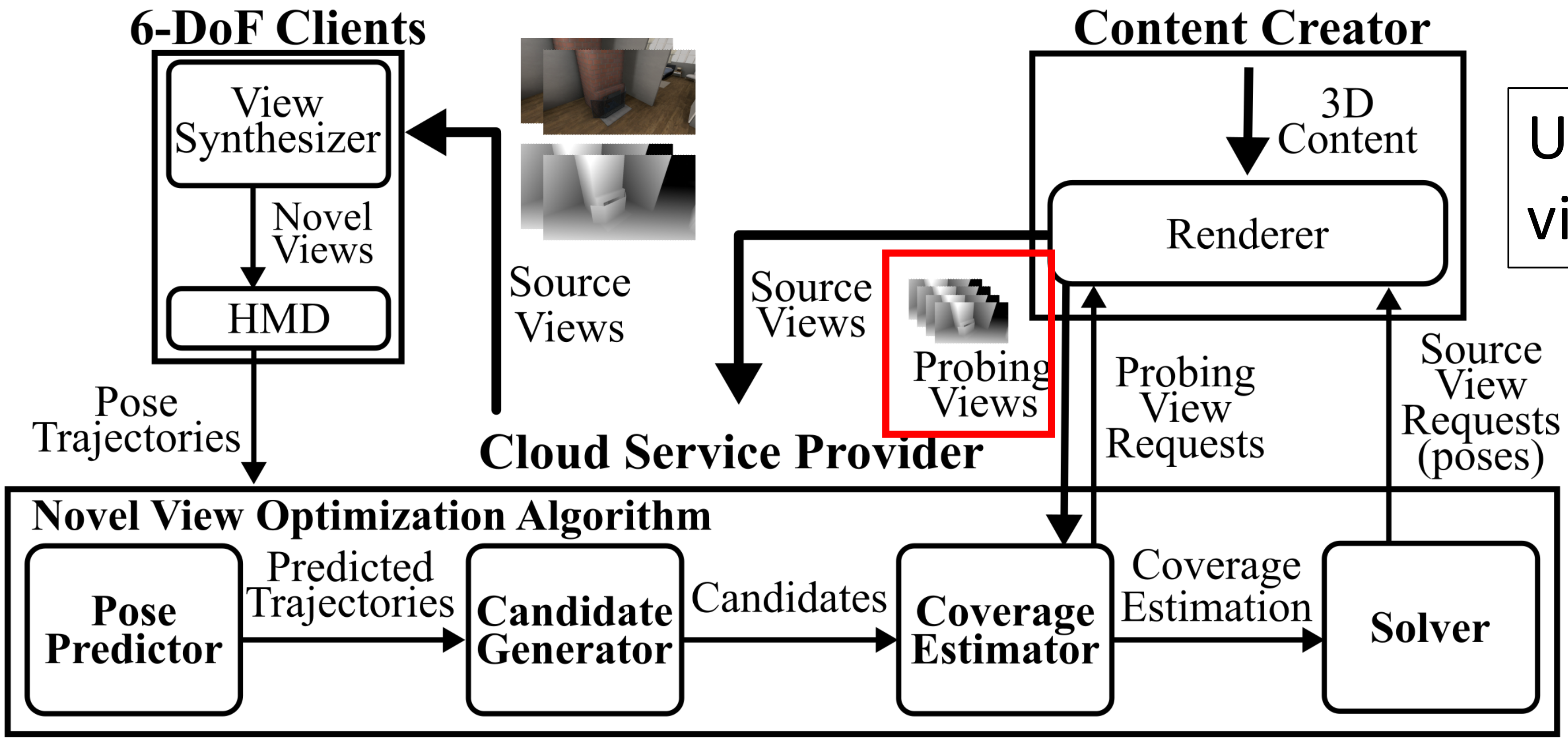


- Start from $\{s_j\} = \mathrm{Uni}()$
- Always iterate in the terminal sequences
  - Terminal sequence: $\sum s_j = N$
- Clear-than-set iteration
  - **Clear** one of the 1s to 0 such that g value decreases the least
  - **Set** one of the 0s to 1 such that g value increases the most
  - Duplicated $\{s_j\}$ are ignored



