Understanding and Enhancing Visual Place Recognition through Embedding Space Interpretability and Uncertainty Estimation

Master's Degree Thesis

Candidate: Davide SFERRAZZA

Supervisors:

- **Prof. Carlo MASONE**
- **Dr. Gabriele BERTON**
- **Dr. Gabriele TRIVIGNO**

Politecnico di Torino

1859

Table of Contents

- Task
- Goals
- Understand Embedding Information
  - Methodology
  - Experiments
- Uncertainty Estimation
  - Methodology
  - Experiments
- Conclusions

Table of Contents

- Task
- Goals
- Understand Embedding Information
  - Methodology
  - Experiments
- Uncertainty Estimation
  - Methodology
  - Experiments
- Conclusions

Visual Place Recognition

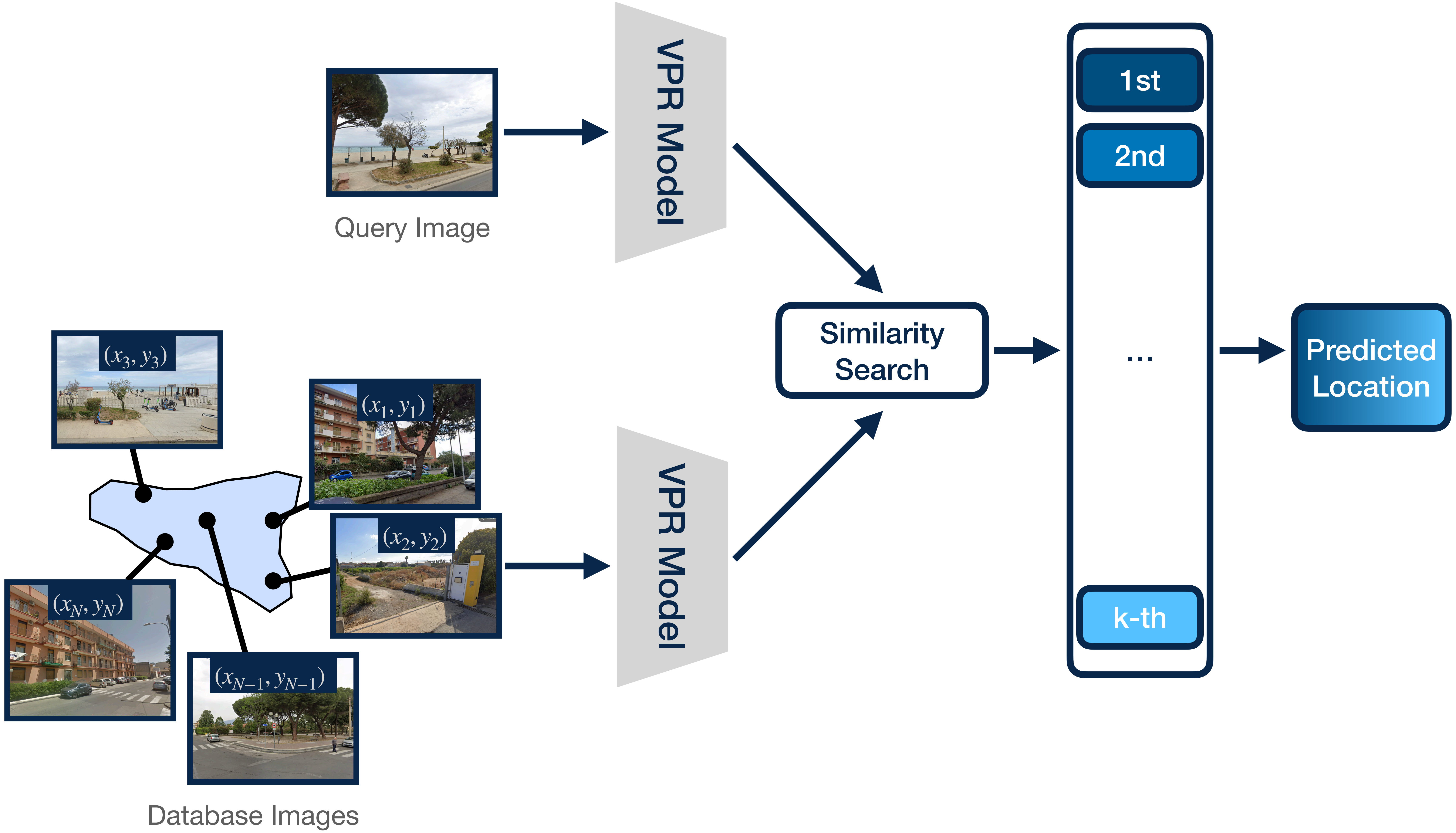
(VPR)

- **Question: «Where was this picture taken?»**
- **Input: only visual content**
- **Output: location prediction**

Query from SF-XL (*) test v1
(*) «Rethinking Visual Geo-localization for Large-Scale Applications» (CVPR 2022)

