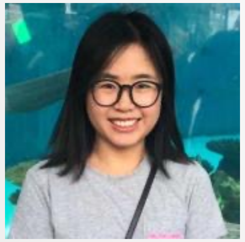
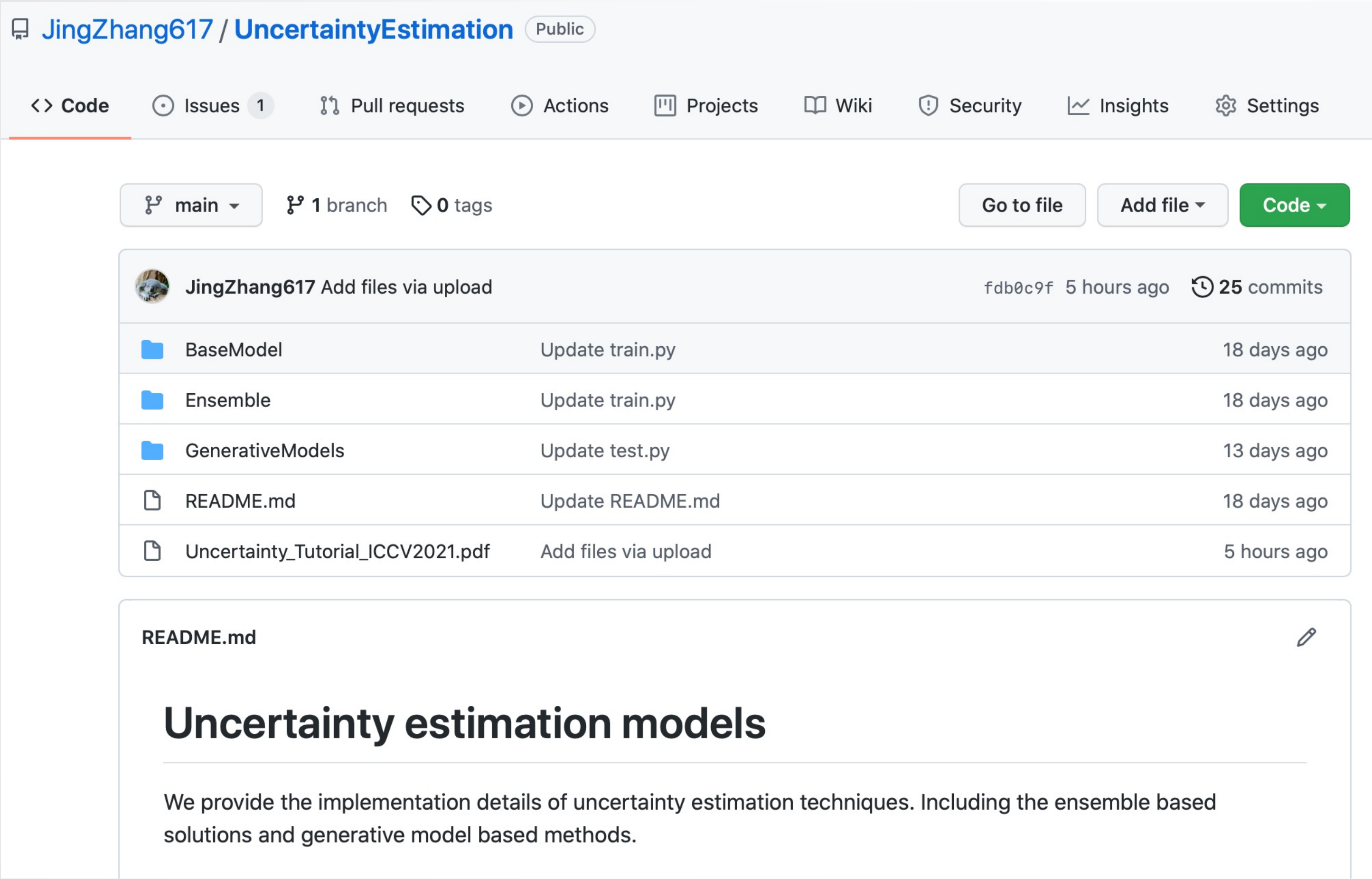
2021 ICCV OCTOBER 11-17 VIRTUAL