Component Diagram of Each Party

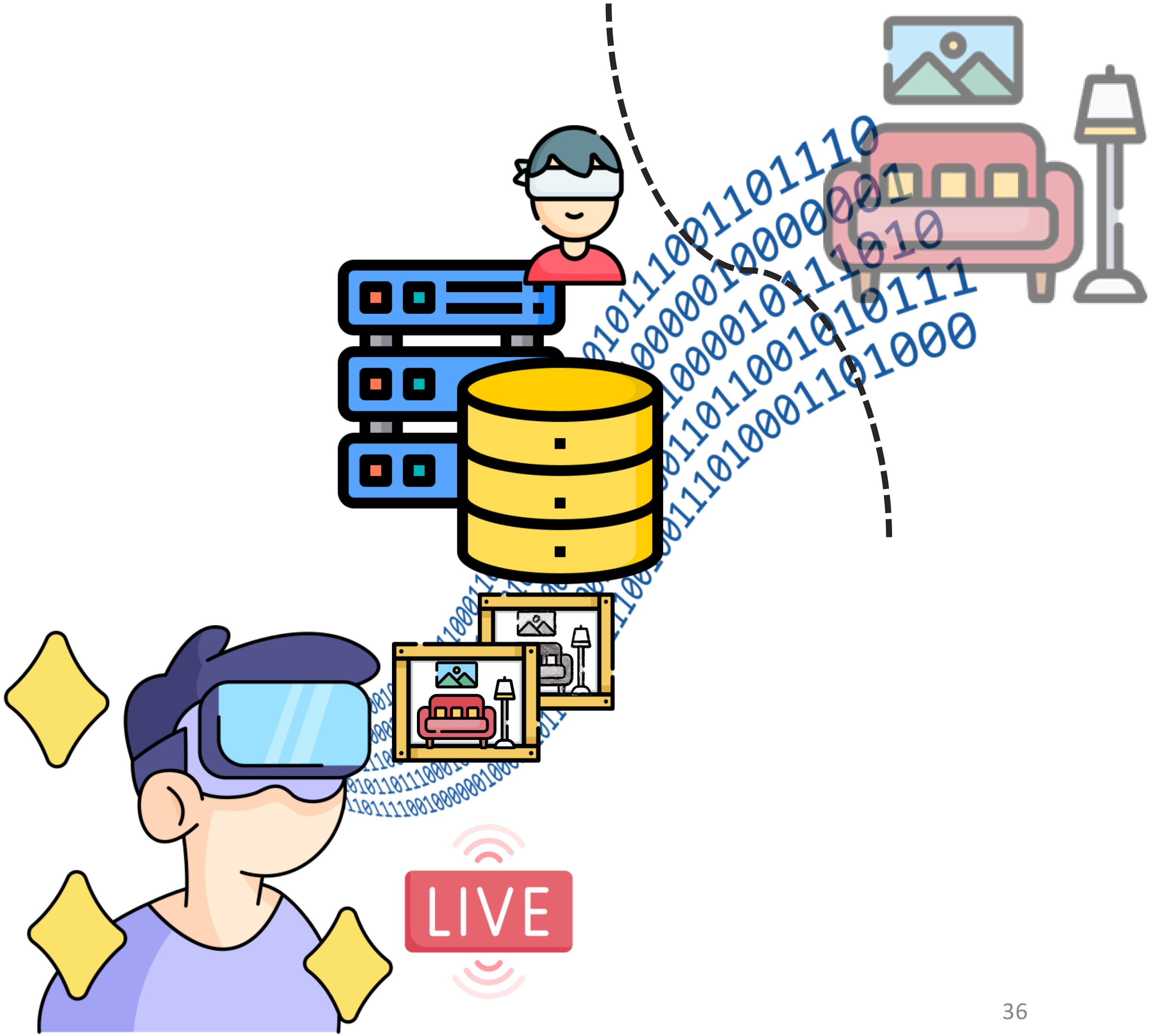
- Probing view: Low resolution depth image (1/16 of original)



Update source views every second

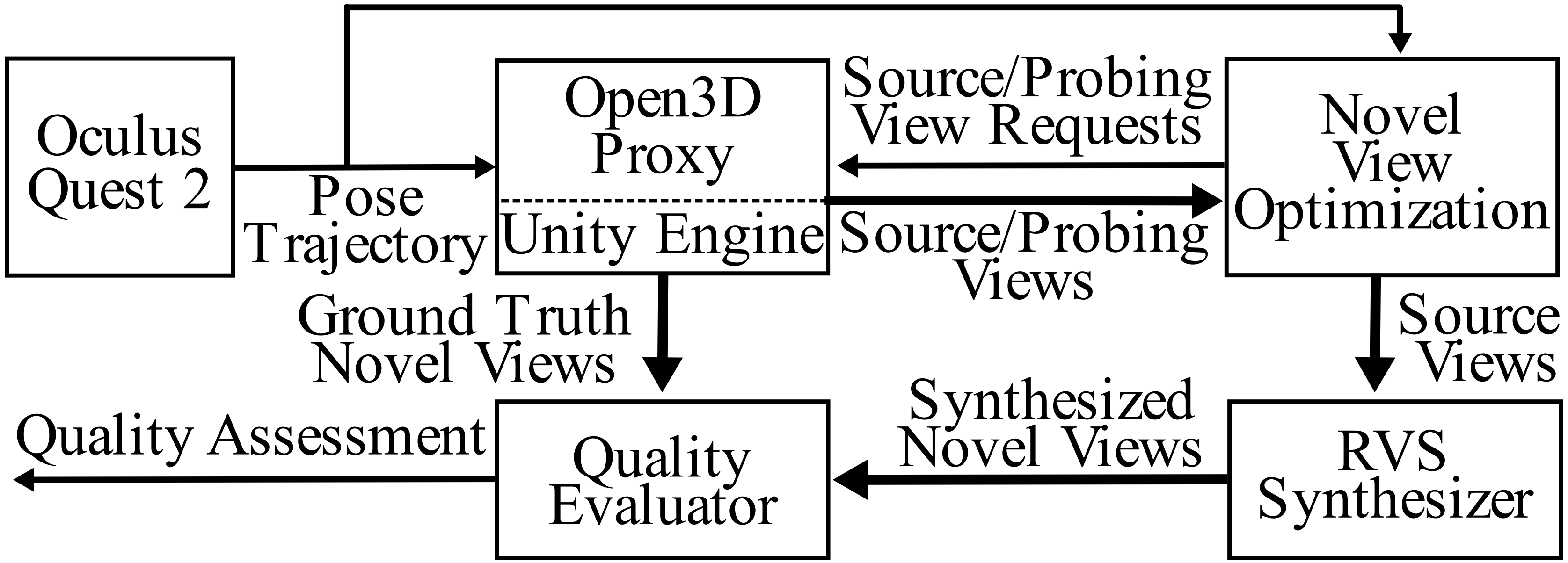
Outline

- Inspiration
- Goal & Challenges
- Related Work
- System Design
- Novel View Optimization
- Cloud Service Provider
    - Pose Predictor
    - Candidate Generator
    - Coverage Estimator
    - Solver & Algorithms
- Implementation
- Evaluations
- Conclusion & Future Work