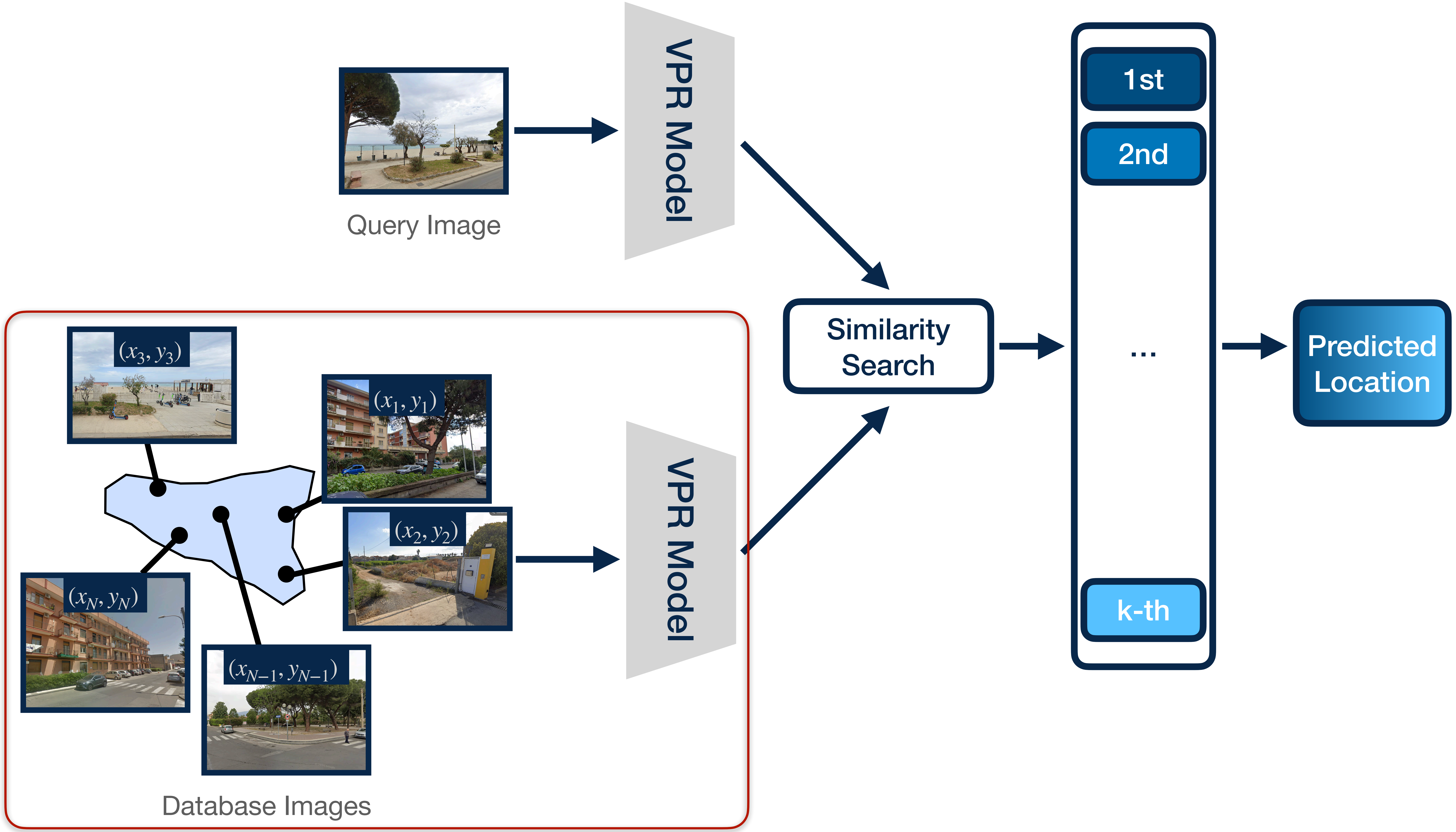
Visual Place Recognition

Pipeline



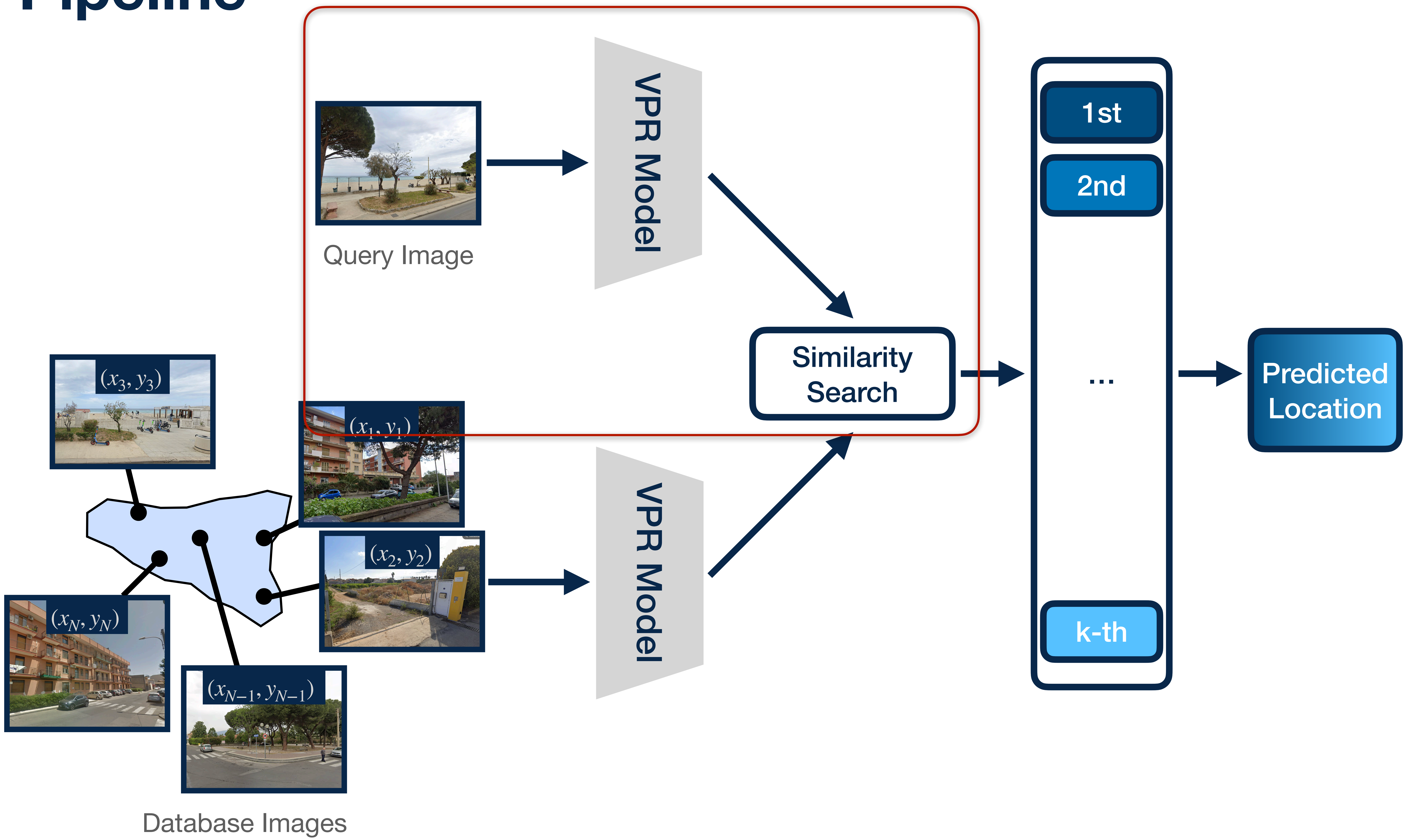
Visual Place Recognition

Pipeline



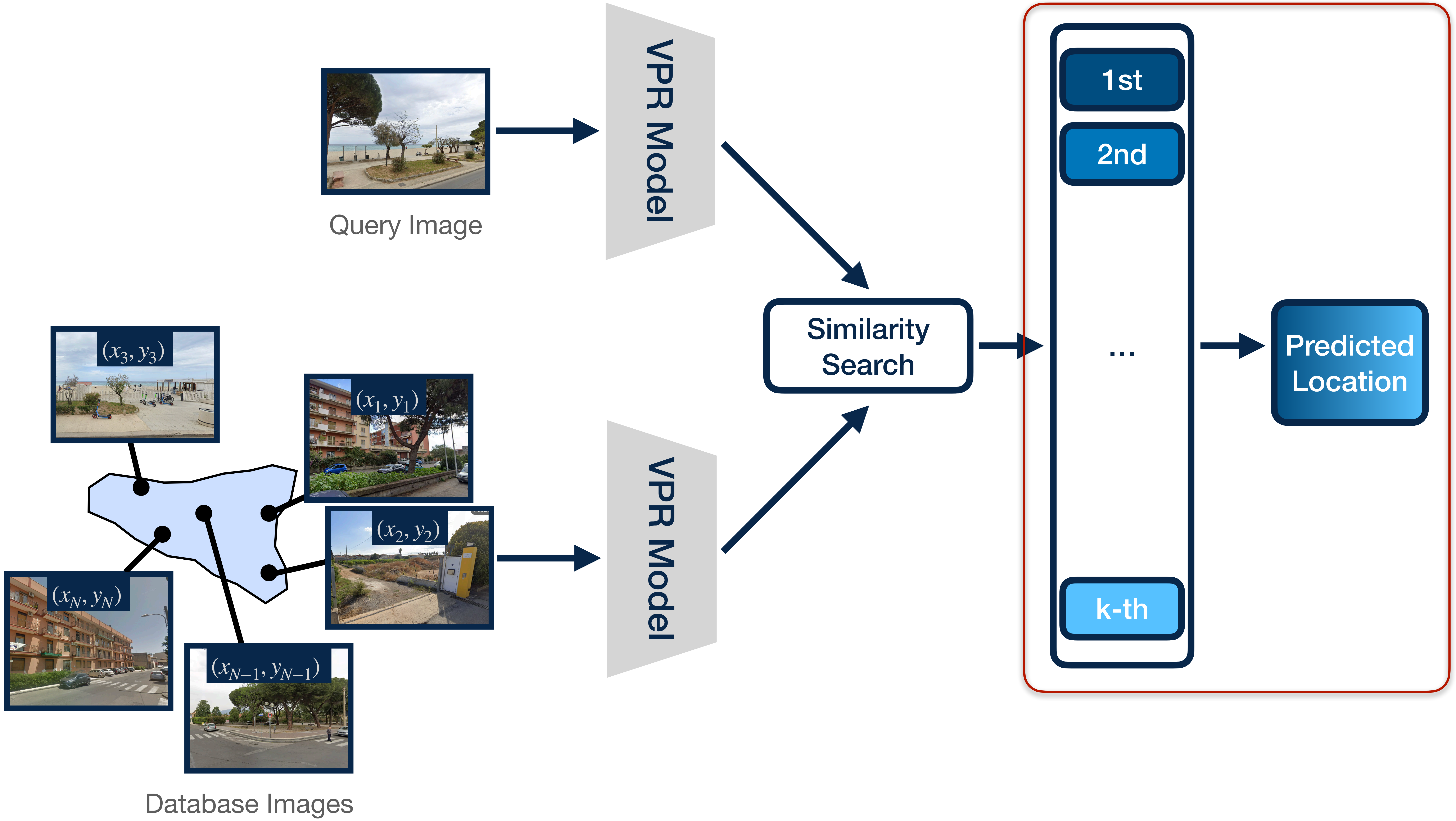
Visual Place Recognition

Pipeline



Visual Place Recognition

Pipeline



Visual Place Recognition
Pipeline

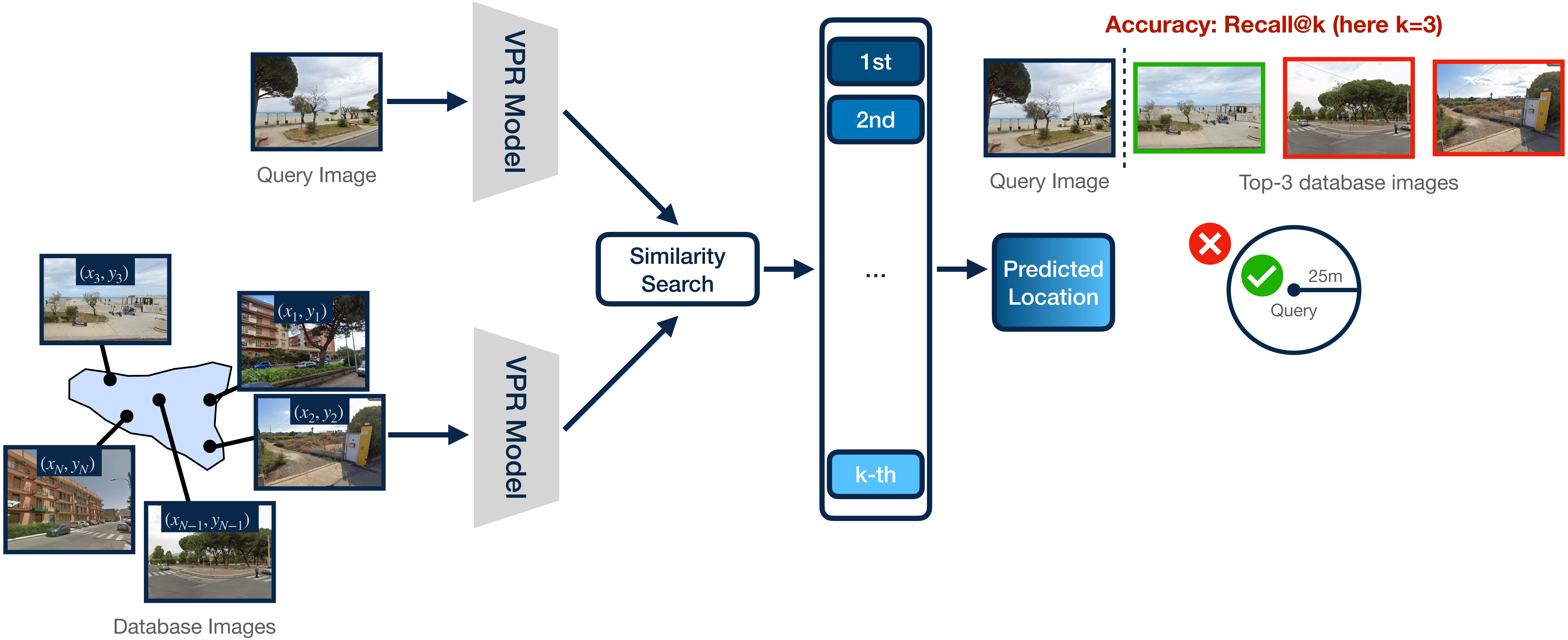


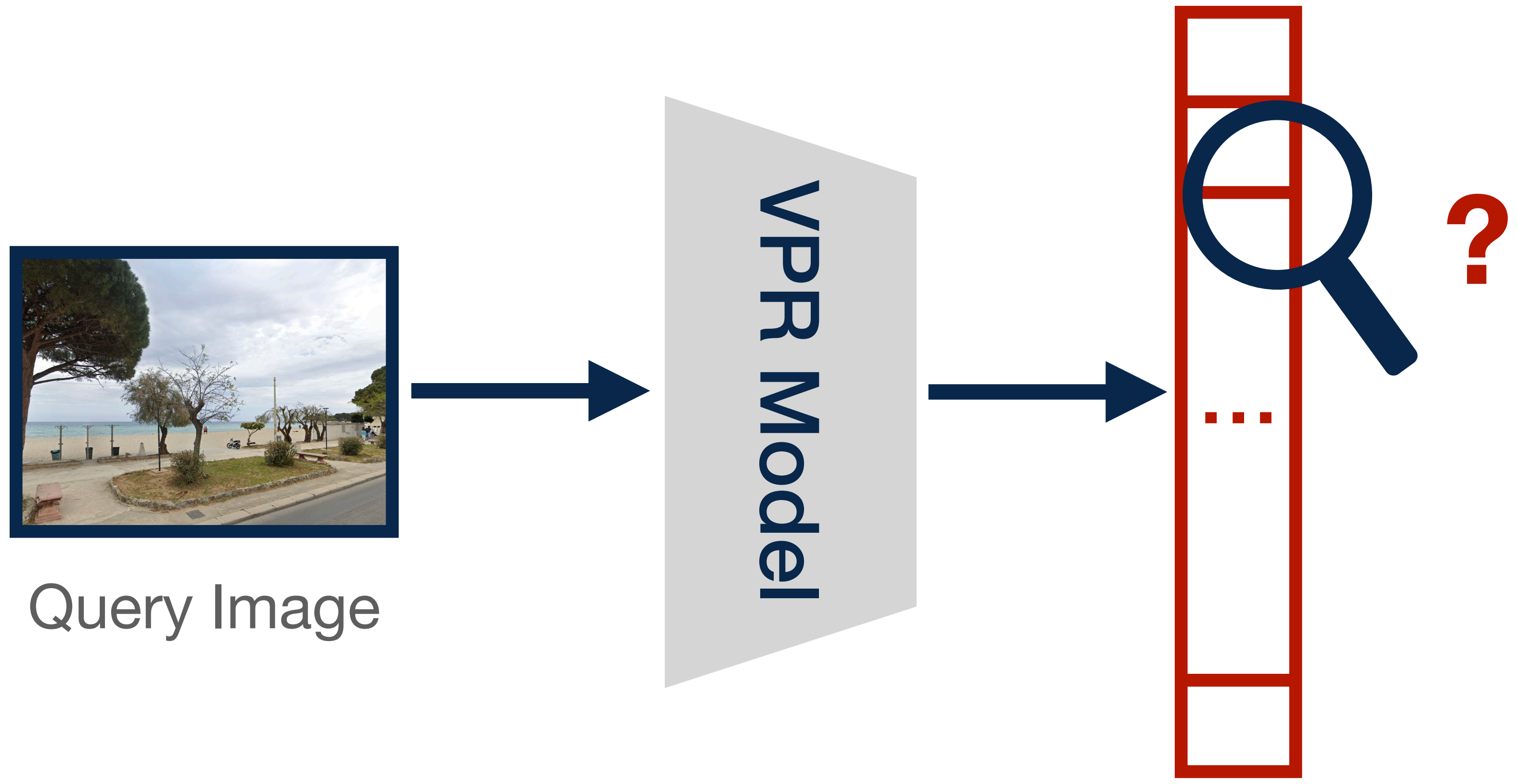
Table of Contents

- Task
- **Goals**
- Understand Embedding Information
  - Methodology
  - Experiments
- Uncertainty Estimation
  - Methodology
  - Experiments
- Conclusions

Goals

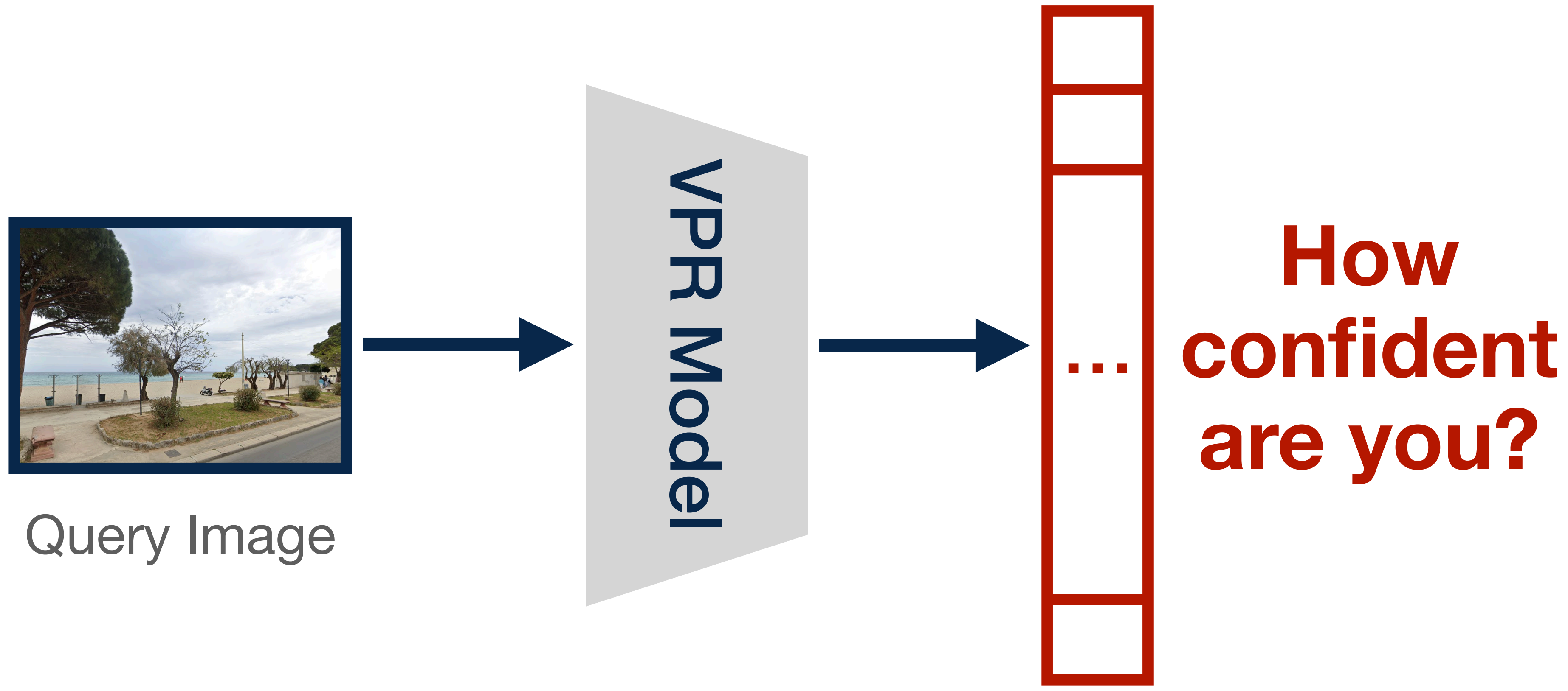
Embedding Information Inspection

- What **information** is **retained** in image **embeddings?**



Uncertainty Estimation

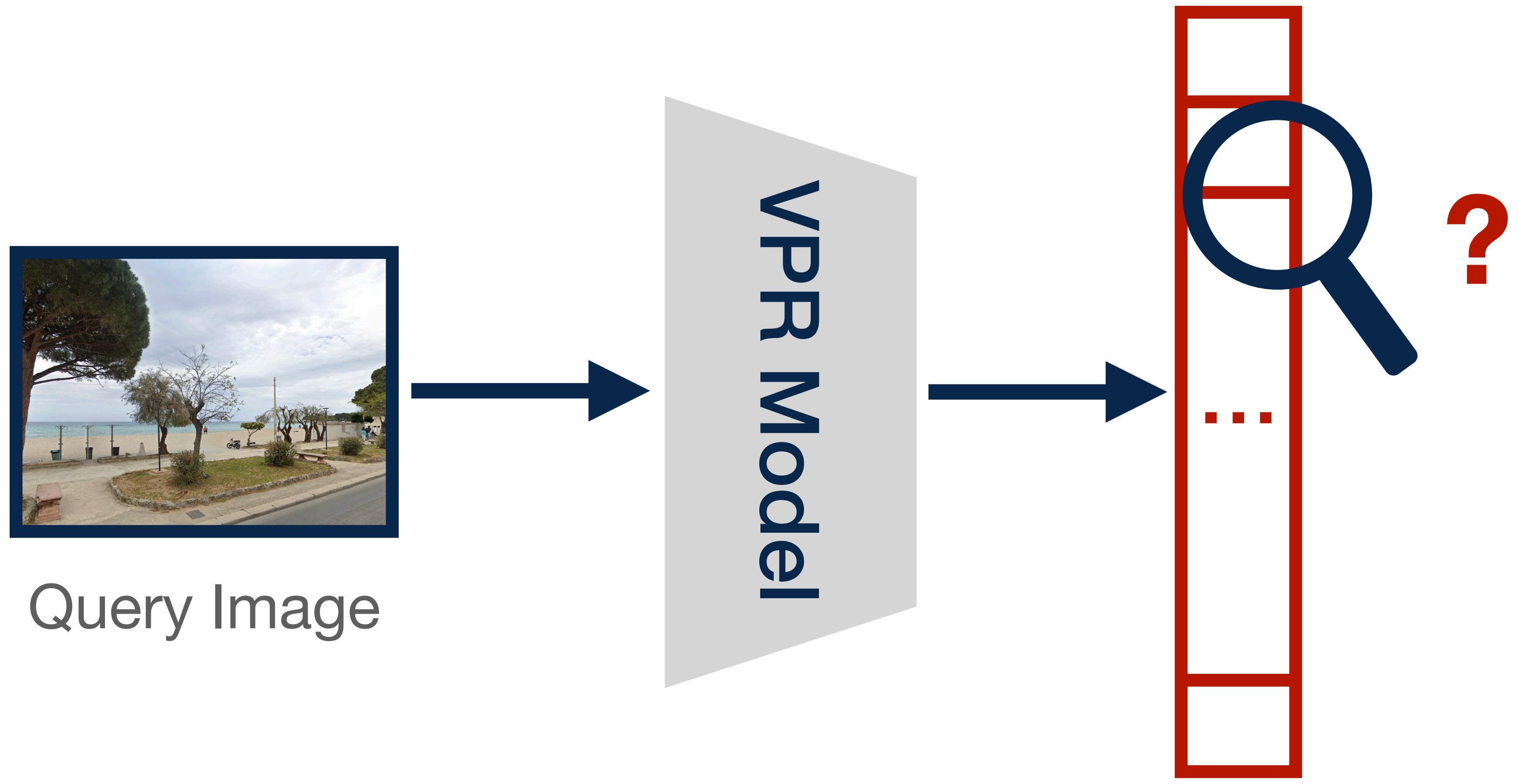
- **How certain** is the **model** in its **predictions?**



Goals

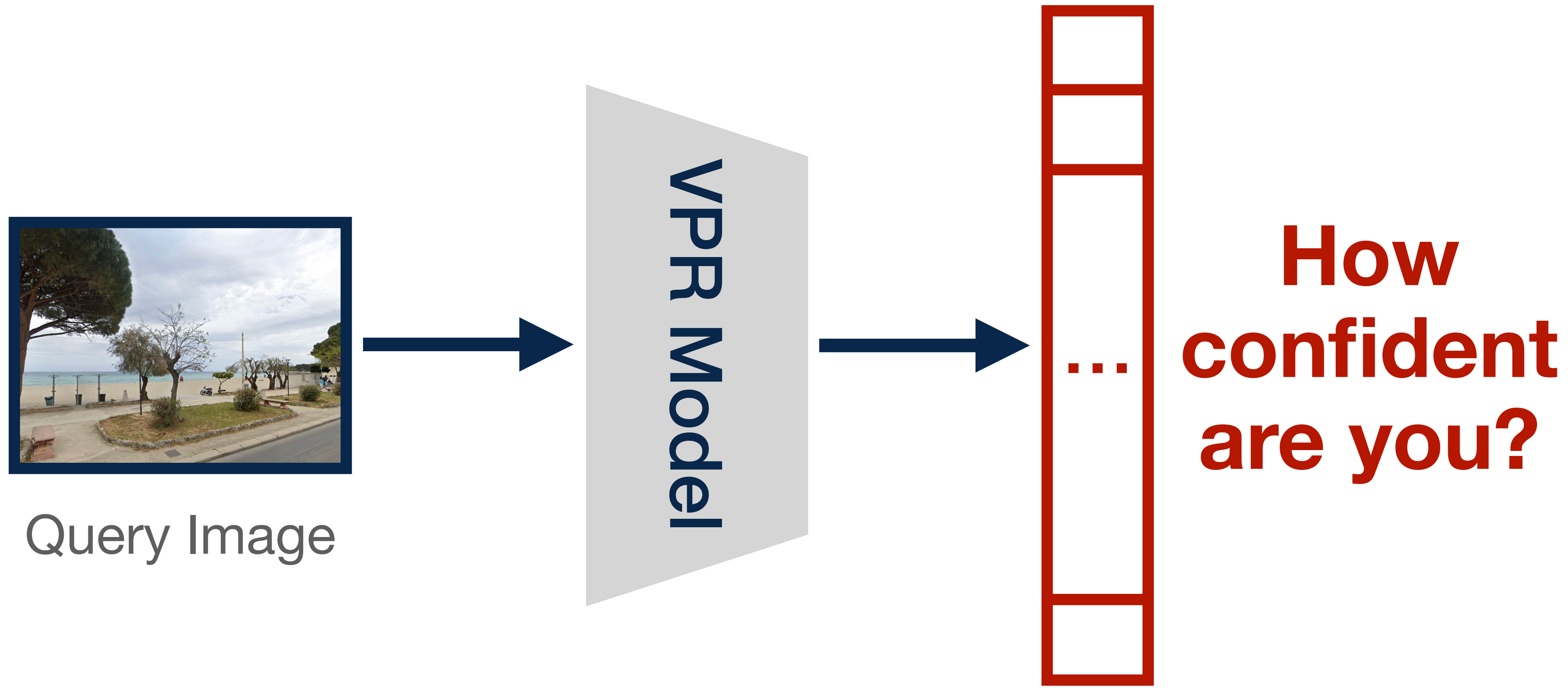
Embedding Information Inspection

- What **information** is **retained** in image **embeddings**?



Uncertainty Estimation

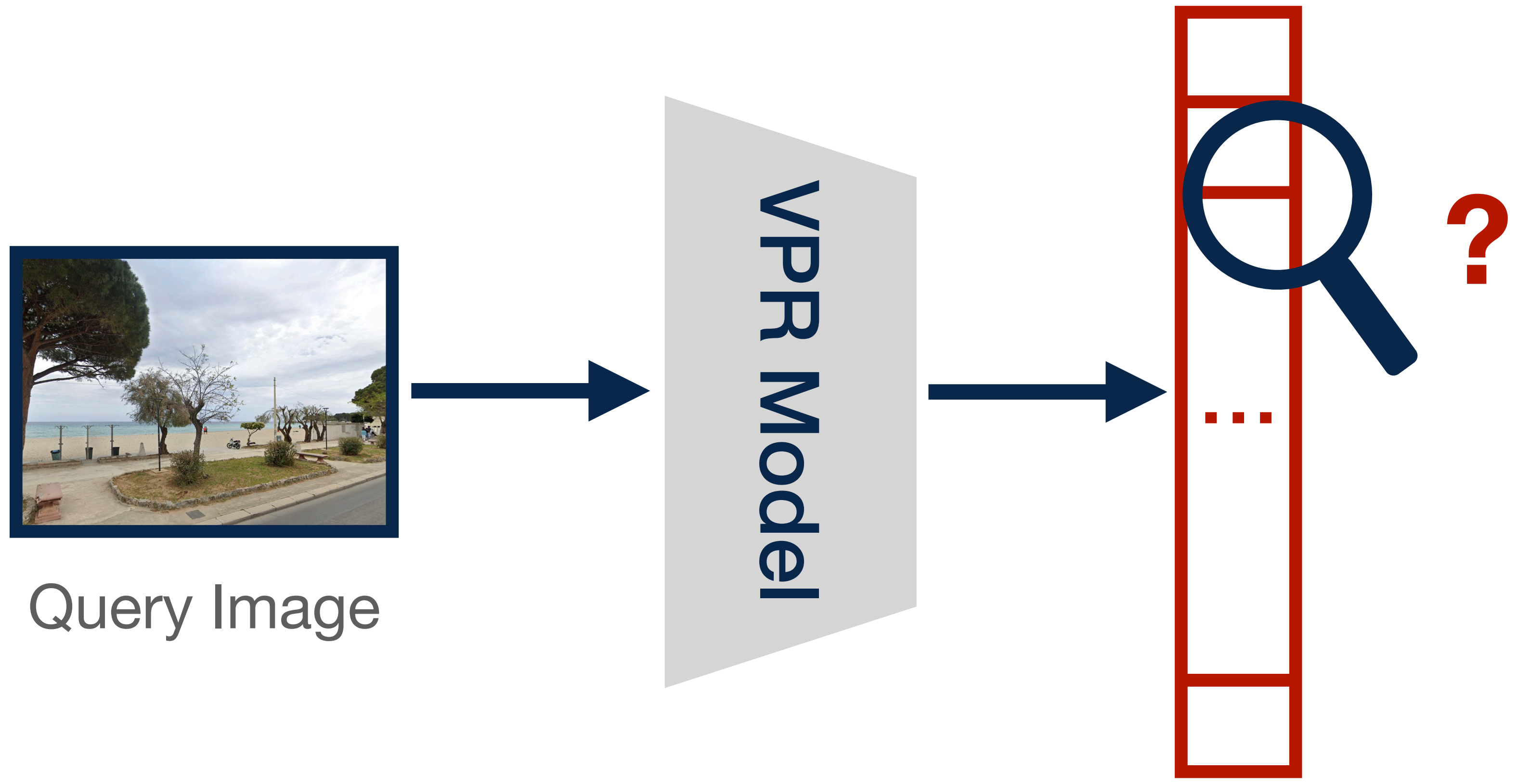
- **How certain** is the **model** in its **predictions**?