Uncertainty Estimation for Dense Prediction Tasks

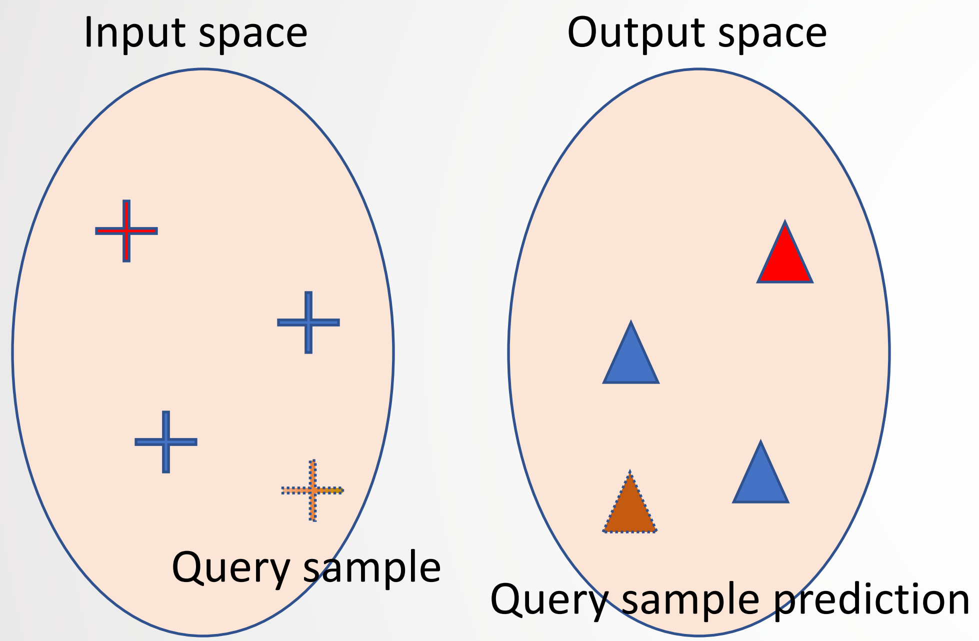
Jing Zhang, Yuchao Dai, Deng-Ping Fan, Nick Barnes, Peyman Moghadam, Christian Walder, Mehrtash Harandi