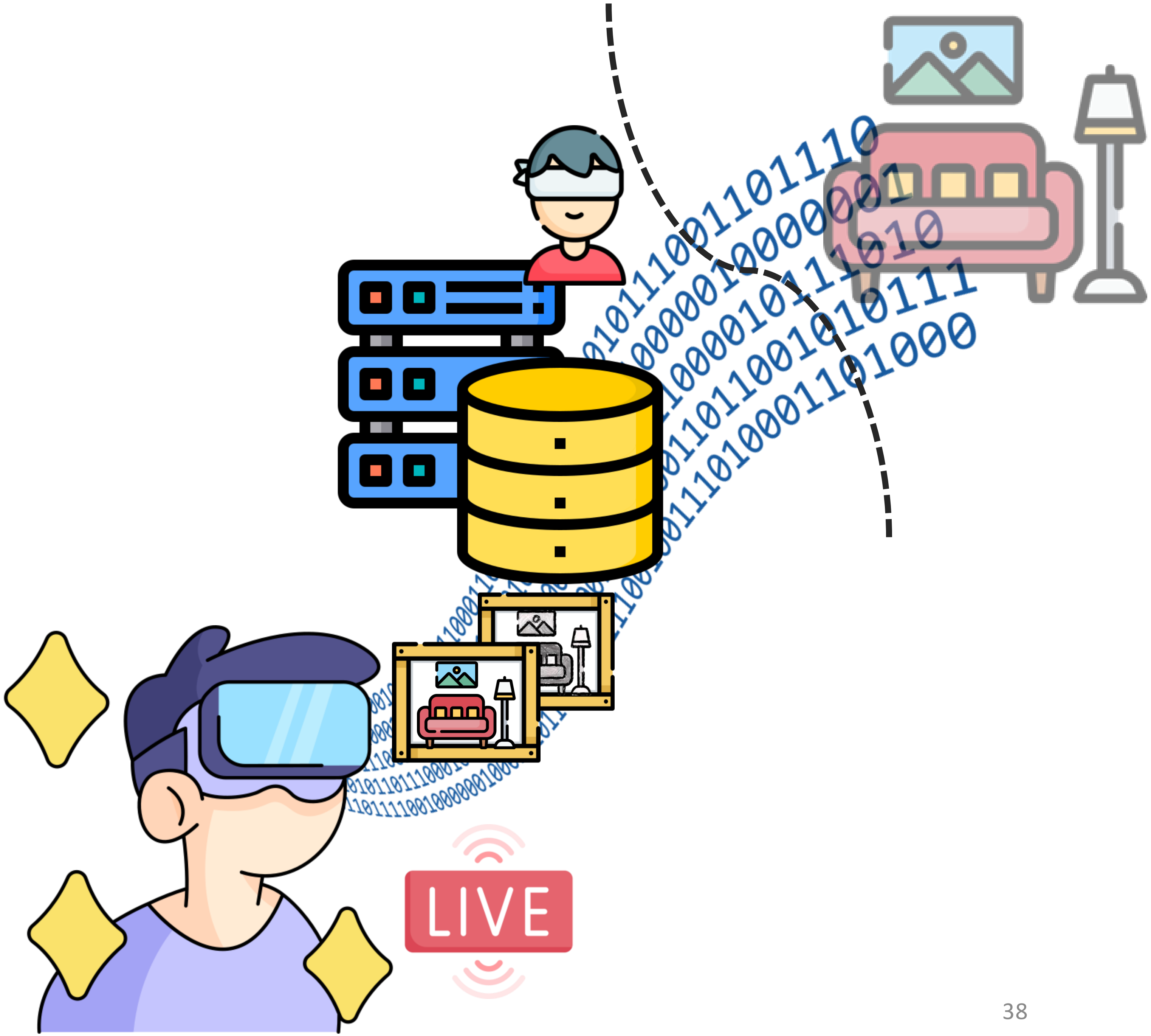
Testbed

- Render depth images using an Open3D renderer
- Offload RVS synthesizer to PC
- Unity Engine as high quality source view provider



Outline

- Inspiration
- Goal & Challenges
- Related Work
- System Design
- Novel View Optimization
- Cloud Service Provider
  - Pose Predictor
  - Candidate Generator
  - Coverage Estimator
  - Solver & Algorithms
- Implementation
- Evaluations
- Conclusion & Future Work



Comparison Algorithms / Candidate Generator (IXR'22)

Content creator provides scalar coverage ratio from a pose to another

$$\underset{\boldsymbol{s}}{\text{maximize}} \sum_{e \in \text{candidates}} \boldsymbol{w}[e] \text{qls}(\boldsymbol{s}^T B_e^* \boldsymbol{s})$$

$$\text{subject to : } \boxed{\boldsymbol{s} \in Q,}$$ source view budgets

- $\boldsymbol{s}$: Boolean column vector denote a selection
- Matrix approximation of set union operations

- S-Cdd
  - Generate a candidate if a pose cannot cover 75+% of the previous candidates
- C2I: Integer programming solver
- C2G: Greedily select the best 2 candidates at a time
- Opt
  - Select all the source views
  - Highest performance given candidates

SSIM $l(\mathbf{x},\mathbf{y}) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \cdot c(\mathbf{x},\mathbf{y}) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \cdot s(\mathbf{x},\mathbf{y}) = \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3}$

Setup

Content

House

Big Room

Small Room

Default parameters

- Number of 6-DoF clients = 16
- Source view budgets $N \in \{8, 16, \mathbf{24}, 32, 40\}$
- Candidates $M \in \{32, 32, \mathbf{48}, 64, 80\}$
- Solver $\in \{\mathrm{C2G}, \mathrm{C2I}, \mathrm{Uni}, \mathrm{BB}, \mathbf{UM}, \mathrm{Opt}\}$
- Candidate generator ∈ {S-Cdd, **proposed**}

Device specification

- CPU: AMD Ryzen 7 5700X 8-core
- GPU: NVIDIA Geforce RTX 3090 Ti

Quality Metrics

- Peak Signal to Noise Ratio (PSNR) = $20 \log(255/\sqrt{\mathrm{MSE}})$
- Structural Similarity (SSIM)
- Video Multi-Method Assessment Fusion (VMAF)