Goals

Embedding Information Inspection

- What **information** is **retained** in image **embeddings**?



Uncertainty Estimation

- **How certain** is the **model** in its **predictions?**

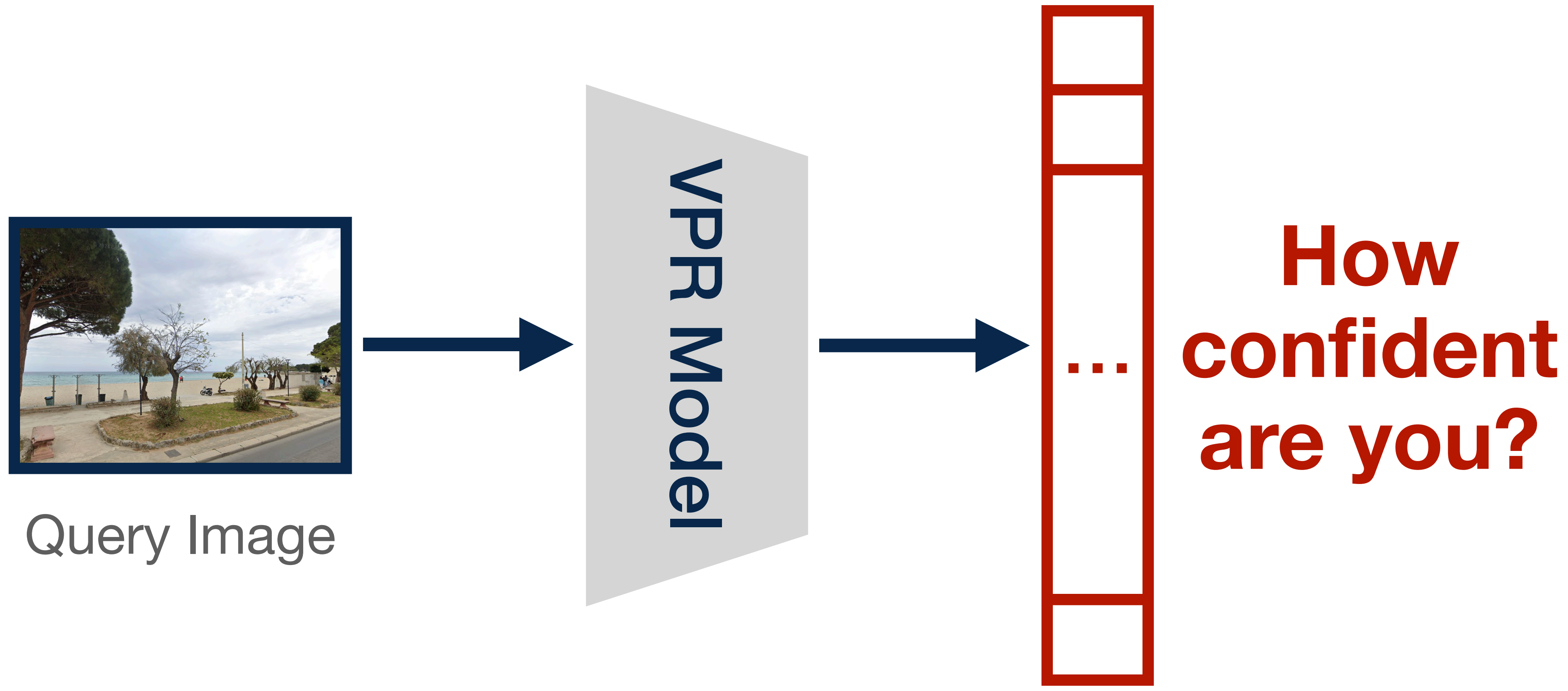


Table of Contents

- Task
- Goals
- **Understand Embedding Information**
  - **Methodology**
  - Experiments
- Uncertainty Estimation
  - Methodology
  - Experiments
- Conclusions

Image Generation via Diffusion Models

Generative AI Models

Midjourney

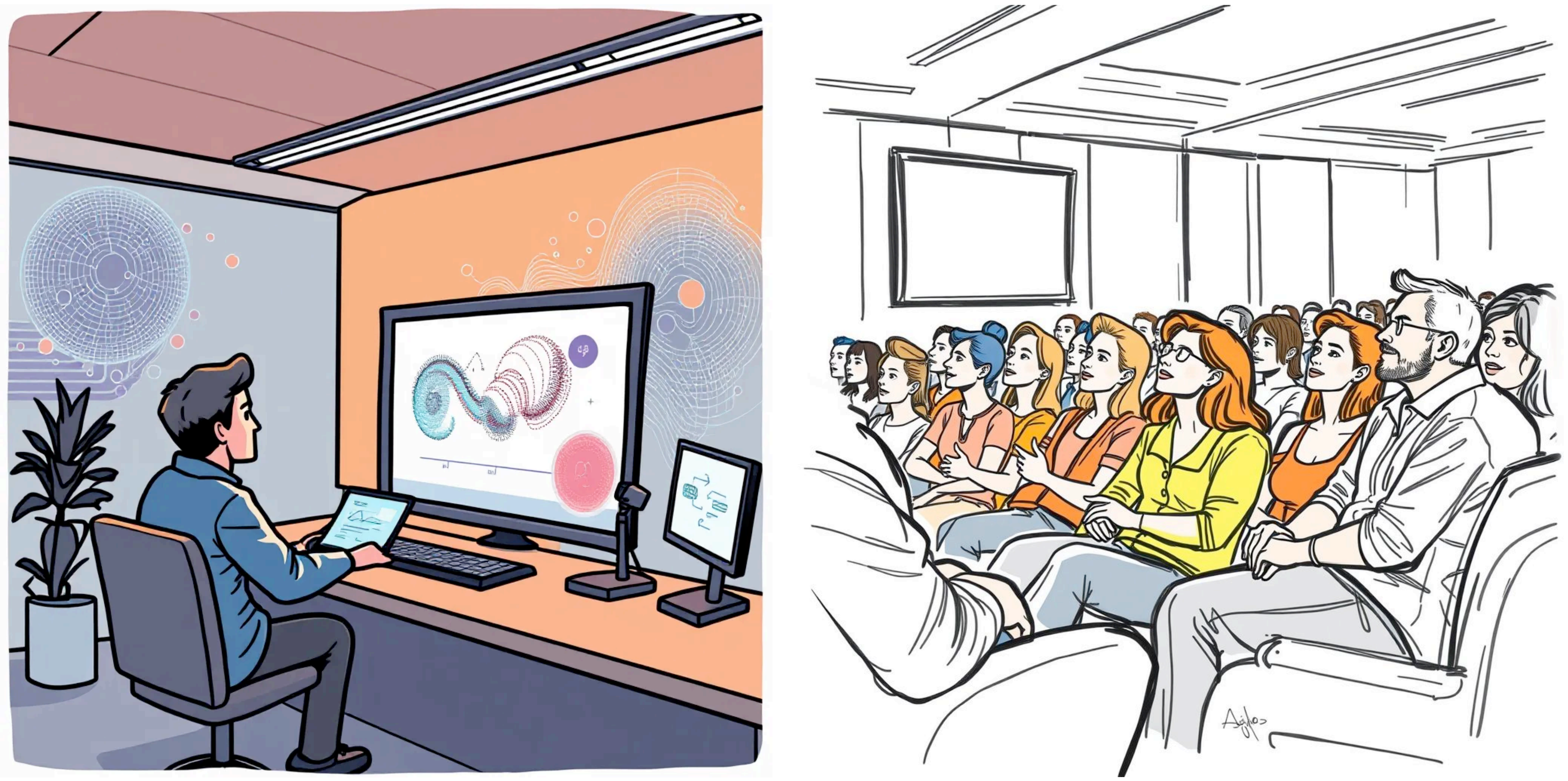
Imagen

Google DeepMind

stability.ai

DALL·E

Adobe Firefly



Images generated using Stable Diffusion

Prompts (conditions):

1. ‘A MSc student in Artificial Intelligence creating a presentation about Diffusion Models, cartoon style’,
2. ‘Audience enjoying a presentation together, sketch style’

How Diffusion Models Work

Forward vs Reverse Process

Forward Process

Left-most real images from SF-XL (*) training set
(*) «Rethinking Visual Geo-localization for Large-Scale Applications» (CVPR 2022)

Reverse Process

How Diffusion Models Work

Forward vs Reverse Process

Forward Process

Left-most real images from SF-XL (*) training set
(*) «Rethinking Visual Geo-localization for Large-Scale Applications» (CVPR 2022)

Reverse Process

How Diffusion Models Work

Forward vs Reverse Process

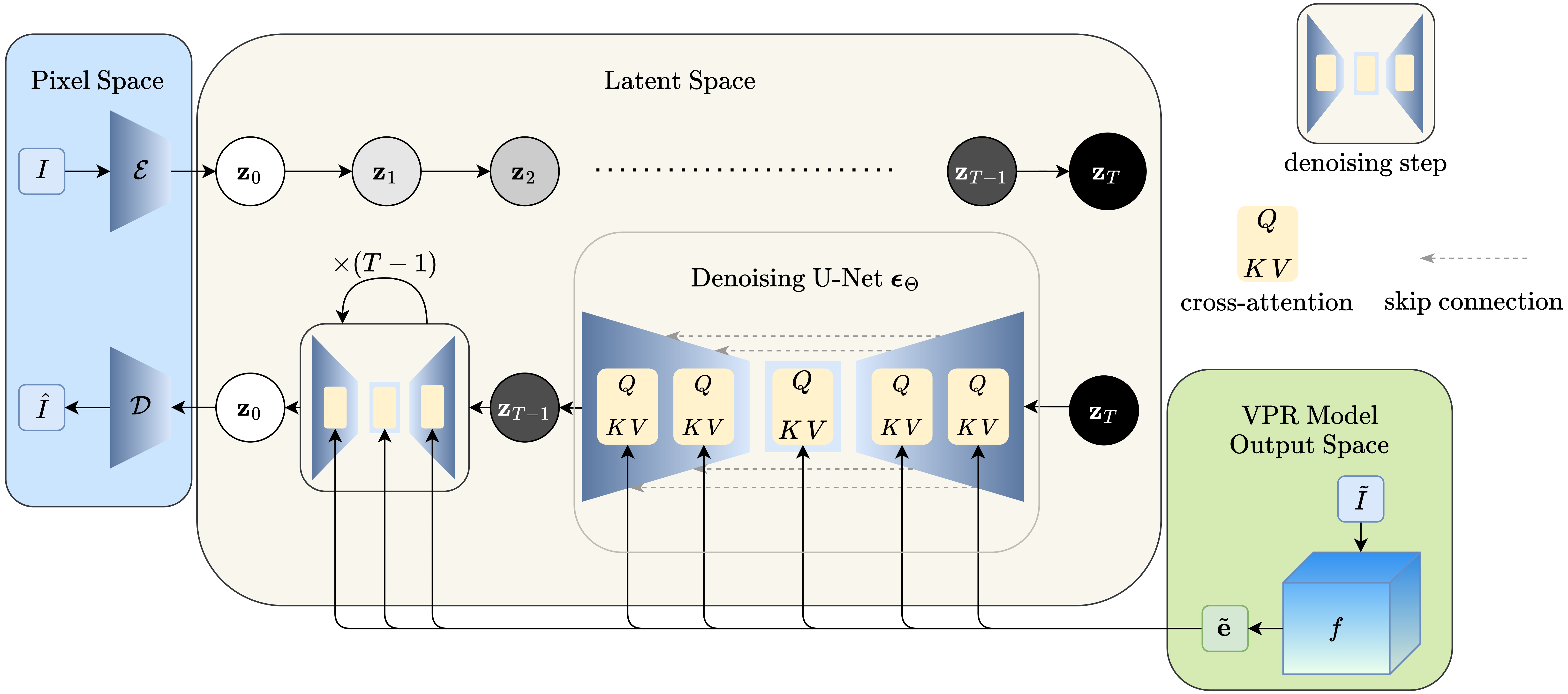
Forward Process

Left-most real images from SF-XL (*) training set
(*) «Rethinking Visual Geo-localization for Large-Scale Applications» (CVPR 2022)

Reverse Process

Latent Diffusion Models (LDMs)

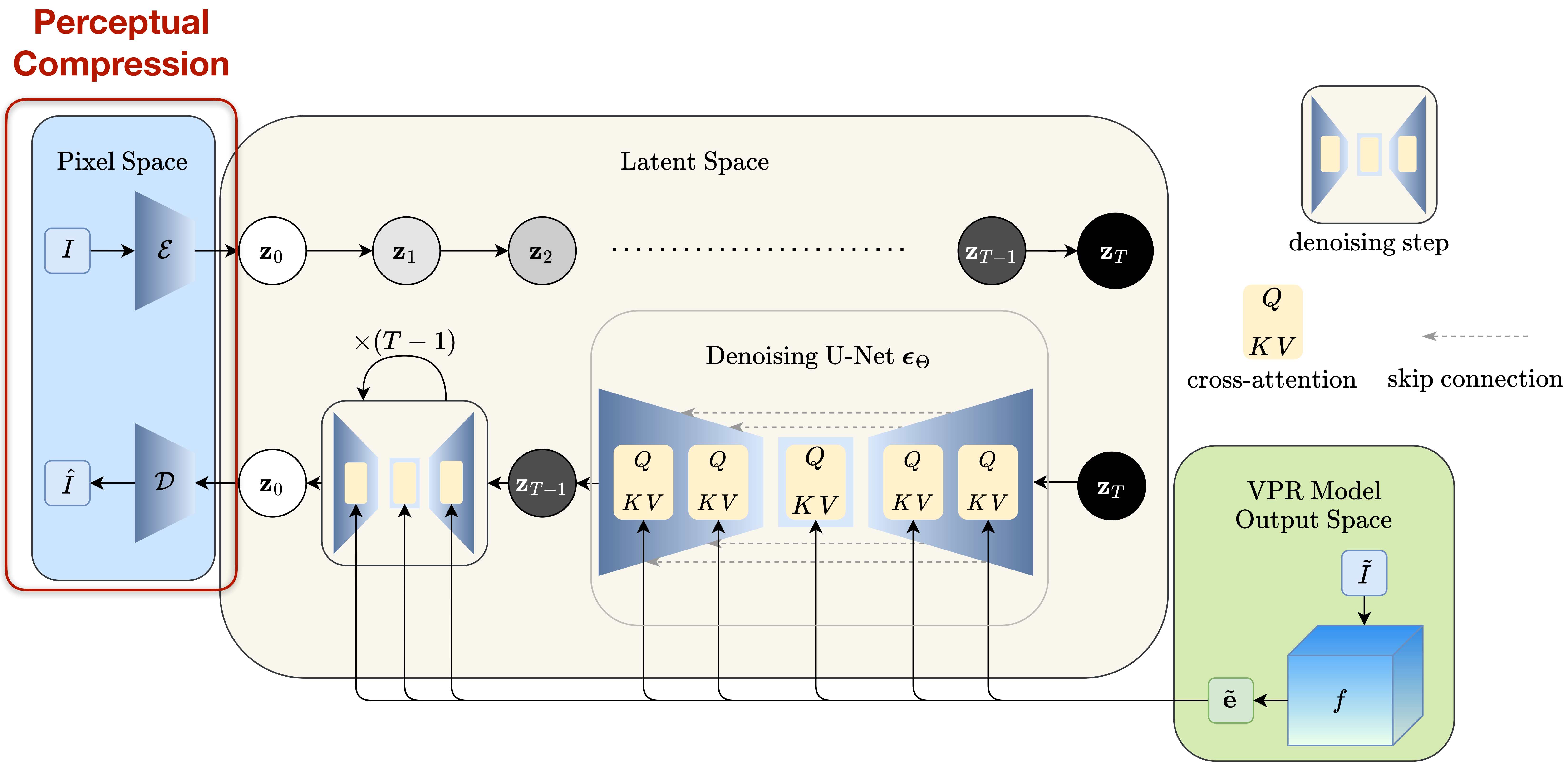
As a lens for VPR



«High-Resolution Image Synthesis with Latent Diffusion Models», (CVPR 2022)

Latent Diffusion Models (LDMs)