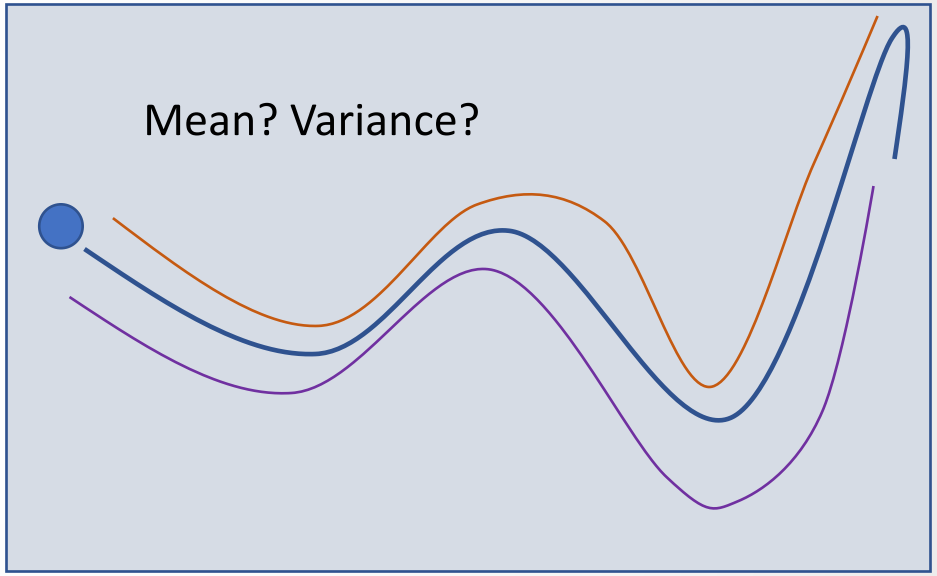
      



Code and tutorial material are available

Uncertainty is inherent within Machine Learning





Easy sample? Hard sample?

Outline

- Motivation
- Background
- Aleatoric uncertainty and Epistemic Uncertainty
- Uncertainty Approximation
    - ✓Ensemble solutions
    - ✓Generative model solutions
    - ✓Bayesian latent variable model solutions
- Experiments
- Discussion and Conclusion

Motivation

- Distribution estimation vs Point estimation
  - ➢Classification: prediction with **confidence**
  - ➢Regression: prediction with **variance**

Uncertainty: a mechanism to understand model limitations

Motivation

- Camouflaged object detection
- Salient object detection
- Monocular depth estimation

Classification

Regression

Motivation

- Which one is salient?

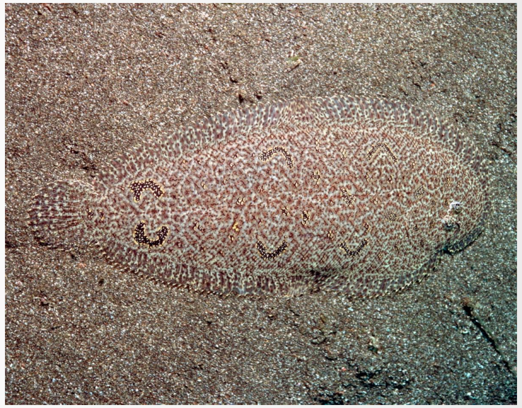
Motivation

- Which one is salient?