Sample results

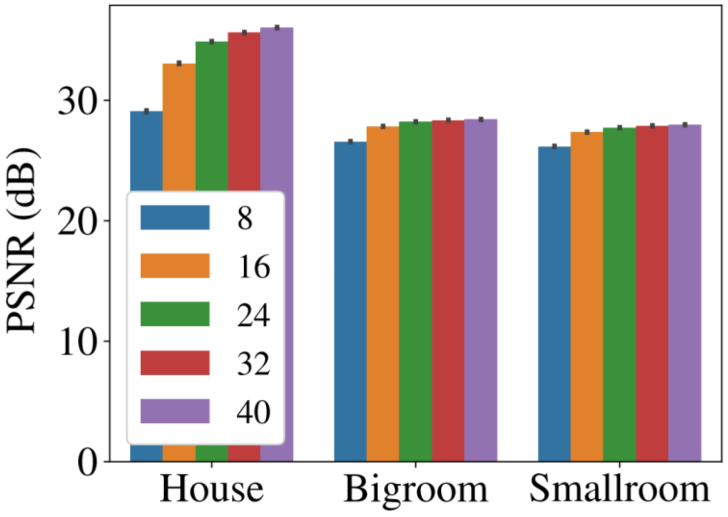
- Best frame in PSNR from a random synthesized video

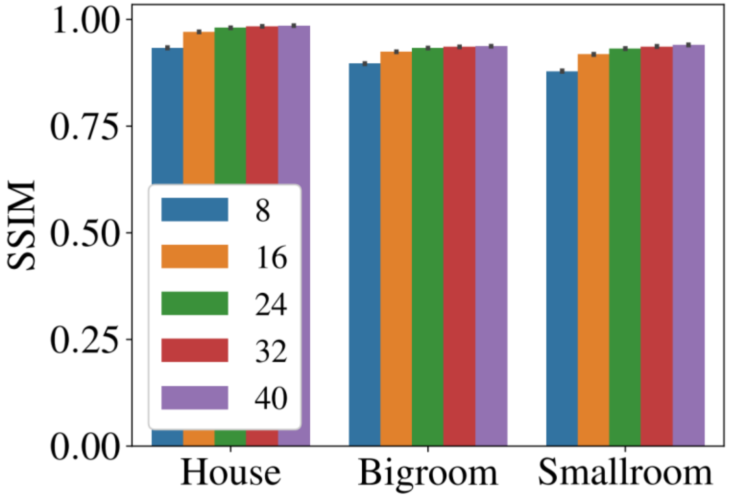
- Worst frame in PSNR from a random synthesized video
- Artifacts: blur, distortion

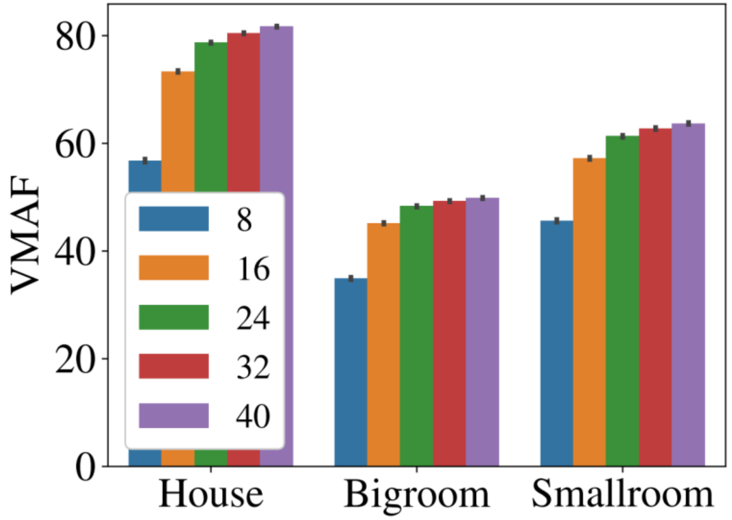
See demo videos

Quality Saturates as N increases

- Quality saturates when $N = 24$
- VMAF performs relatively worse
  - Our formulation does not consider temporal continuity