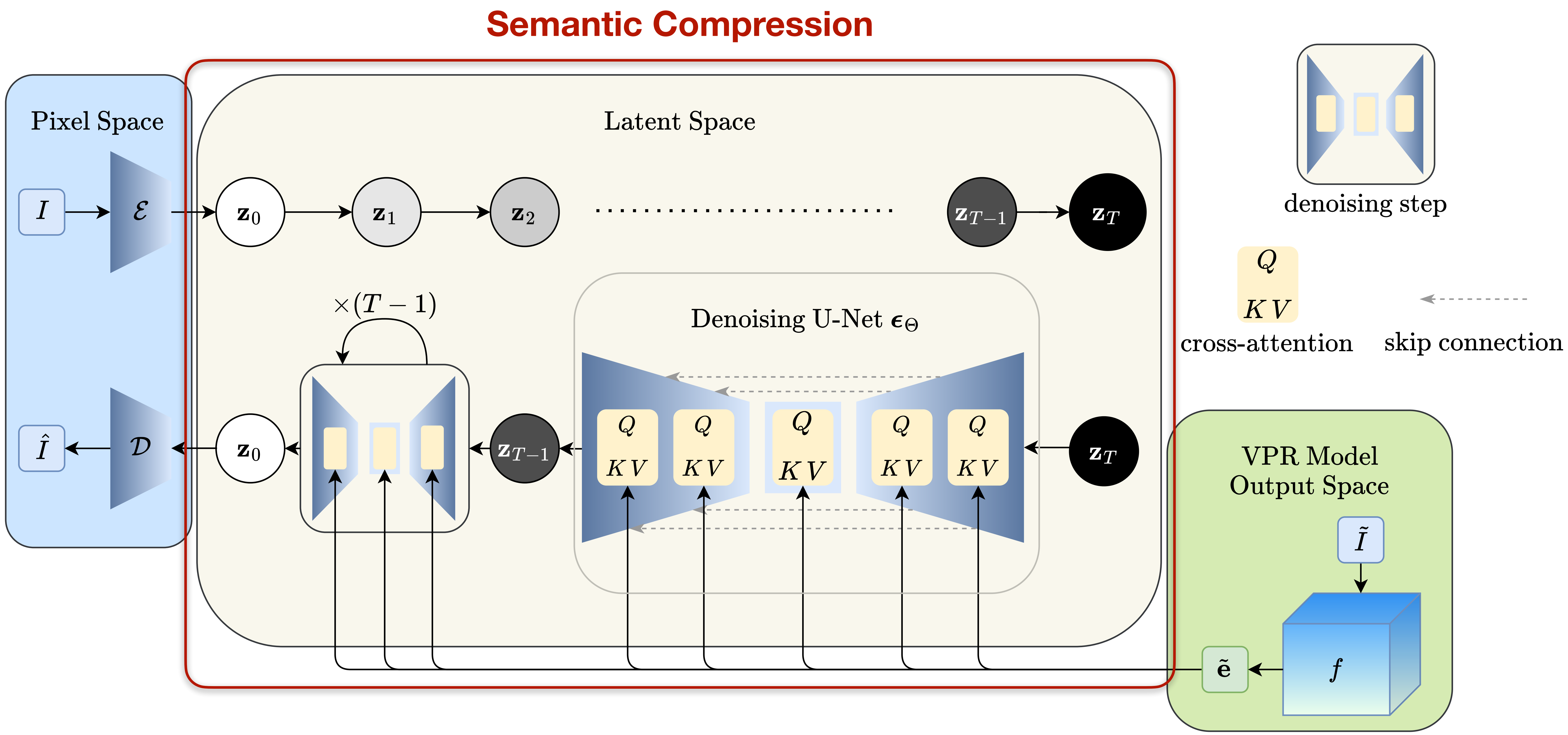
As a lens for VPR



«High-Resolution Image Synthesis with Latent Diffusion Models», (CVPR 2022)

Latent Diffusion Models (LDMs)

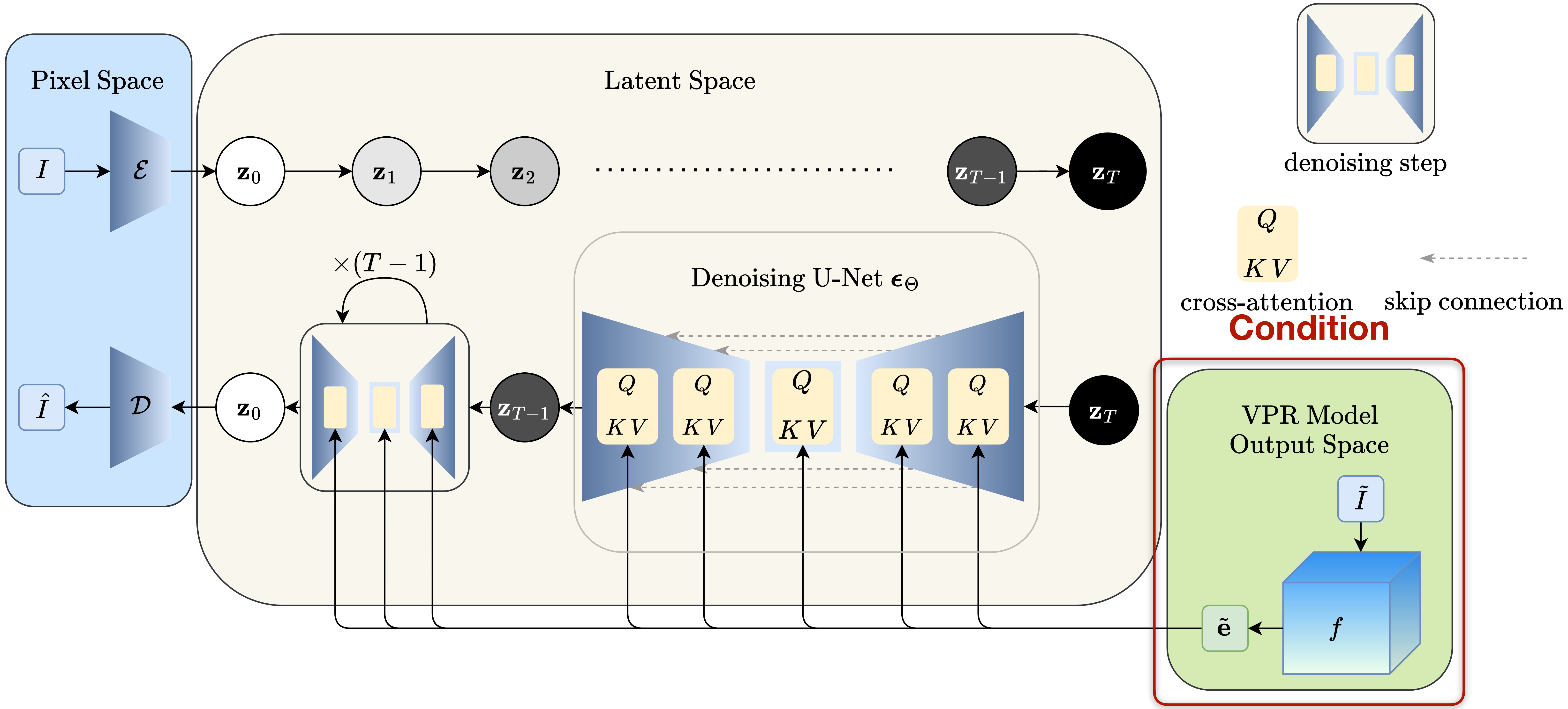
As a lens for VPR



«High-Resolution Image Synthesis with Latent Diffusion Models», (CVPR 2022)

Latent Diffusion Models (LDMs)

As a lens for VPR



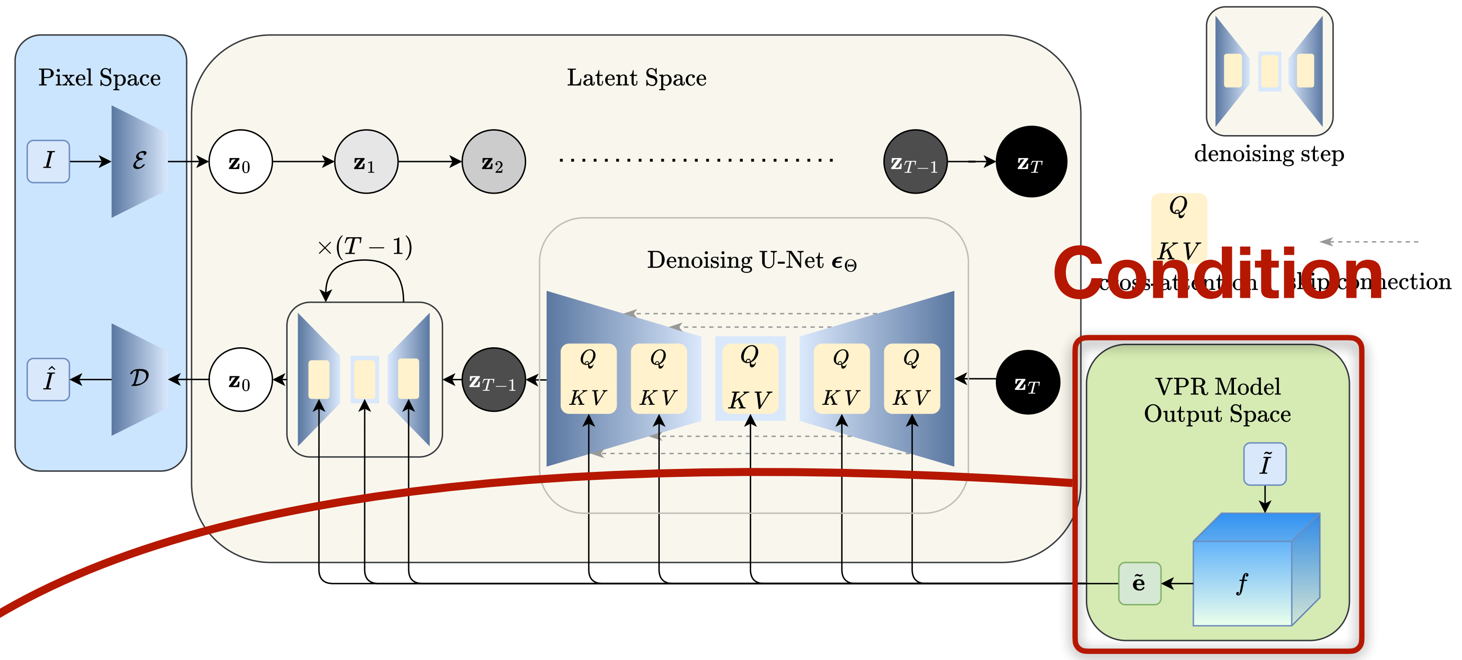
This is the step I acted on

«High-Resolution Image Synthesis with Latent Diffusion Models», (CVPR 2022)

Table of Contents

- Task
- Goals
- Understand Embedding Information
  - Methodology
  - Experiments
- Uncertainty Estimation
  - Methodology
  - Experiments
- Conclusions

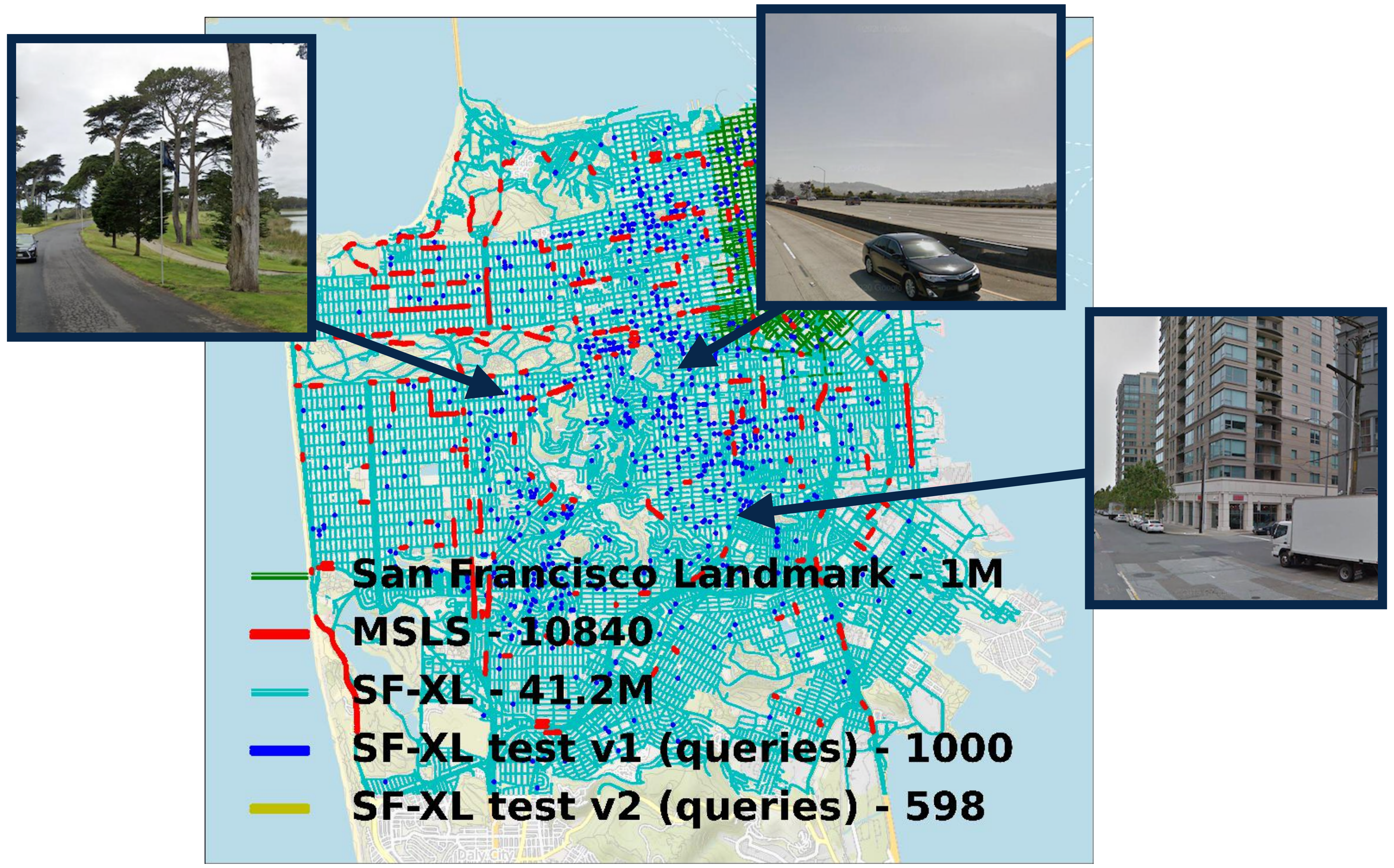
Training LDM models



Method	Backbone	Embedding Dimension d
AP-GeM [14]	ResNet-101 [68]	2048
CliqueMining [13]	DINOv2 [22] (ViT-B/14 [69])	8448
Conv-AP [25]	ResNet-50 [68]	4096
CosPlace [52]	ResNet-50 [68]	32
CosPlace [52]	ResNet-50 [68]	64
CosPlace [52]	ResNet-50 [68]	128
CosPlace [52]	ResNet-50 [68]	512
CosPlace [52]	ResNet-50 [68]	2048
CricaVPR [17]	DINOv2 [22] (ViT-B/14 [69])	10752
EigenPlaces [19]	ResNet-50 [68]	128
EigenPlaces [19]	ResNet-50 [68]	512
EigenPlaces [19]	ResNet-50 [68]	2048
MixVPR [18]	ResNet-50 [68]	4096
NetVLAD [15]	VGG-16 [70]	4096
SALAD [51]	DINOv2 [22] (ViT-B/14 [69])	8448
SFRS [71]	VGG-16 [70]	4096

Table 6.1: Visual Place Recognition models used for conditioning.

Training data
(48 million images processed with duplicates)



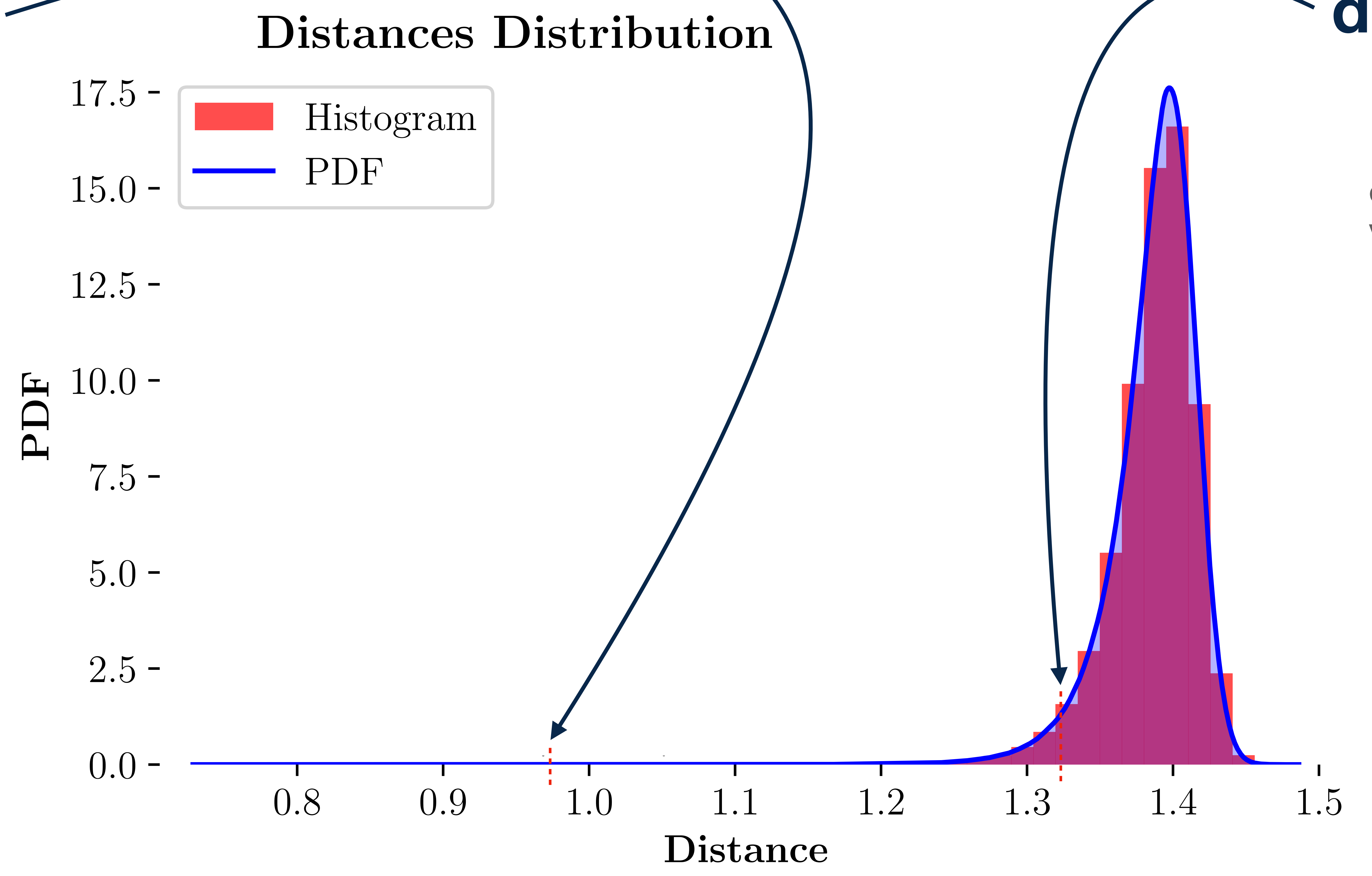
Set	Latitude	# Images
Training	37.71 – 37.81	5,350,506
Validation	37.70	256,914

Table 6.2: Training and validation sets derived from the training split of the SF-XL [52] dataset used for training LDM models.