Motivation

- Where is the camouflaged object

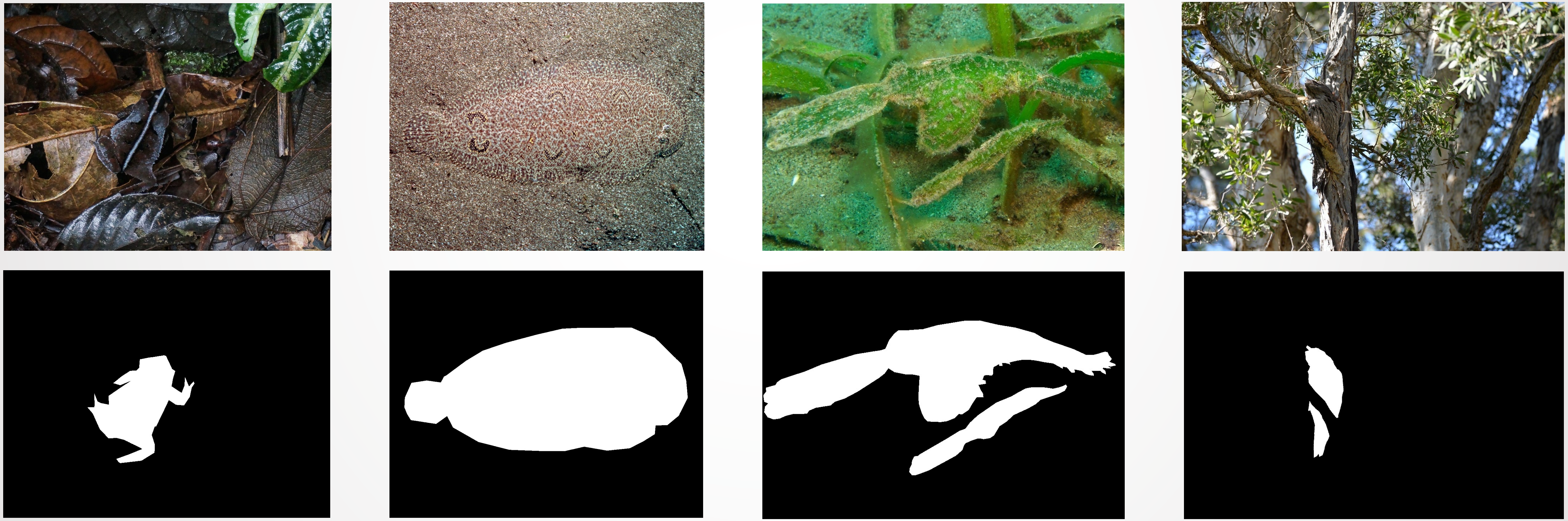


Camouflaged Object Segmentation dataset: Anabranch Network for Camouflaged Object Segmentation, Le et al, CVIU, 2019

Motivation

- Where is the camouflaged object



Motivation – Depth estimation from RGB

- What's the exact distance?

Motivation – Depth estimation from RGB

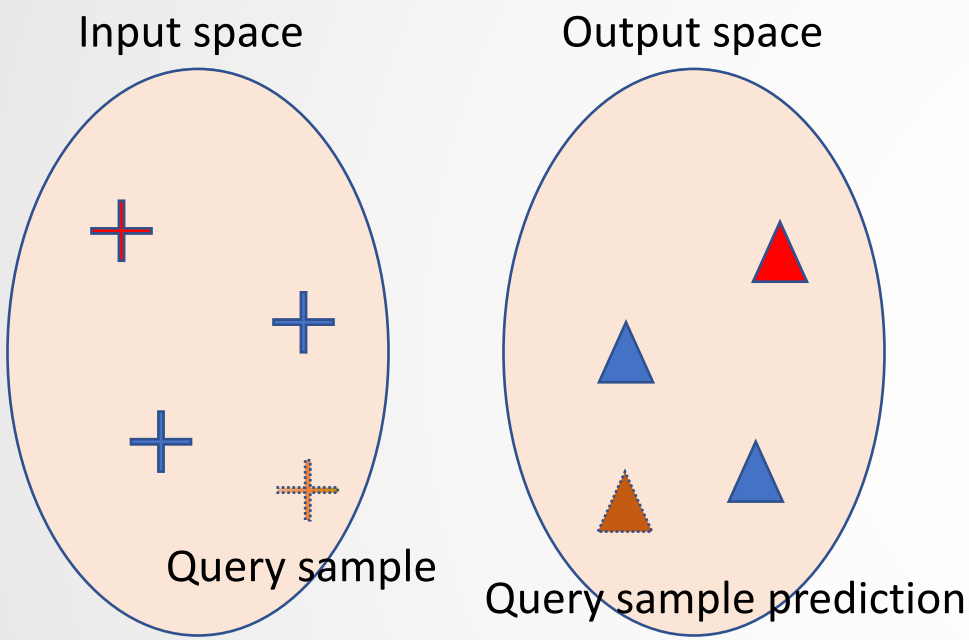
- What's the exact distance?

- Visual ambiguity and thin items foreground, problems background (e.g., distance limitation of sensor, less data available)