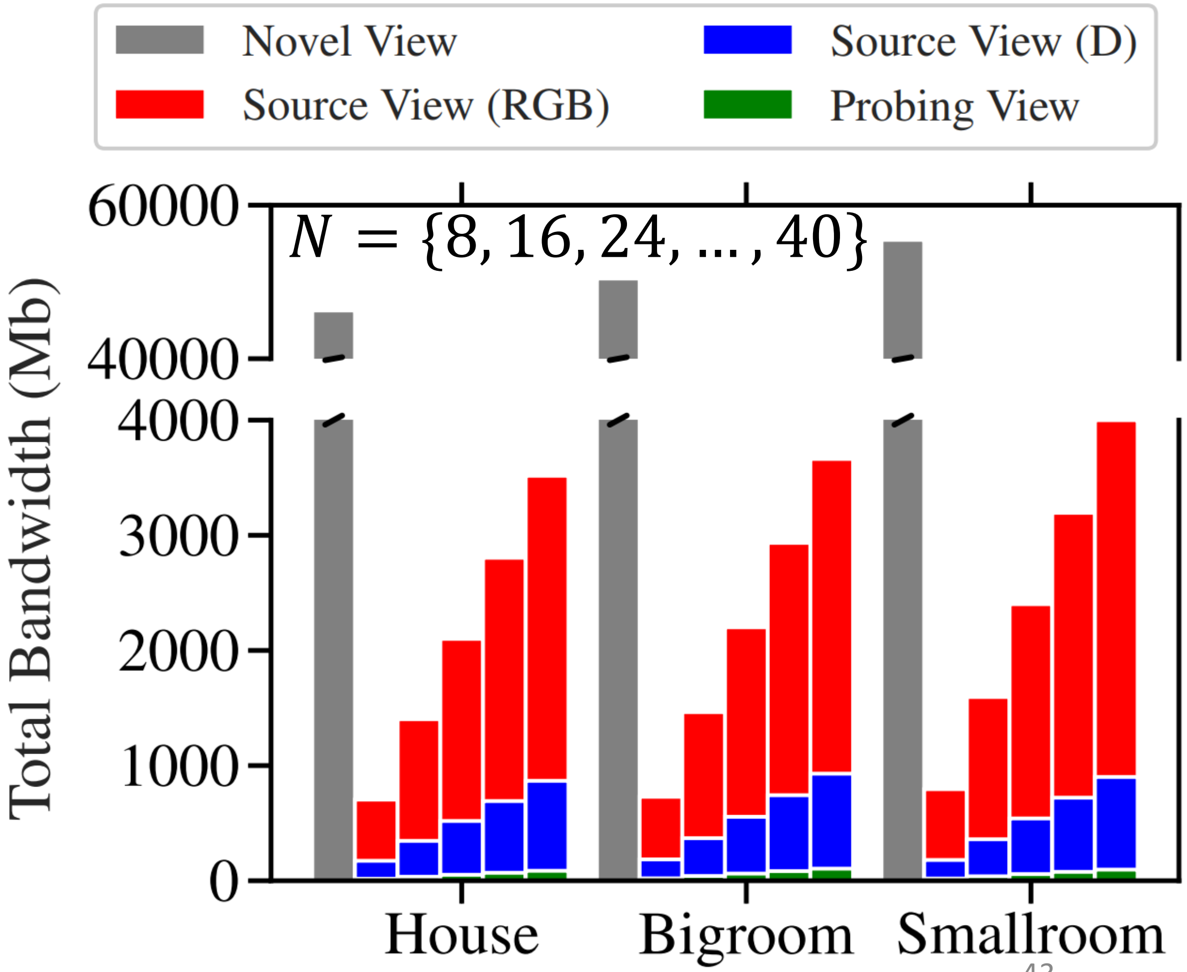






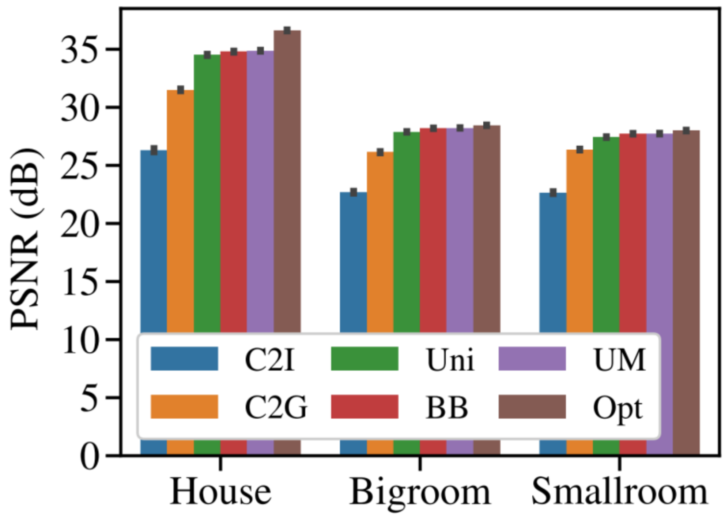
Bandwidth Reduction

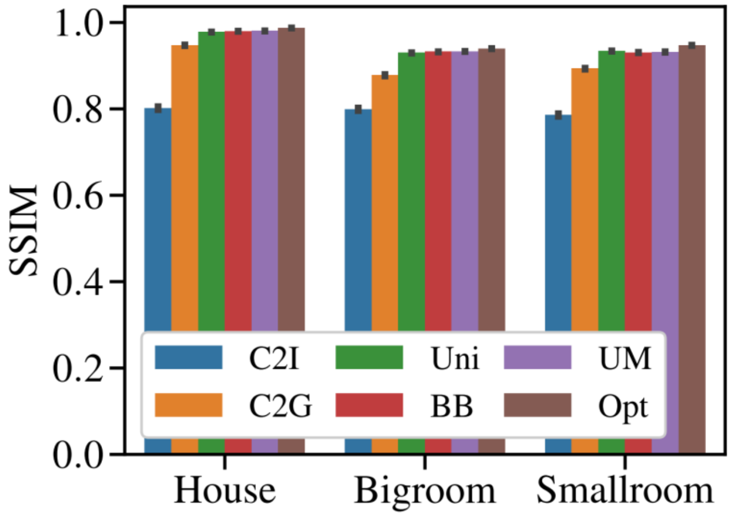
- H.264 encoder, quantization parameter (QP) = 0
- Encode ground truth video at 50 fps
- Encode source views separately
- Save 94% of bandwidth consumption

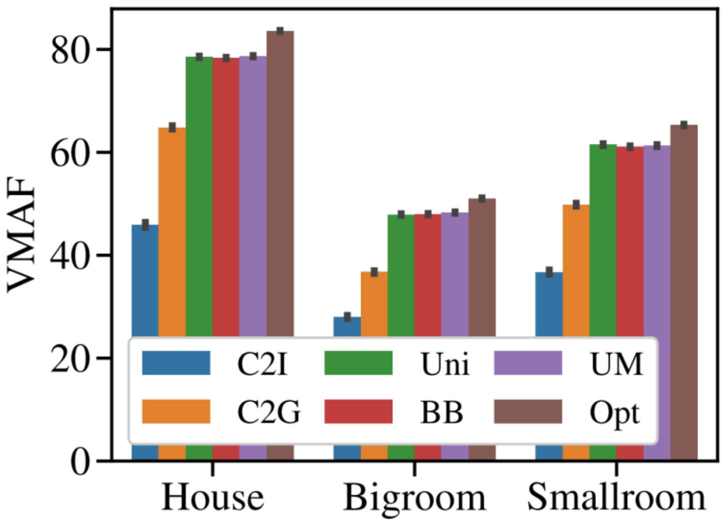


Solver Comparison

- C2I, C2G only have scalar coverage ratio information
- Uni, BB, UM outperform C2I, C2G
- Uni, UM seek for improvement over Uni in PSNR