Are the generated images really closer to conditioning embedding?

d()

L_2 distance of embeddings extracted from a real image in the validation set and a generated image by conditioning on the real embedding using a VPR model



d()

L_2 distance of two embeddings extracted from real images in the validation set using a VPR model

Do hypotheses about VPR models really hold?

- Do VPR models ignore irrelevant details, such as cars and pedestrians, when predicting a location?
- Do VPR models ignore certain contextual information, like the time of day or weather?

How to use the framework I propose?

- Can I compare different VPR models?
- Can I inspect the VPR model's output space in more detail?

Ever-present elements vs. Transient elements

Real image from MSLS [^], all other images are generated using the LDM model conditioned on **MixVPR (*)**.
[^] «Mapillary street-level sequences: A dataset for lifelong place recognition», (CVPR 2020)

(*) «MixVPR: Feature Mixing for Visual Place Recognition», (WACV 2023)

Does another model encode the same information?

Real image from MSLS [^], all other images are generated using the LDM model conditioned on **SALAD (*)**.
[^] «Mapillary street-level sequences: A dataset for lifelong place recognition», (CVPR 2020)

(*) «Optimal Transport Aggregation for Visual Place Recognition», (CVPR 2024)

Night is irrelevant

Real image from SVOX Night [^], all other images are generated using the LDM model conditioned on **SALAD (*)**.
[^] «Adaptive-attentive geolocalization from few queries: A hybrid approach», (WACV 2021)

(*) «Optimal Transport Aggregation for Visual Place Recognition», (CVPR 2024)

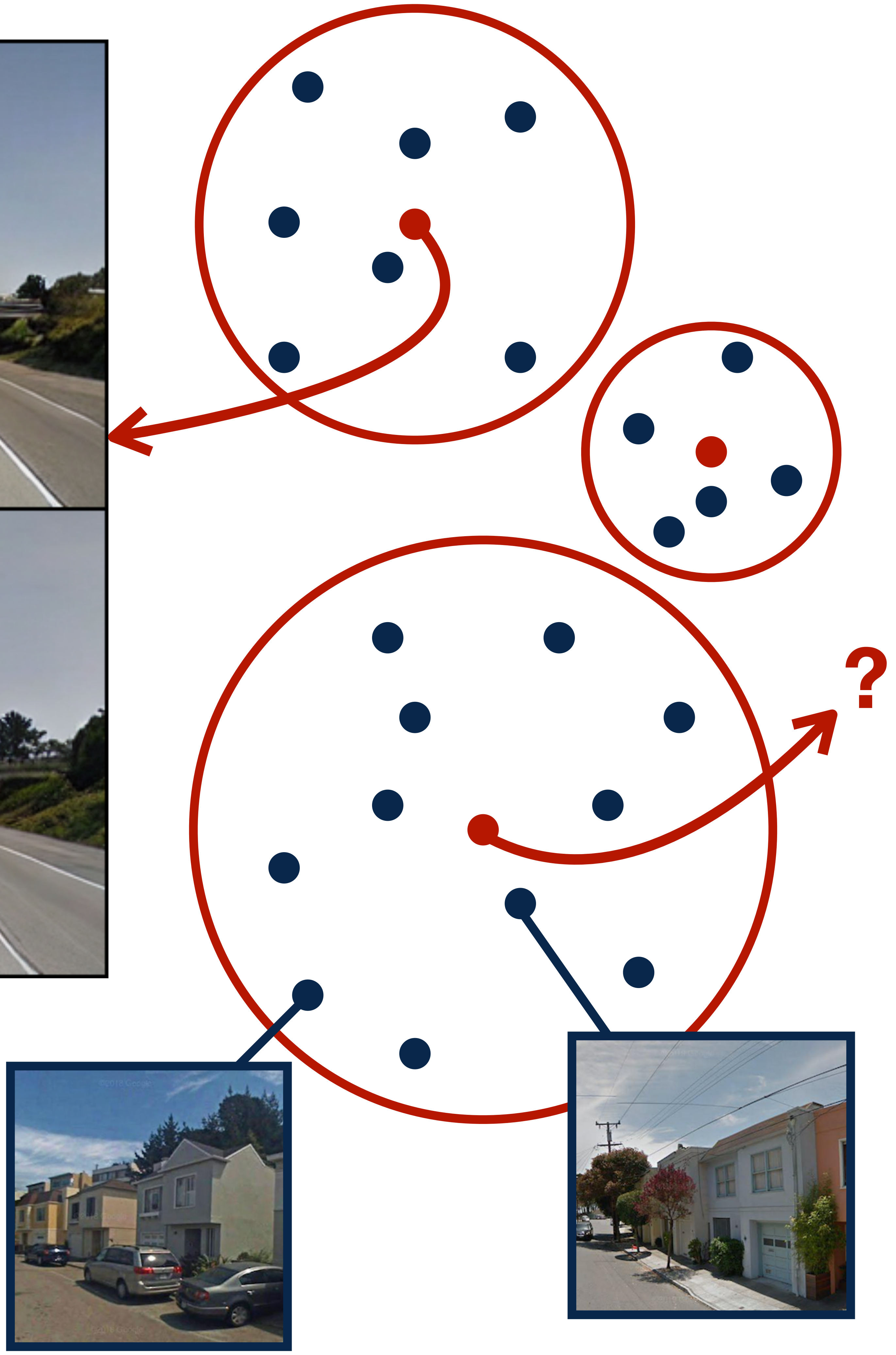
And surprisingly even the snow gets abstracted away

Real image from Nordland [^], all other images are generated using the LDM model conditioned on **SALAD (*)**.
[^] «Are we there yet? challenging SeqSLAM on a 3000 km journey across all four seasons.», (ICRA 2013)

(*) «Optimal Transport Aggregation for Visual Place Recognition», (CVPR 2024)

Sample and show hypothetical embeddings

All images are generated using the LDM model conditioned on a centroid of the validation set for **SALAD (*)**.

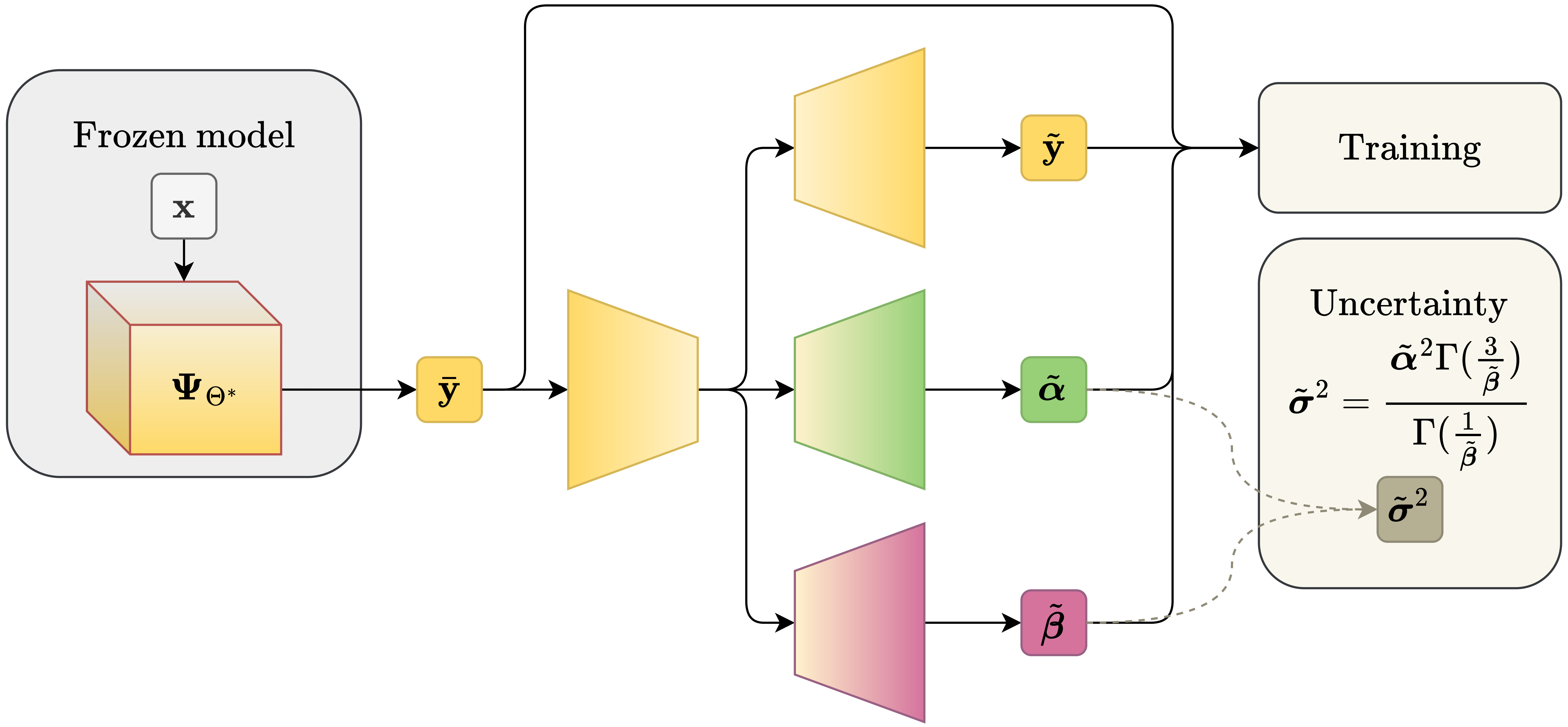


(*) «Optimal Transport Aggregation for Visual Place Recognition», (CVPR 2024)

Table of Contents

- Task
- Goals
- Understand Embedding Information
  - Methodology
  - Experiments
- **Uncertainty Estimation**
  - **Methodology**
  - Experiments
- Conclusions

BayesCap
Post-hoc Uncertainty Estimation

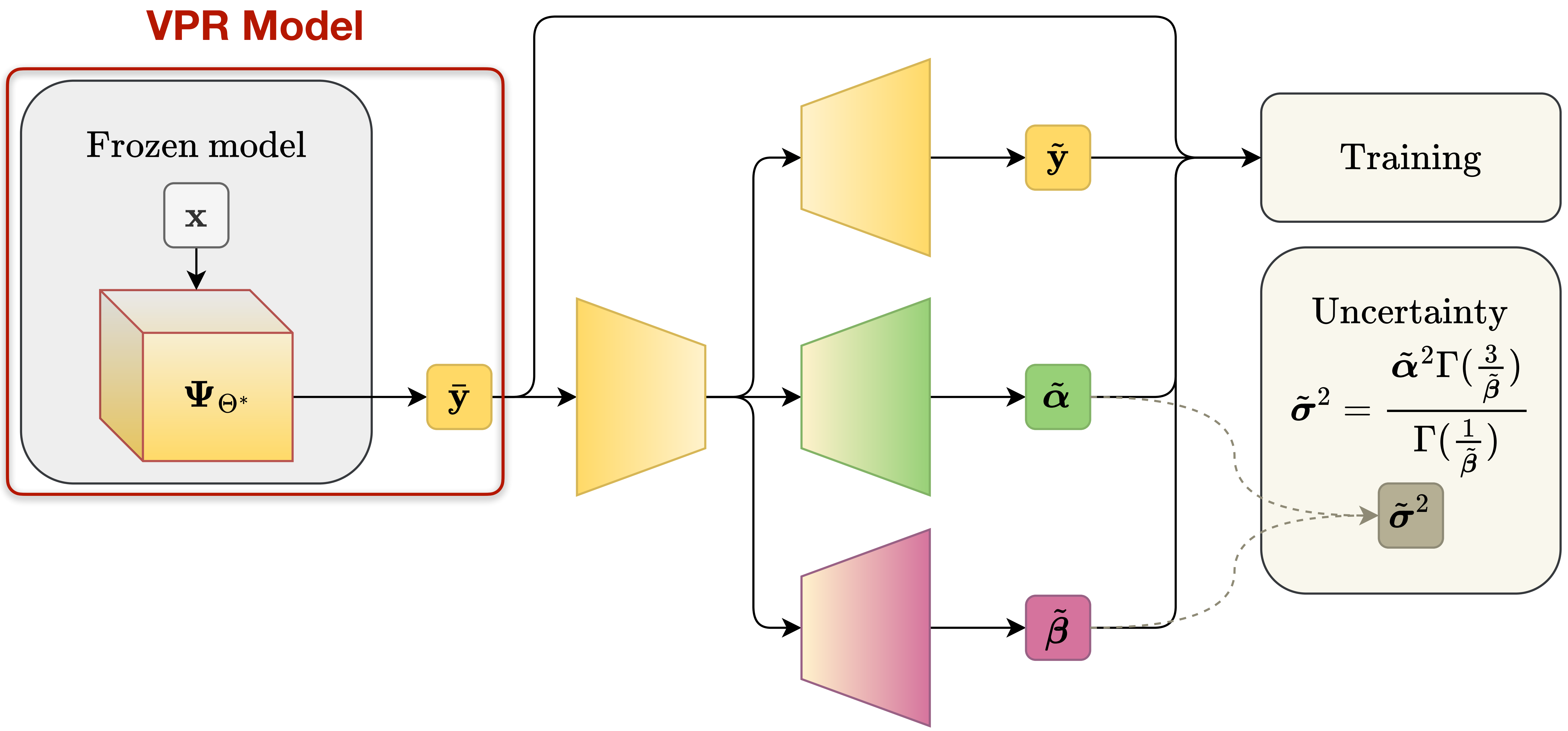


$$\mathscr{L}(\Phi) = \lambda_1 \sum_{i=1}^{N} |\tilde{\mathbf{y}}_i - \bar{\mathbf{y}}_i| + \lambda_2 \sum_{i=1}^{N} \left(\frac{|\tilde{\mathbf{y}}_i - \mathbf{y}_i|}{\tilde{\boldsymbol{\alpha}}_i} \right)^{\tilde{\boldsymbol{\beta}}_i} - \log \frac{\tilde{\boldsymbol{\beta}}_i}{\tilde{\boldsymbol{\alpha}}_i} + \log \Gamma(\frac{1}{\tilde{\boldsymbol{\beta}}_i})$$

«BayesCap: Bayesian Identity Cap for Calibrated Uncertainty in Frozen Neural Networks», (ECCV 2022)

BayesCap

Post-hoc Uncertainty Estimation

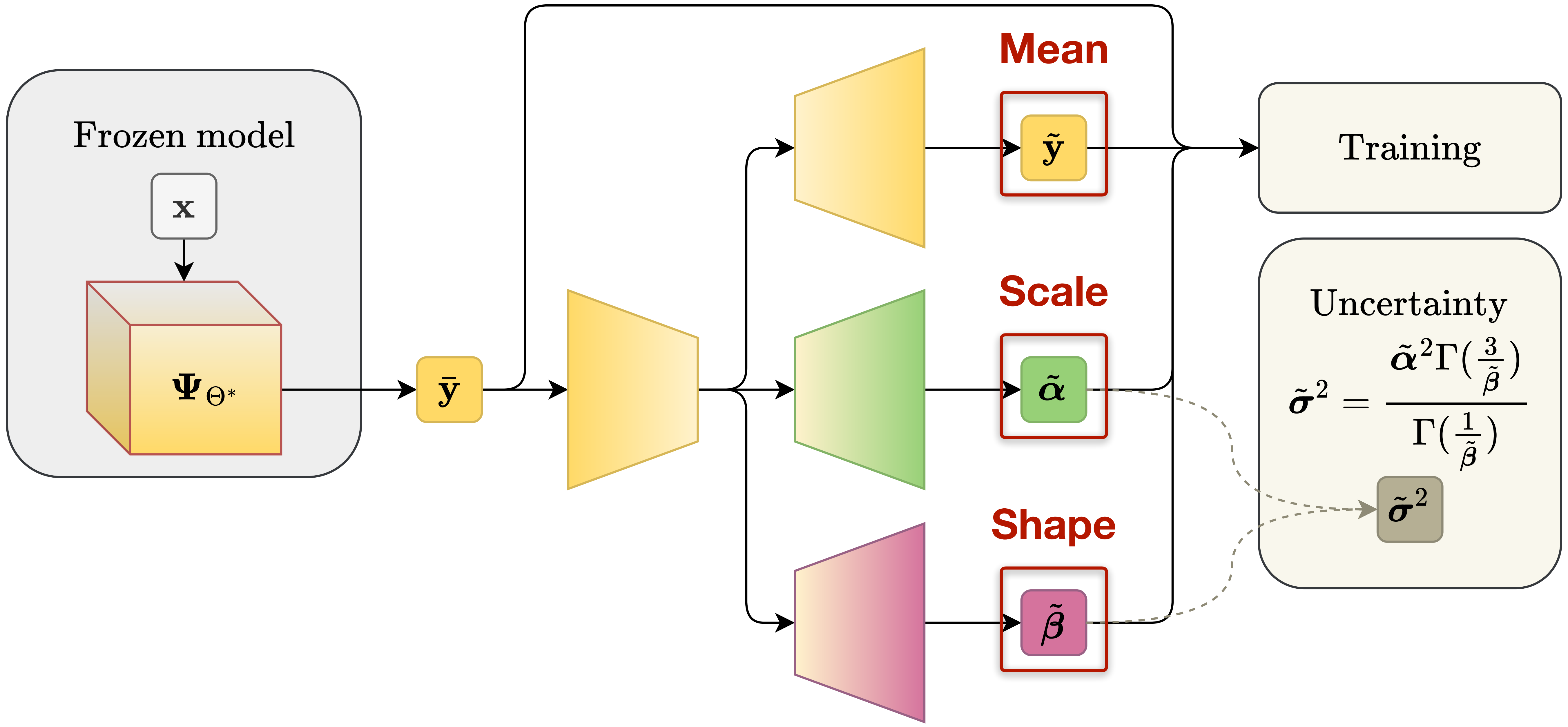


$$\mathscr{L}(\Phi) = \lambda_1 \sum_{i=1}^{N} |\tilde{\mathbf{y}}_i - \bar{\mathbf{y}}_i| + \lambda_2 \sum_{i=1}^{N} \left(\frac{|\tilde{\mathbf{y}}_i - \mathbf{y}_i|}{\tilde{\boldsymbol{\alpha}}_i} \right)^{\tilde{\beta}_i} - \log \frac{\tilde{\boldsymbol{\beta}}_i}{\tilde{\boldsymbol{\alpha}}_i} + \log \Gamma(\frac{1}{\tilde{\boldsymbol{\beta}}_i})$$