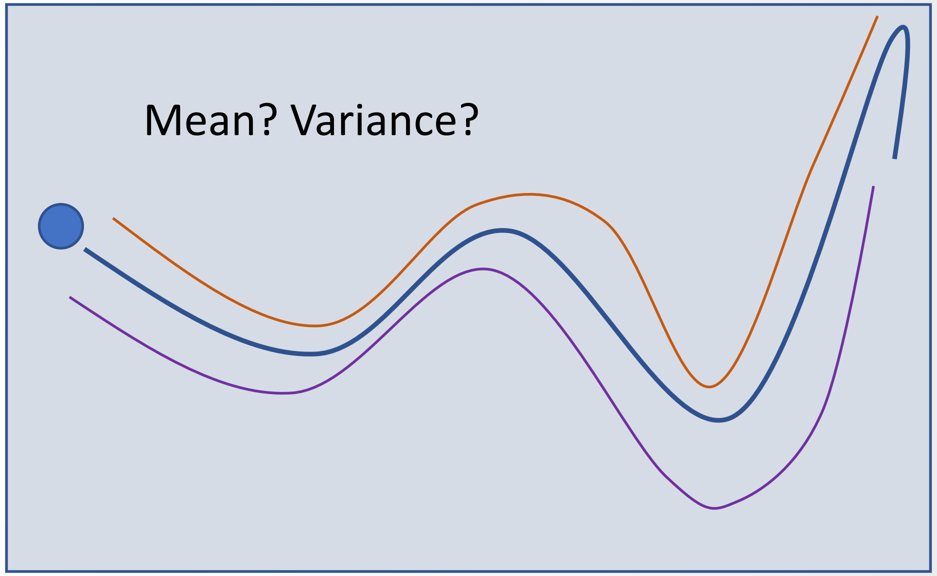
Motivation

- Model can make mistakes sometimes
- Model should be aware when it makes mistakes

Uncertainty: a mechanism to understand model limitations



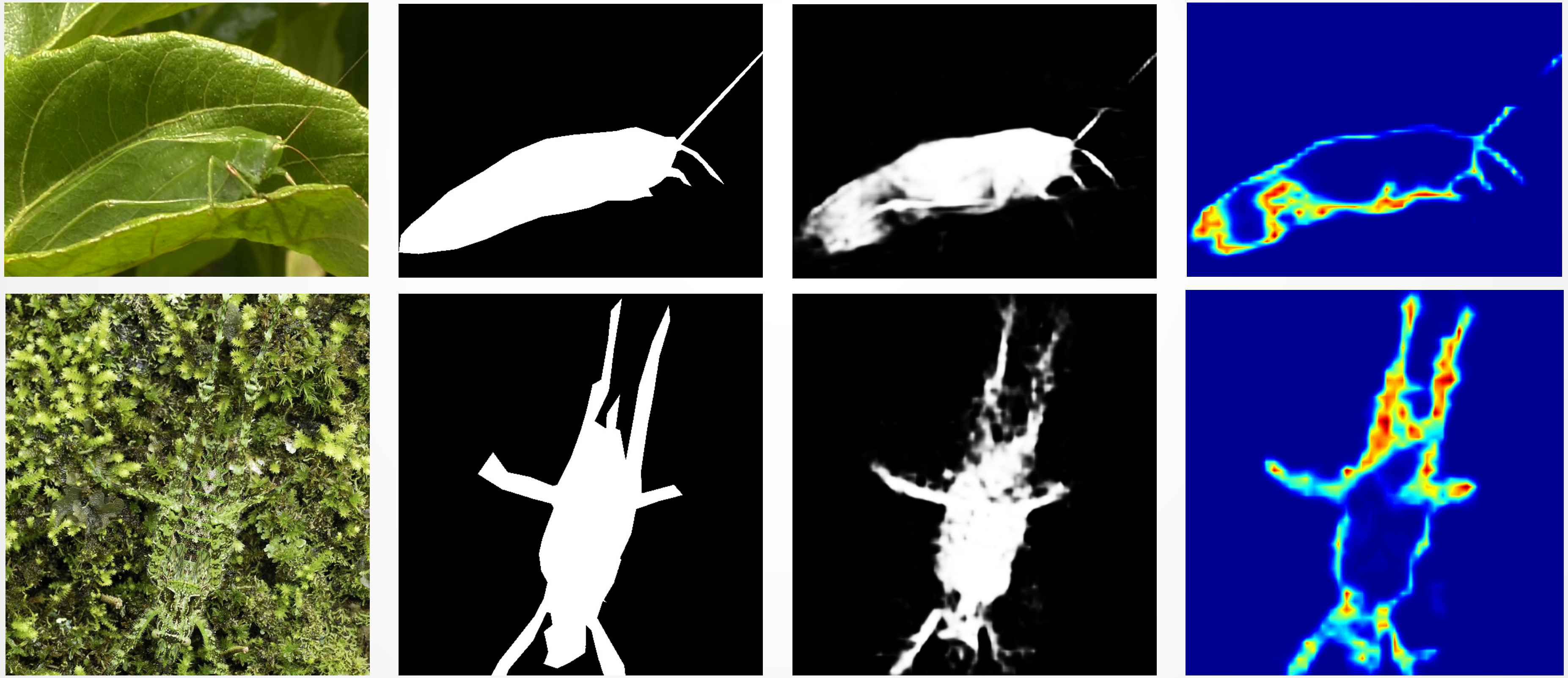
Confidence of p(y|x)?