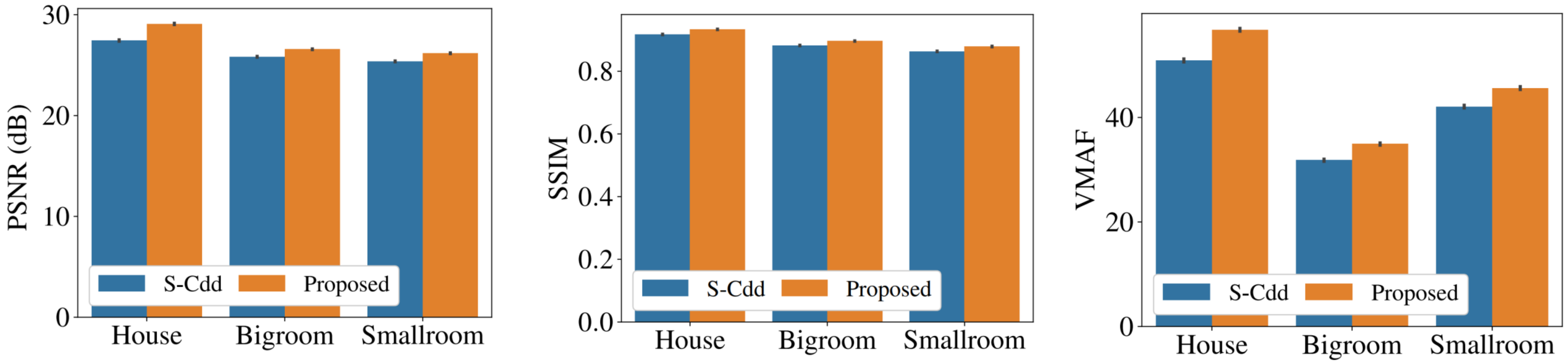






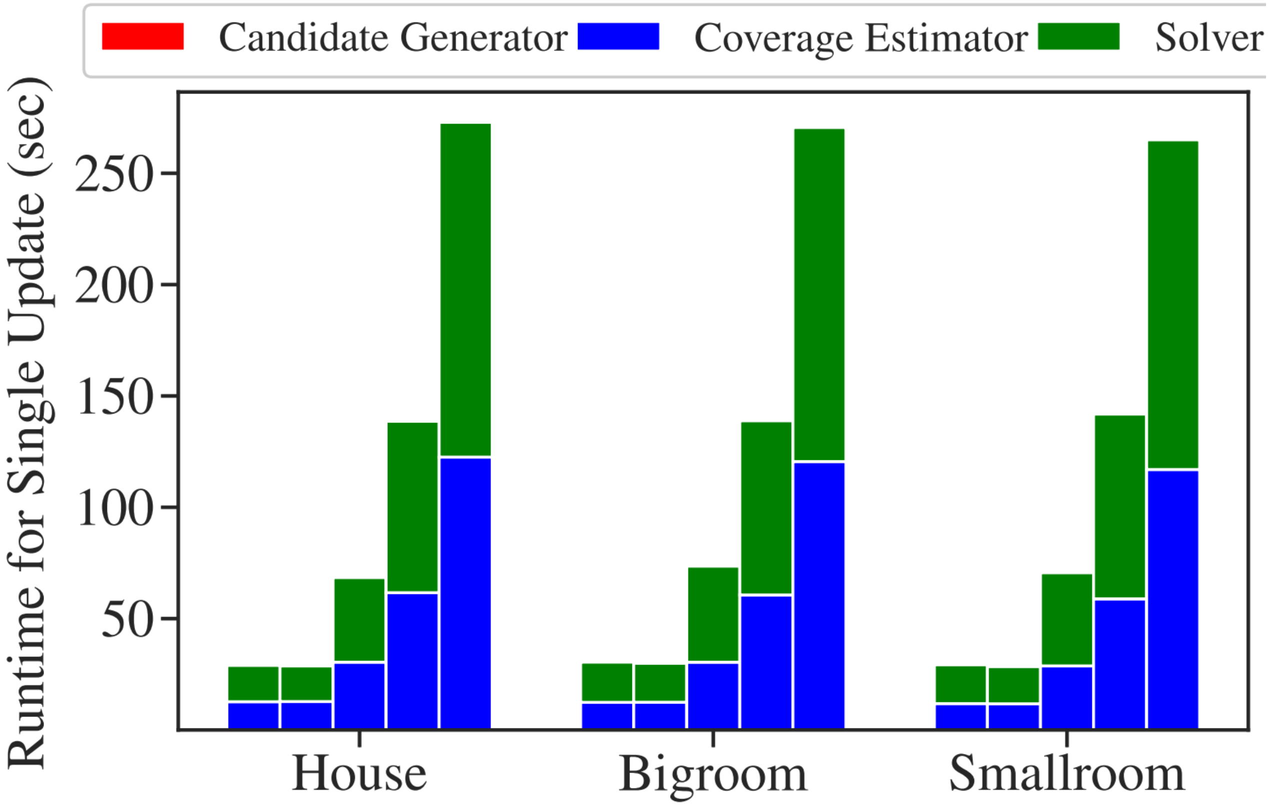
Candidate Generator Comparison

- Solver = UM
- Proposed generator consistently outperforms S-Cdd
- Proposed generator feeds high quality inputs to the pipeline