«BayesCap: Bayesian Identity Cap for Calibrated Uncertainty in Frozen Neural Networks», (ECCV 2022)

BayesCap

Post-hoc Uncertainty Estimation



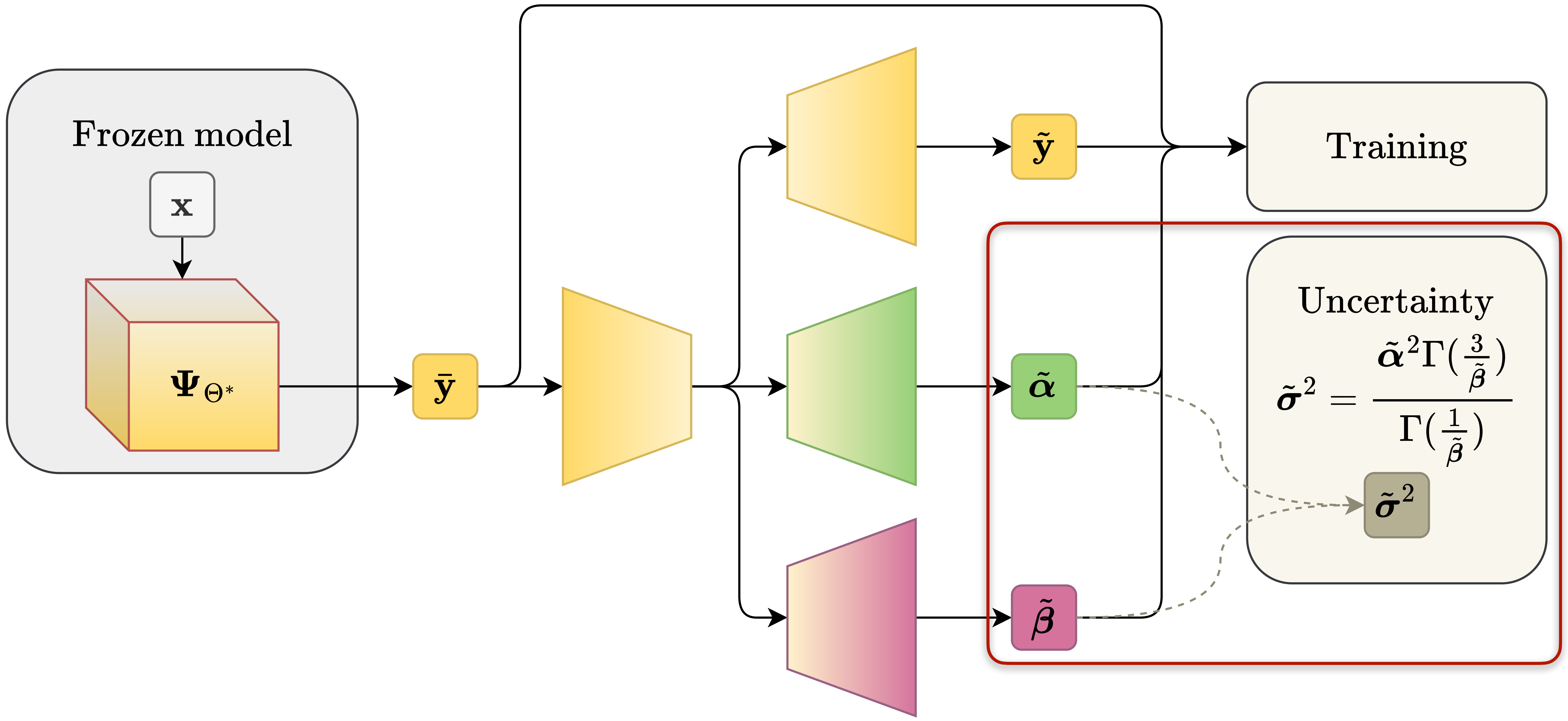
$$\frac{\tilde{\beta}}{2\tilde{\alpha}\Gamma(\frac{1}{\tilde{\beta}})} e^{-\left(\frac{|\tilde{\mathbf{y}}-\mathbf{y}|}{\tilde{\alpha}}\right)^{\tilde{\beta}}}$$

$$\mathscr{L}(\Phi) = \lambda_1 \sum_{i=1}^{N} |\tilde{\mathbf{y}}_i - \bar{\mathbf{y}}_i| + \lambda_2 \sum_{i=1}^{N} \left(\frac{|\tilde{\mathbf{y}}_i - \mathbf{y}_i|}{\tilde{\alpha}_i}\right)^{\tilde{\beta}_i} - \log \frac{\tilde{\beta}_i}{\tilde{\alpha}_i} + \log \Gamma(\frac{1}{\tilde{\beta}_i})$$

«BayesCap: Bayesian Identity Cap for Calibrated Uncertainty in Frozen Neural Networks», (ECCV 2022)

BayesCap

Post-hoc Uncertainty Estimation

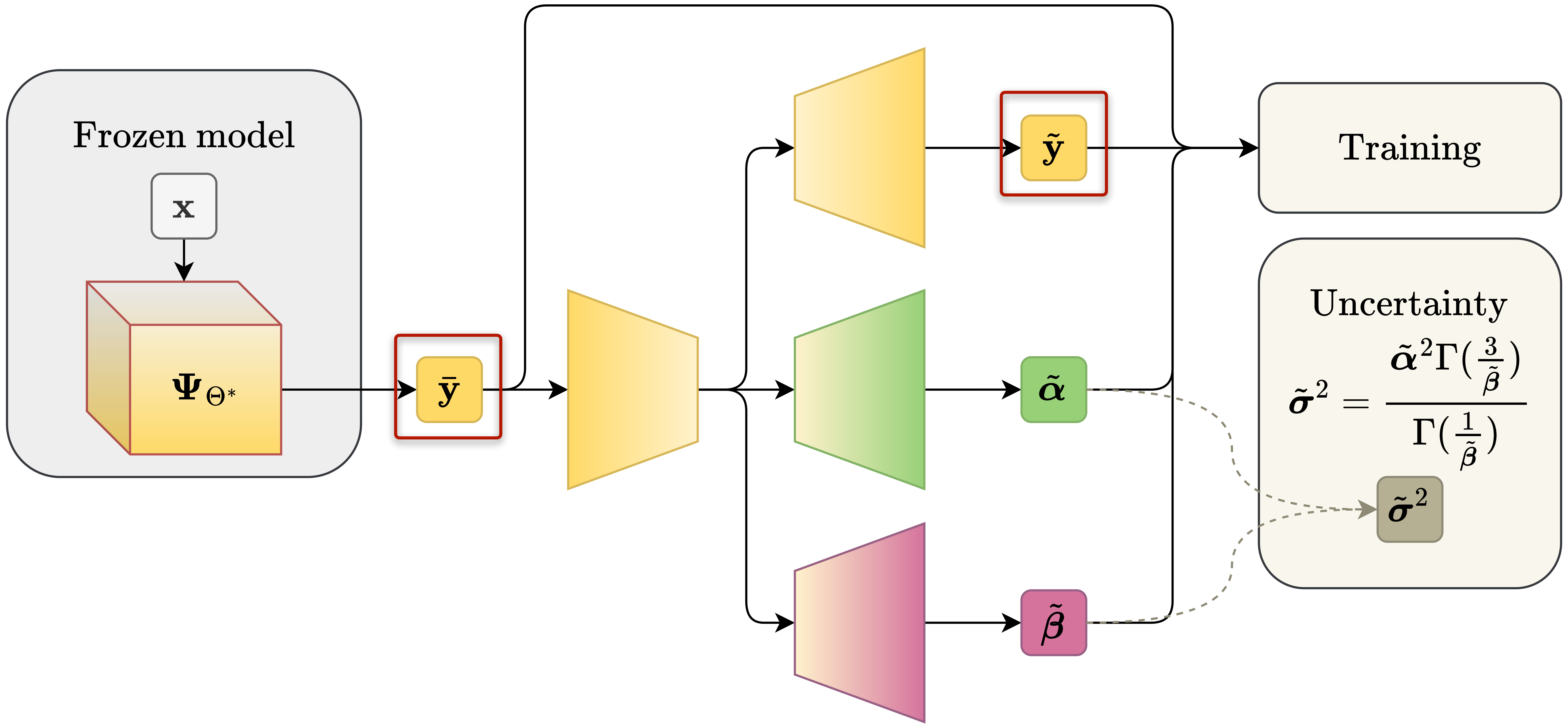


$$\mathscr{L}(\Phi) = \lambda_1 \sum_{i=1}^{N} |\tilde{\mathbf{y}}_i - \bar{\mathbf{y}}_i| + \lambda_2 \sum_{i=1}^{N} \left(\frac{|\tilde{\mathbf{y}}_i - \mathbf{y}_i|}{\tilde{\boldsymbol{\alpha}}_i} \right)^{\tilde{\beta}_i} - \log \frac{\tilde{\boldsymbol{\beta}}_i}{\tilde{\boldsymbol{\alpha}}_i} + \log \Gamma(\frac{1}{\tilde{\boldsymbol{\beta}}_i})$$

«BayesCap: Bayesian Identity Cap for Calibrated Uncertainty in Frozen Neural Networks», (ECCV 2022)

BayesCap

Post-hoc Uncertainty Estimation



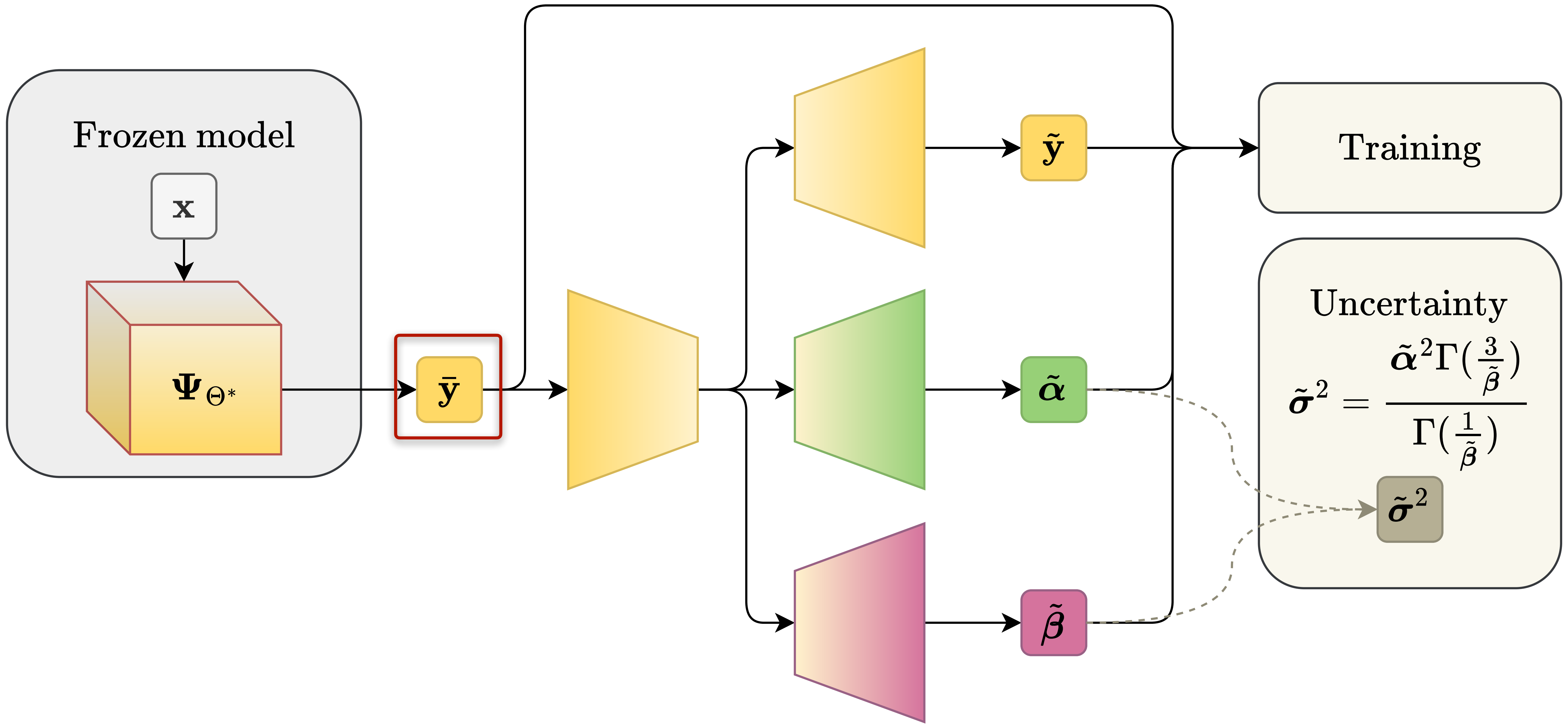
Reconstruction Loss

$$\mathscr{L}(\Phi) = \lambda_1 \boxed{\sum_{i=1}^{N} |\tilde{\mathbf{y}}_i - \bar{\mathbf{y}}_i|} + \lambda_2 \sum_{i=1}^{N} \left(\frac{|\tilde{\mathbf{y}}_i - \mathbf{y}_i|}{\tilde{\boldsymbol{\alpha}}_i} \right)^{\tilde{\boldsymbol{\beta}}_i} - \log \frac{\tilde{\boldsymbol{\beta}}_i}{\tilde{\boldsymbol{\alpha}}_i} + \log \Gamma(\frac{1}{\tilde{\boldsymbol{\beta}}_i})$$

«BayesCap: Bayesian Identity Cap for Calibrated Uncertainty in Frozen Neural Networks», (ECCV 2022)

BayesCap

Post-hoc Uncertainty Estimation



How to obtain ground-truth embedding $\mathbf{y}$?

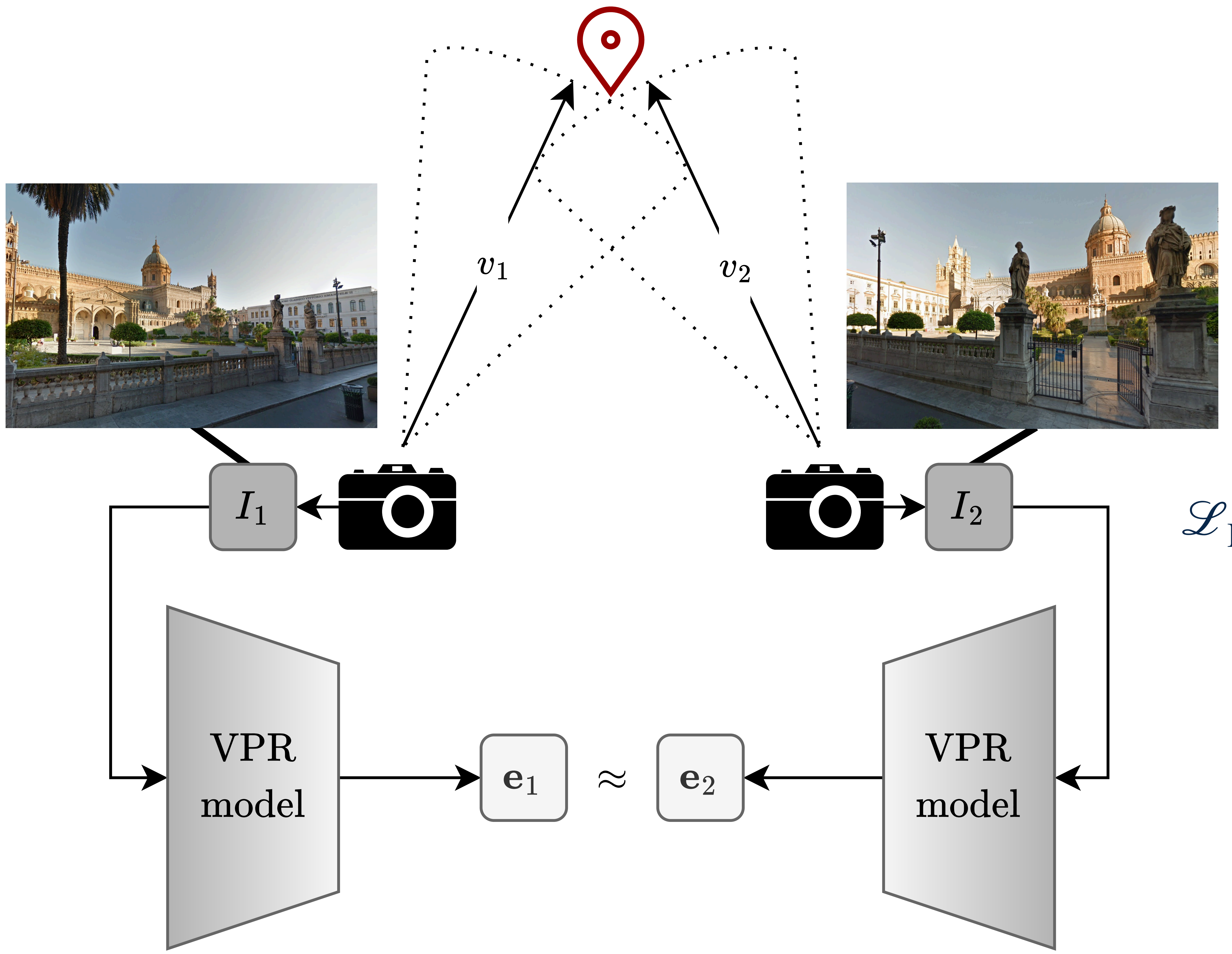
Maximum Likelihood Estimation

$$\mathscr{L}(\Phi) = \lambda_1 \sum_{i=1}^{N} |\tilde{\mathbf{y}}_i - \bar{\mathbf{y}}_i| + \lambda_2 \boxed{\sum_{i=1}^{N} \left(\frac{|\tilde{\mathbf{y}}_i - \mathbf{y}_i|}{\tilde{\boldsymbol{\alpha}}_i}\right)^{\tilde{\boldsymbol{\beta}}_i} - \log \frac{\tilde{\boldsymbol{\beta}}_i}{\tilde{\boldsymbol{\alpha}}_i} + \log \Gamma(\frac{1}{\tilde{\boldsymbol{\beta}}_i})}$$