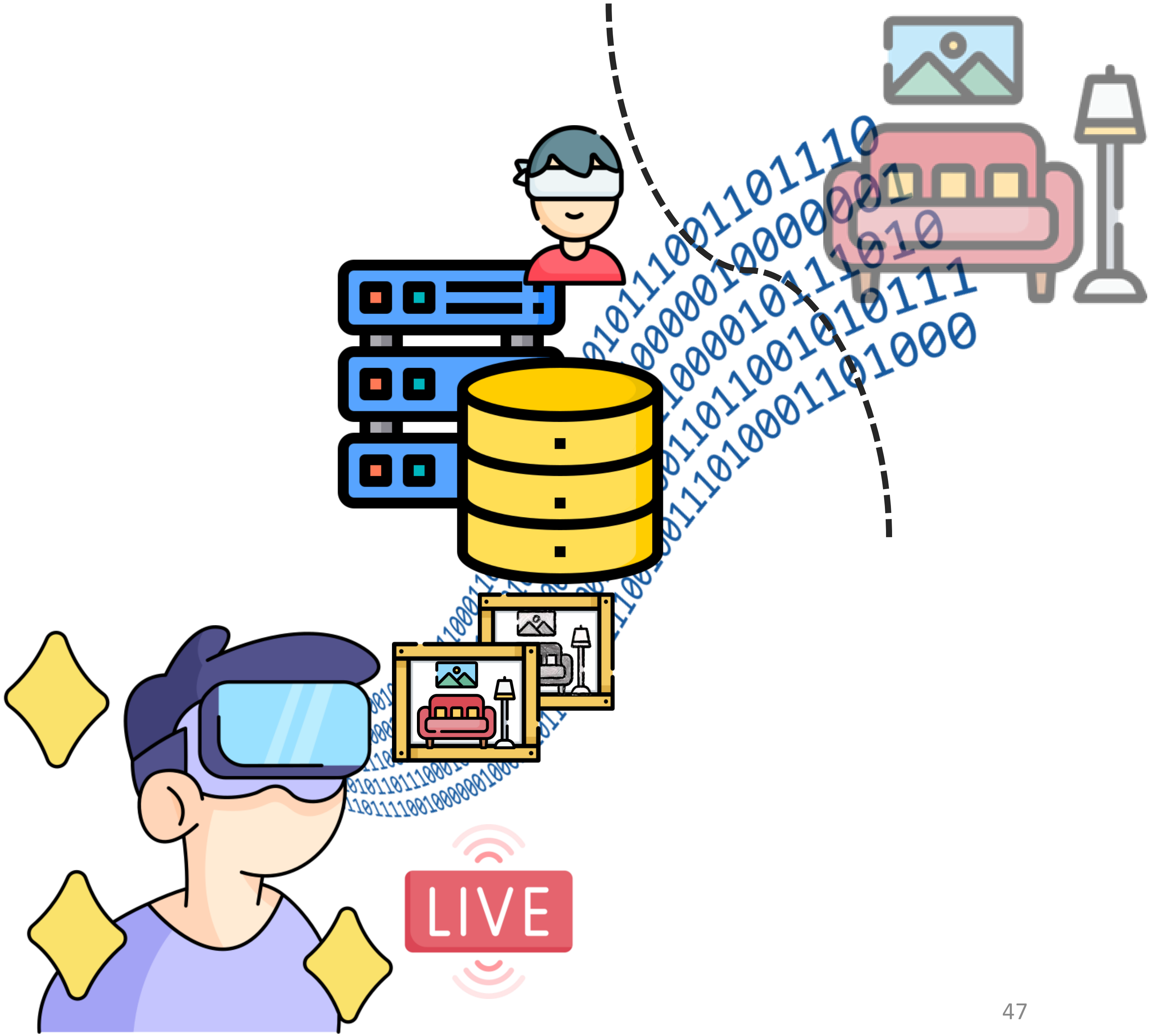
Runtime Distribution

- Solver = UM
- Number of iterations = 128
- Candidate generator runs fast
- Coverage estimator is implemented in CPU
- Solver is implemented in GPU
  - Frequently evaluate g value



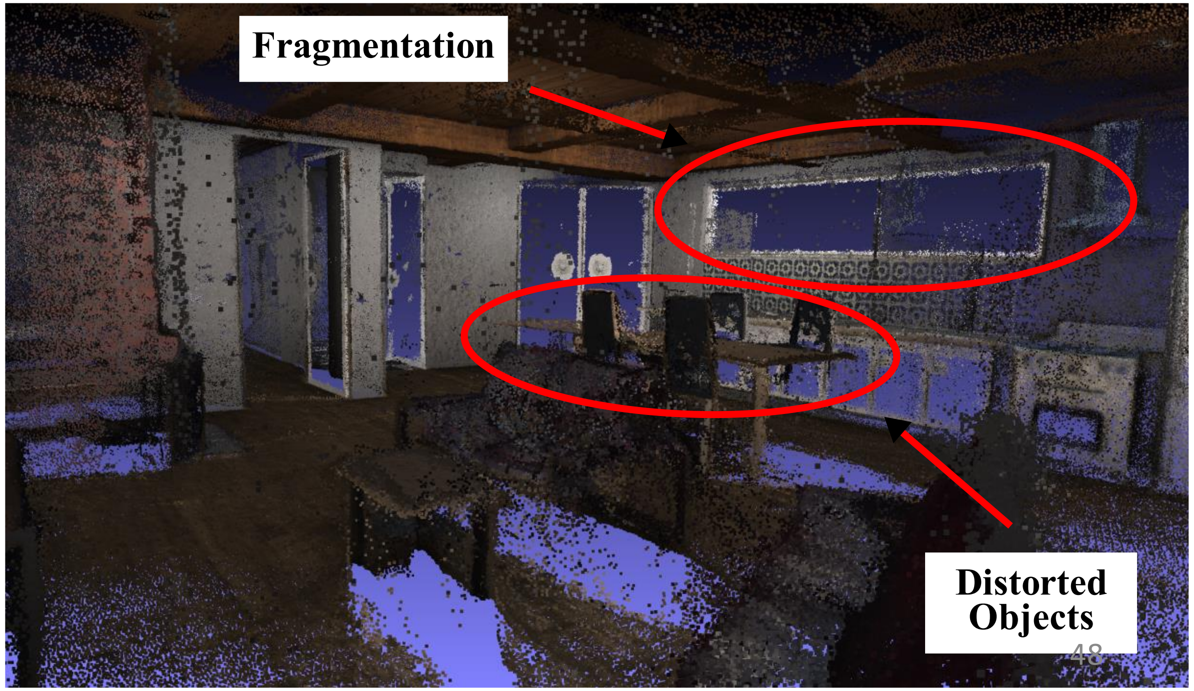
Outline

- Inspiration
- Goal & Challenges
- Related Work
- System Design
- Novel View Optimization
- Cloud Service Provider
  - Pose Predictor
  - Candidate Generator
  - Coverage Estimator
  - Solver & Algorithms
- Implementation
- Evaluations
- Conclusion & Future Work