«BayesCap: Bayesian Identity Cap for Calibrated Uncertainty in Frozen Neural Networks», (ECCV 2022)

Viewpoint Invariance

Desired property for VPR models

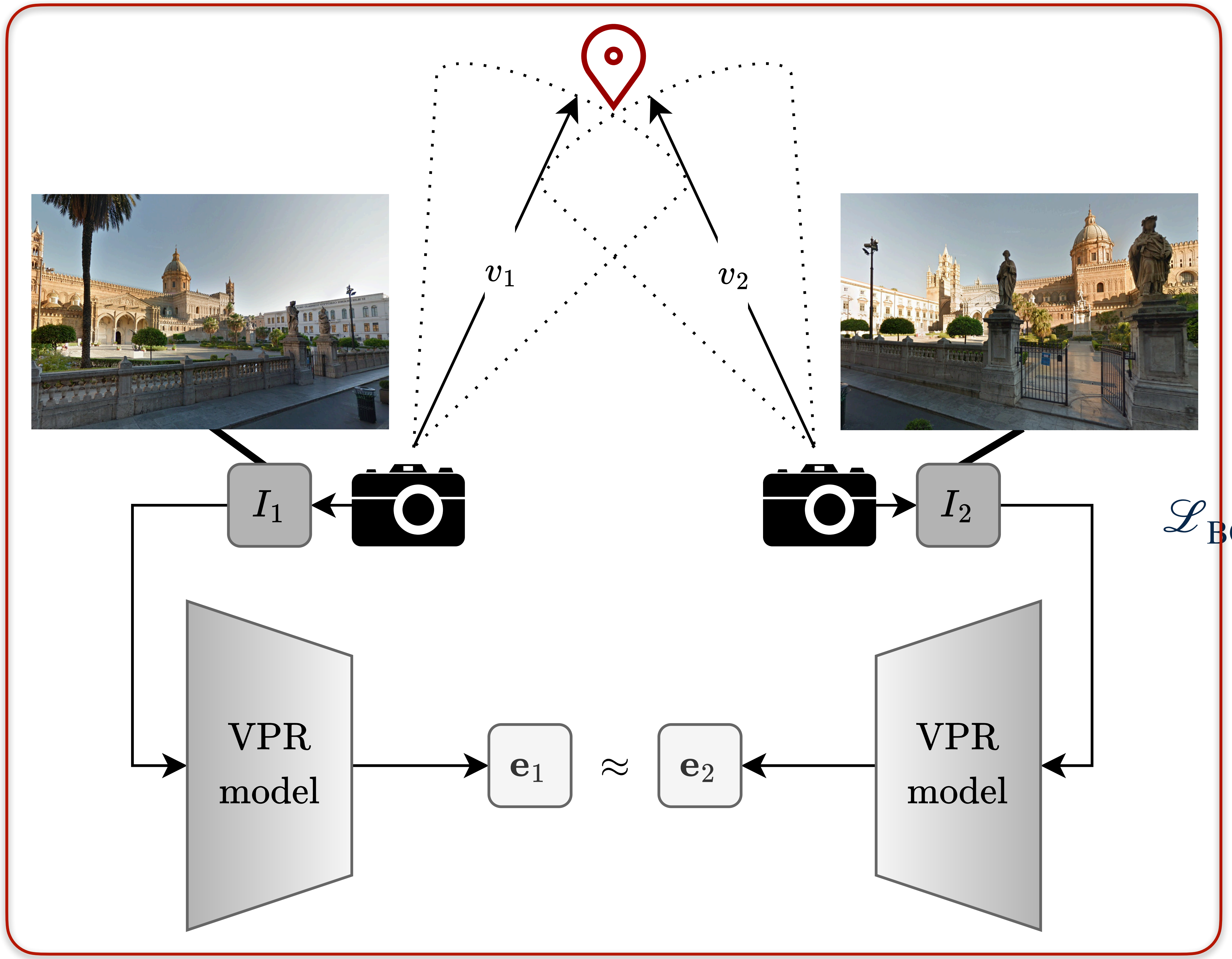


$$\mathcal{L}_{\mathrm{BCC}} = \frac{\lambda_1}{2}\sum_{i=1}^{N}\left|\tilde{\mathbf{y}}_i^{(1)} - \mathbf{e}_i^{(1)}\right| + \frac{\lambda_2}{2}\sum_{i=1}^{N}\left(\frac{\left|\tilde{\mathbf{y}}_i^{(1)} - \mathbf{e}_i^{(2)}\right|}{\tilde{\boldsymbol{\alpha}}_i^{(1)}}\right)^{\tilde{\boldsymbol{\beta}}_i^{(1)}} - \log\frac{\tilde{\boldsymbol{\beta}}_i^{(1)}}{\tilde{\boldsymbol{\alpha}}_i^{(1)}} + \log\Gamma\left(\frac{1}{\tilde{\boldsymbol{\beta}}_i^{(1)}}\right)$$

$$+\frac{\lambda_1}{2}\sum_{i=1}^{N}\left|\tilde{\mathbf{y}}_i^{(2)} - \mathbf{e}_i^{(2)}\right| + \frac{\lambda_2}{2}\sum_{i=1}^{N}\left(\frac{\left|\tilde{\mathbf{y}}_i^{(2)} - \mathbf{e}_i^{(1)}\right|}{\tilde{\boldsymbol{\alpha}}_i^{(2)}}\right)^{\tilde{\boldsymbol{\beta}}_i^{(2)}} - \log\frac{\tilde{\boldsymbol{\beta}}_i^{(2)}}{\tilde{\boldsymbol{\alpha}}_i^{(2)}} + \log\Gamma\left(\frac{1}{\tilde{\boldsymbol{\beta}}_i^{(2)}}\right)$$

Viewpoint Invariance

Desired **property** for VPR models

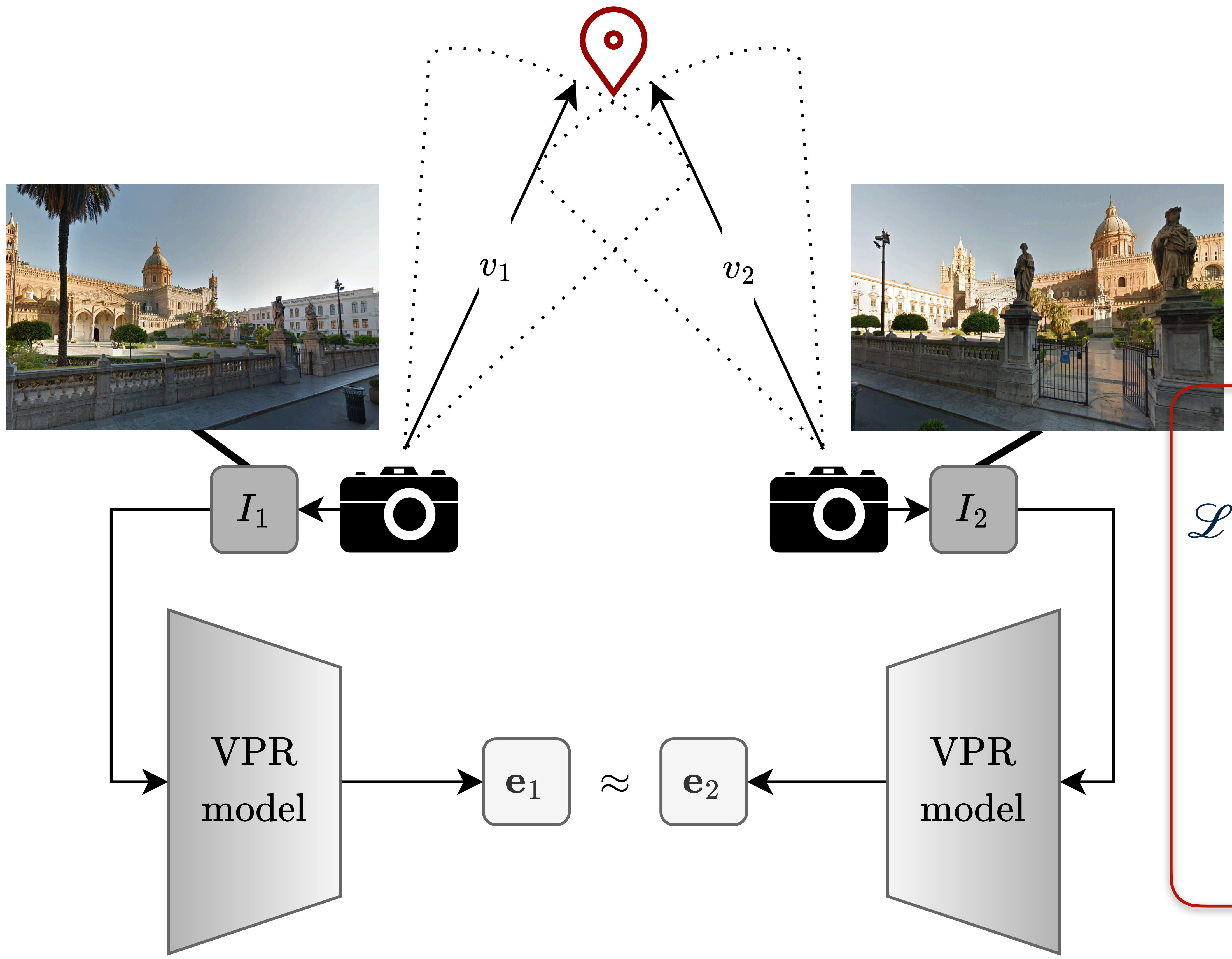


$$\mathscr{L}_{\mathrm{BCC}} = \frac{\lambda_1}{2}\sum_{i=1}^{N}\left|\tilde{\mathbf{y}}_i^{(1)} - \mathbf{e}_i^{(1)}\right| + \frac{\lambda_2}{2}\sum_{i=1}^{N}\left(\frac{\left|\tilde{\mathbf{y}}_i^{(1)} - \mathbf{e}_i^{(2)}\right|}{\tilde{\boldsymbol{\alpha}}_i^{(1)}}\right)^{\tilde{\boldsymbol{\beta}}_i^{(1)}} - \log\frac{\tilde{\boldsymbol{\beta}}_i^{(1)}}{\tilde{\boldsymbol{\alpha}}_i^{(1)}} + \log\Gamma\left(\frac{1}{\tilde{\boldsymbol{\beta}}_i^{(1)}}\right)$$

$$+\frac{\lambda_1}{2}\sum_{i=1}^{N}\left|\tilde{\mathbf{y}}_i^{(2)} - \mathbf{e}_i^{(2)}\right| + \frac{\lambda_2}{2}\sum_{i=1}^{N}\left(\frac{\left|\tilde{\mathbf{y}}_i^{(2)} - \mathbf{e}_i^{(1)}\right|}{\tilde{\boldsymbol{\alpha}}_i^{(2)}}\right)^{\tilde{\boldsymbol{\beta}}_i^{(2)}} - \log\frac{\tilde{\boldsymbol{\beta}}_i^{(2)}}{\tilde{\boldsymbol{\alpha}}_i^{(2)}} + \log\Gamma\left(\frac{1}{\tilde{\boldsymbol{\beta}}_i^{(2)}}\right)$$

Viewpoint Invariance

Desired **property** for VPR models



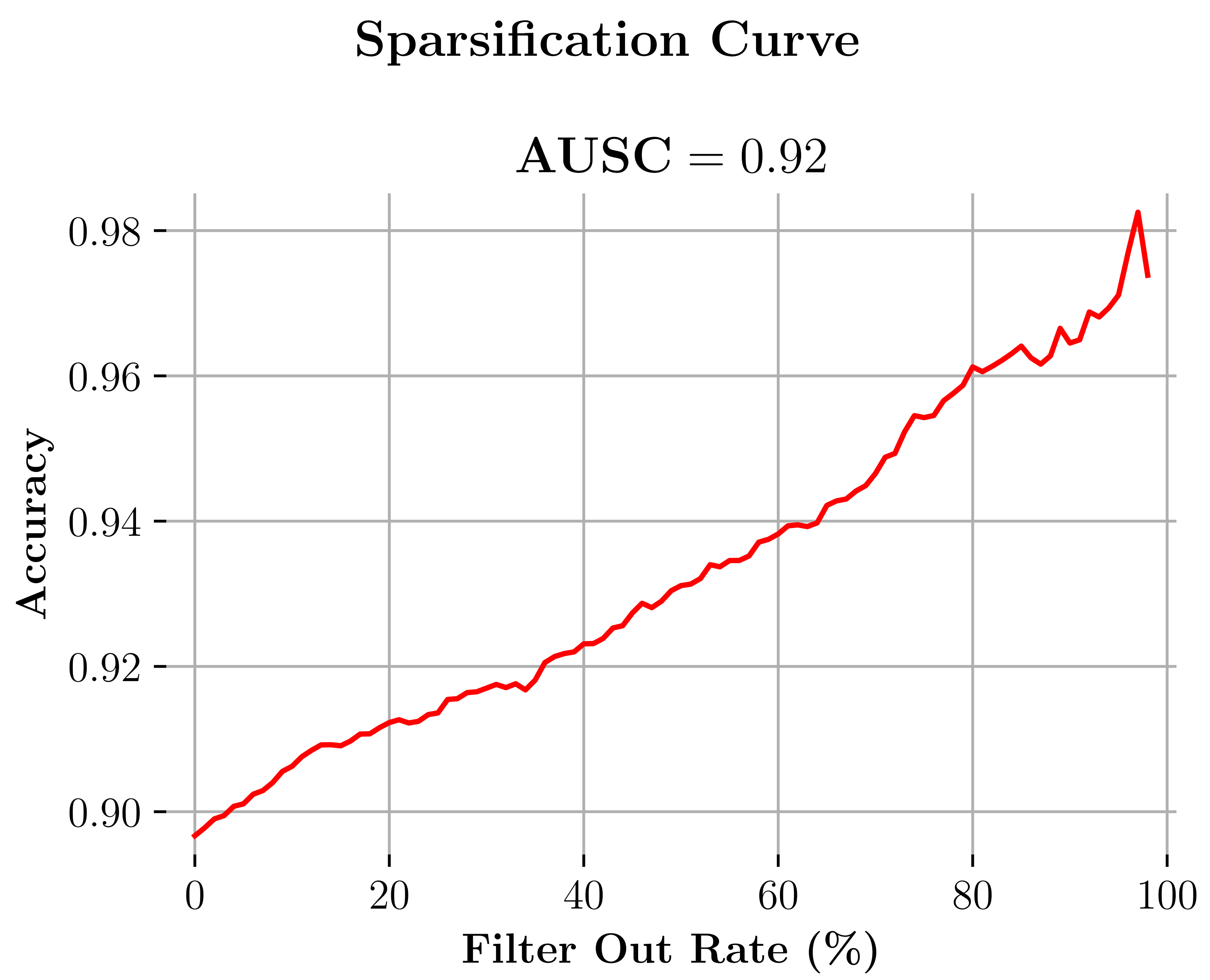
one embedding as ground truth and the other one as embedding to reconstruct, and vice versa

$$\begin{aligned}\mathcal{L}_{\mathrm{BCC}} = {} & \frac{\lambda_1}{2}\sum_{i=1}^{N}\left|\tilde{\mathbf{y}}_i^{(1)} - \mathbf{e}_i^{(1)}\right| + \frac{\lambda_2}{2}\sum_{i=1}^{N}\left(\frac{\left|\tilde{\mathbf{y}}_i^{(1)} - \mathbf{e}_i^{(2)}\right|}{\tilde{\boldsymbol{\alpha}}_i^{(1)}}\right)^{\tilde{\boldsymbol{\beta}}_i^{(1)}} - \log\frac{\tilde{\boldsymbol{\beta}}_i^{(1)}}{\tilde{\boldsymbol{\alpha}}_i^{(1)}} + \log\Gamma\left(\frac{1}{\tilde{\boldsymbol{\beta}}_i^{(1)}}\right) \\ & + \frac{\lambda_1}{2}\sum_{i=1}^{N}\left|\tilde{\mathbf{y}}_i^{(2)} - \mathbf{e}_i^{(2)}\right| + \frac{\lambda_2}{2}\sum_{i=1}^{N}\left(\frac{\left|\tilde{\mathbf{y}}_i^{(2)} - \mathbf{e}_i^{(1)}\right|}{\tilde{\boldsymbol{\alpha}}_i^{(2)}}\right)^{\tilde{\boldsymbol{\beta}}_i^{(2)}} - \log\frac{\tilde{\boldsymbol{\beta}}_i^{(2)}}{\tilde{\boldsymbol{\alpha}}_i^{(2)}} + \log\Gamma\left(\frac{1}{\tilde{\boldsymbol{\beta}}_i^{(2)}}\right)\end{aligned}$$

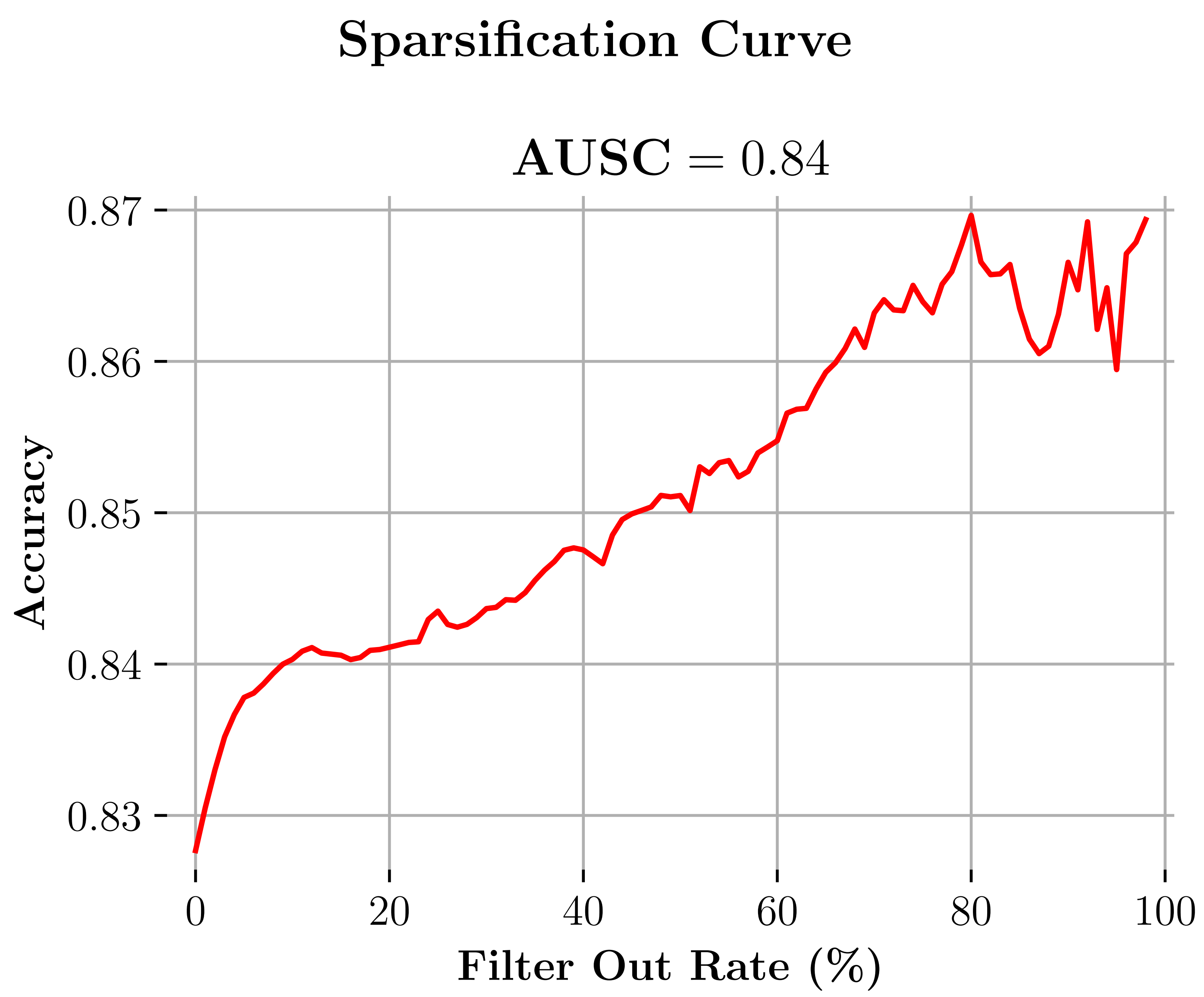
Table of Contents

- Task
- Goals
- Understand Embedding Information
  - Methodology
  - Experiments
- Uncertainty Estimation
  - Methodology
  - Experiments
- Conclusions

Uncertainty Estimation Experiments

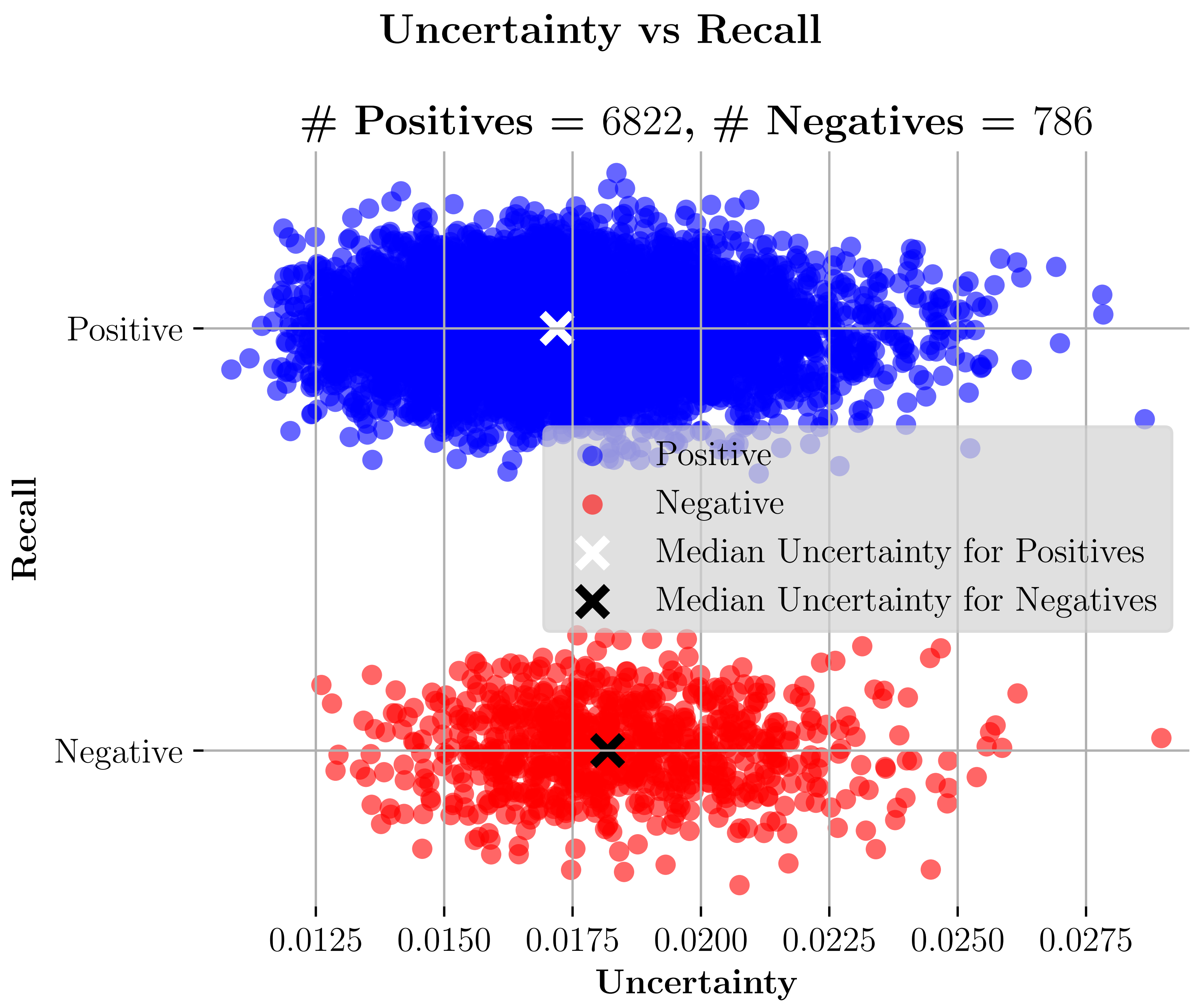


Pitts30k val (*)
(*) «NetVLAD: CNN architecture for weakly supervised place recognition» (CVPR 2016)

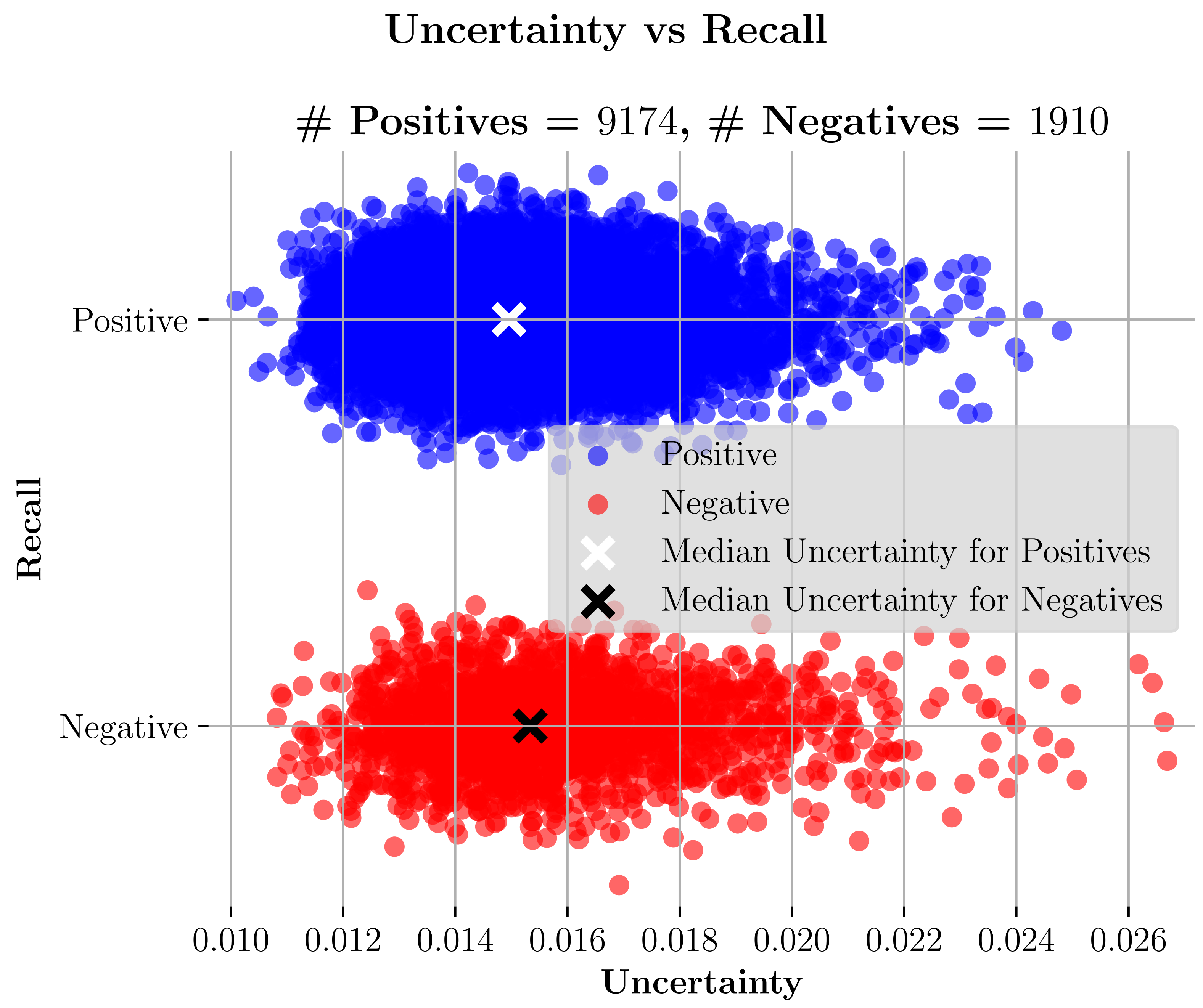


MSLS val (*)
(*) «Mapillary Street-Level Sequences: A Dataset for Lifelong Place Recognition» (CVPR 2020)

Uncertainty Estimation Experiments



Pitts30k val (*)
(*) «NetVLAD: CNN architecture for weakly supervised place recognition» (CVPR 2016)



MSLS val (*)
(*) «Mapillary Street-Level Sequences: A Dataset for Lifelong Place Recognition» (CVPR 2020)

Table of Contents

- Task
- Goals
- Understand Embedding Information
  - Methodology
  - Experiments
- Uncertainty Estimation
  - Methodology
  - Experiments
- Conclusions

Conclusions

Embedding Information Inspection

- Capture relationships between image content and embeddings
- Tool to visualize embeddings
  - Even hypothetical embeddings (e.g., centroids)

Uncertainty Estimation

- Provide useful aggregated uncertainty scores, according to the evaluation metrics
- Need for better separation between positive and negative queries

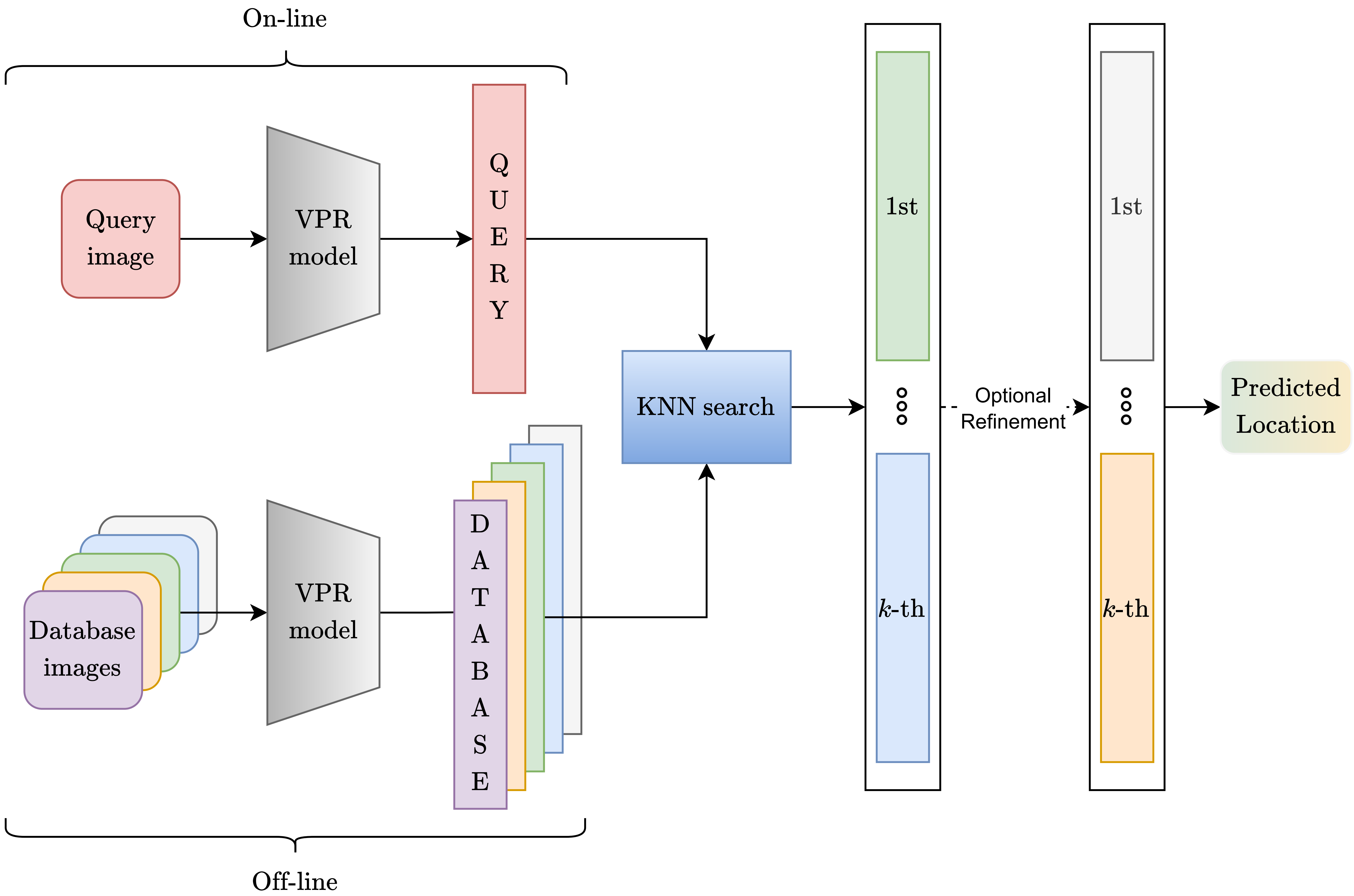
An extension of my thesis work on uncertainty estimation has been accepted as an article to the "Image Matching: Local Features and Beyond" workshop at CVPR 2025

Thank you for your attention!

"I am indeed amazed when I consider how weak my mind is and how prone to error"
Rene Descartes

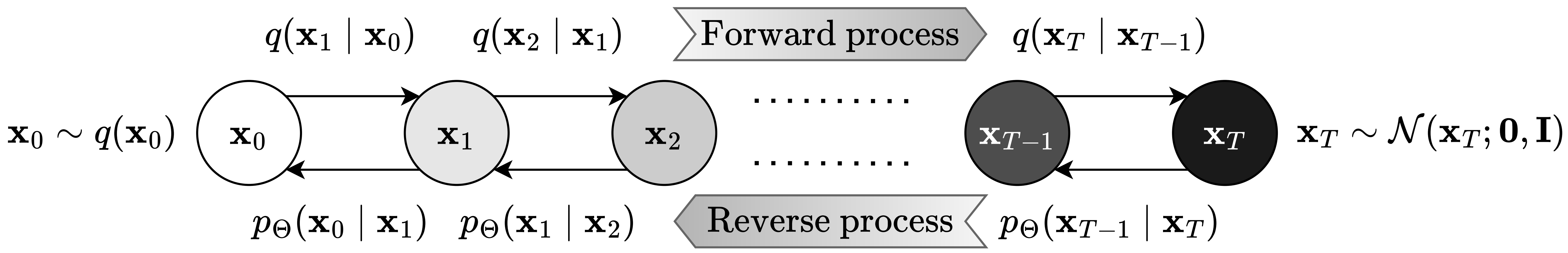
Visual Place Recognition

Pipeline



Diffusion Models

Generative AI Models



«Denoising Diffusion Probabilistic Models», (NeurIPS 2020)

But are the generated images really closer to conditioning embedding?

Statistic	L_1	L_2
Maximum value	46.90	1.30
Minimum value	21.11	0.58
Mean value	33.54	0.93
Standard Deviation	4.57	0.13
$s=1$ CFG [39]	30.97	0.86
$s=2$ CFG [39]	28.54	0.79

But are the generated images really closer to conditioning embedding?

CosPlace (*) classes in the validation set

Statistic	L_1	L_2
Maximum value	46.90	1.30
Minimum value	21.11	0.58
Mean value	33.54	0.93
Standard Deviation	4.57	0.13
$s = 1$ CFG [39]	30.97	0.86
$s = 2$ CFG [39]	28.54	0.79

(*) «Rethinking Visual Geo-localization for Large-Scale Application», (CVPR 2022)

But are the generated images really closer to conditioning embedding?

LDM model

Statistic	L_1	L_2
Maximum value	46.90	1.30
Minimum value	21.11	0.58
Mean value	33.54	0.93
Standard Deviation	4.57	0.13
$s = 1$ CFG [39]	30.97	0.86
$s = 2$ CFG [39]	28.54	0.79

But are the generated images really closer to conditioning embedding?

	Statistic	L_1	L_2
CosPlace (*) classes in the validation set	Maximum value	46.90	1.30
	Minimum value	21.11	0.58
	Mean value	33.54	0.93
	Standard Deviation	4.57	0.13
LDM model	$s = 1$ CFG [39]	30.97	0.86
	$s = 2$ CFG [39]	28.54	0.79

(*) «Rethinking Visual Geo-localization for Large-Scale Application», (CVPR 2022)