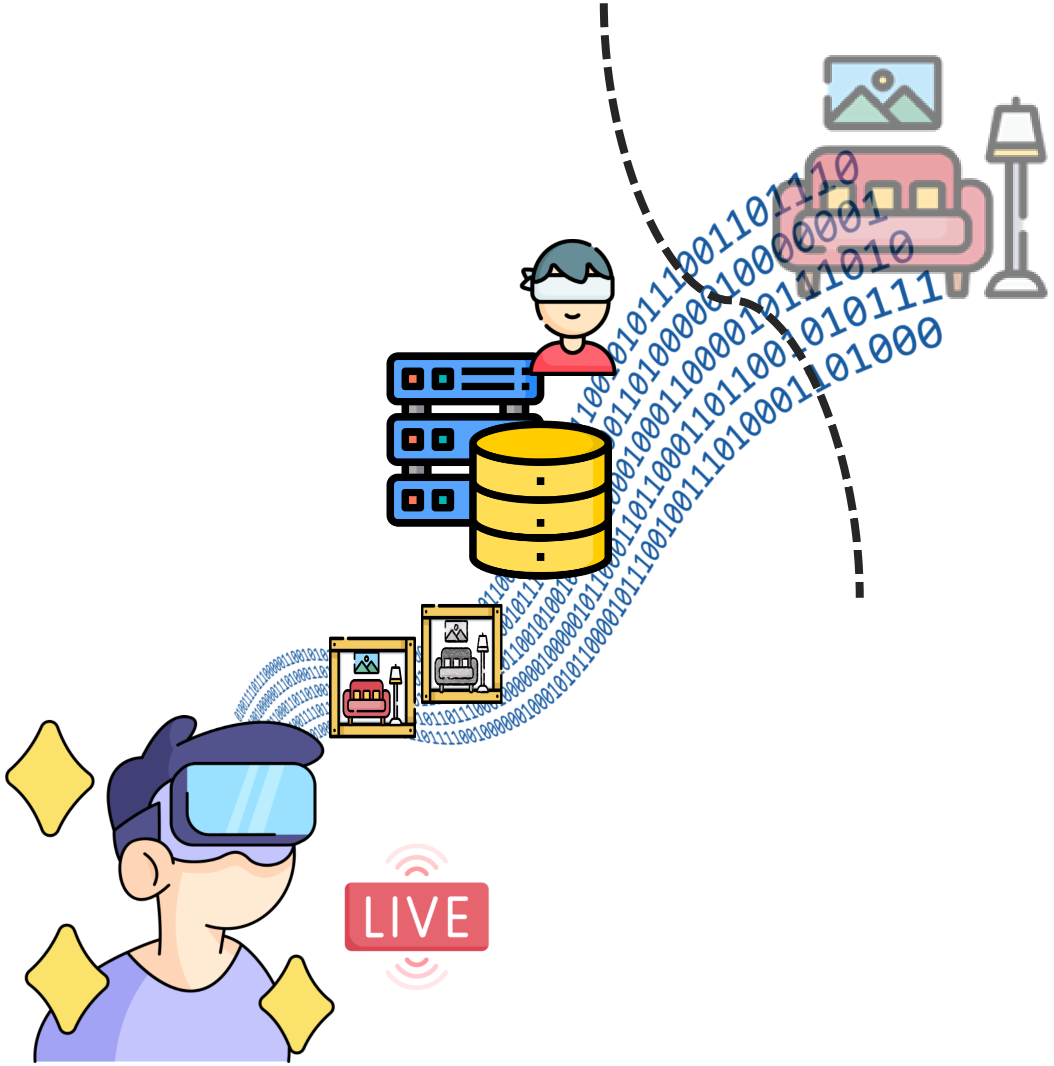
Defense against Structure-from-Motion (SfM)

- Colmap
  - J. L. Schonberger and J.-M. Frahm. Structure-from-motion revisited. In ¨ Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR'16), Las Vegas, NV, June 2016.
- 720 color images with 960x540 resolution, 10+ GPU hours



Remarks

- Conclusion
    1. Propose a content creator friendly blind streaming system
    2. Compute coverage maps without access to 3D content
    3. Improve quality by 2.27 dB in PSNR, 12 in VMAF compared to scalar coverage ratio blind streaming system
- Future work
    1. Parallelism in frequently-evaluated g values
    2. Employ real-time view synthesis in HMDs
    3. Formulate optimization objective that considers temporal continuity



Thank you for your attention!

Sheng-Ming Tang (shengming0308@gapp.nthu.edu.tw)

Thanks for the help of Prof. Hsu, the committees,
Ching-Ting Wang, Yuan-Chun Sun, Jia-Wei Fang, Kuan-Yu Lee, and all lab mates.

Publications:

- **S. Tang**, Y. Sun, J. Fang, K. Lee, C. Wang and C. Hsu, ”Optimal Camera Placement for 6 Degree-of-Freedom Immersive Video Streaming Without Accessing 3D Scenes”, in Proc. of Interactive eXtended Reality (IXR’22), Lisbon, Portugal, October 2022.
- **S. Tang**, C. Hsu, Z. Tian, and X. Su, ”An Aerodynamic, Computer Vision, and Network Simulator for Networked Drone Applications”, in Proc. of ACM Annual International Conference on Mobile Computing and Networking (MobiCom’21), New Orleans, USA, February 2022, Poster Paper.
- Y. Sun, **S. Tang**, C. Wang, and C. Hsu, ”On Objective and Subjective Quality of 6DoF Synthesized Live Immersive Videos”, in Proc. of ACM Multimedia Workshop on Quality of Experience in Visual Multimedia Applications (QoEVMA’22), Lisbon, Portugal, October 2022.