Variance of prediction?

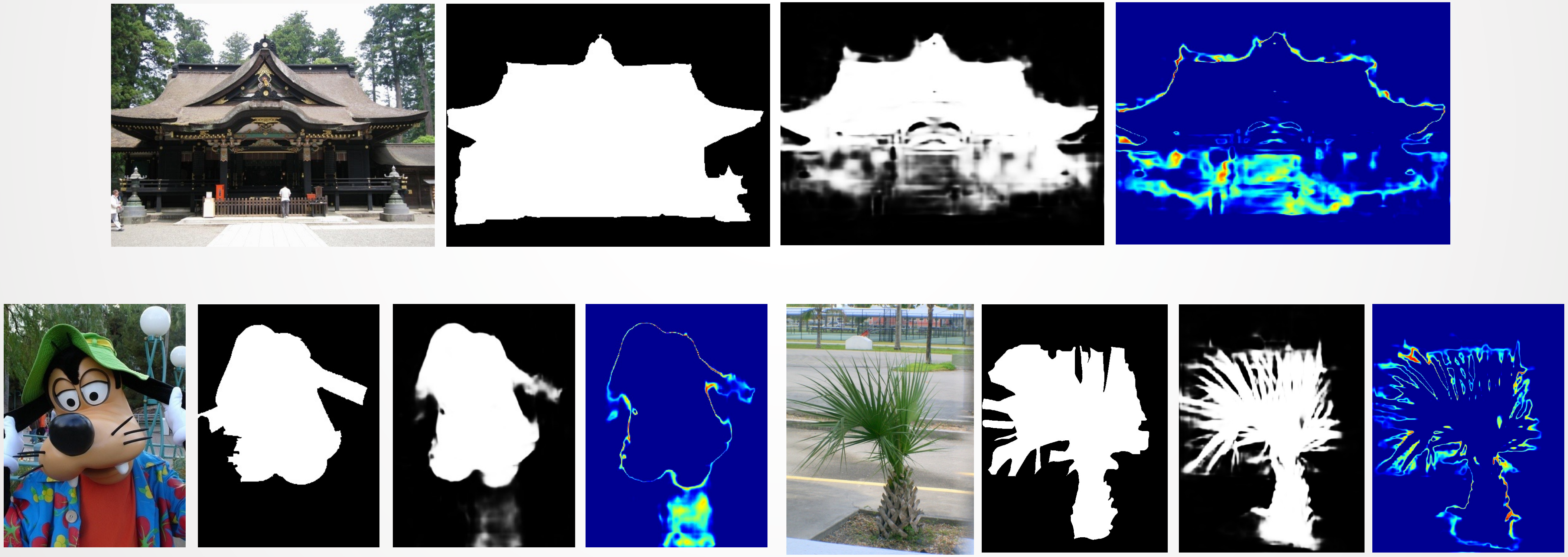
Motivation

- Camouflaged object detection



Motivation

- Salient object detection



Motivation

- Monocular depth estimation

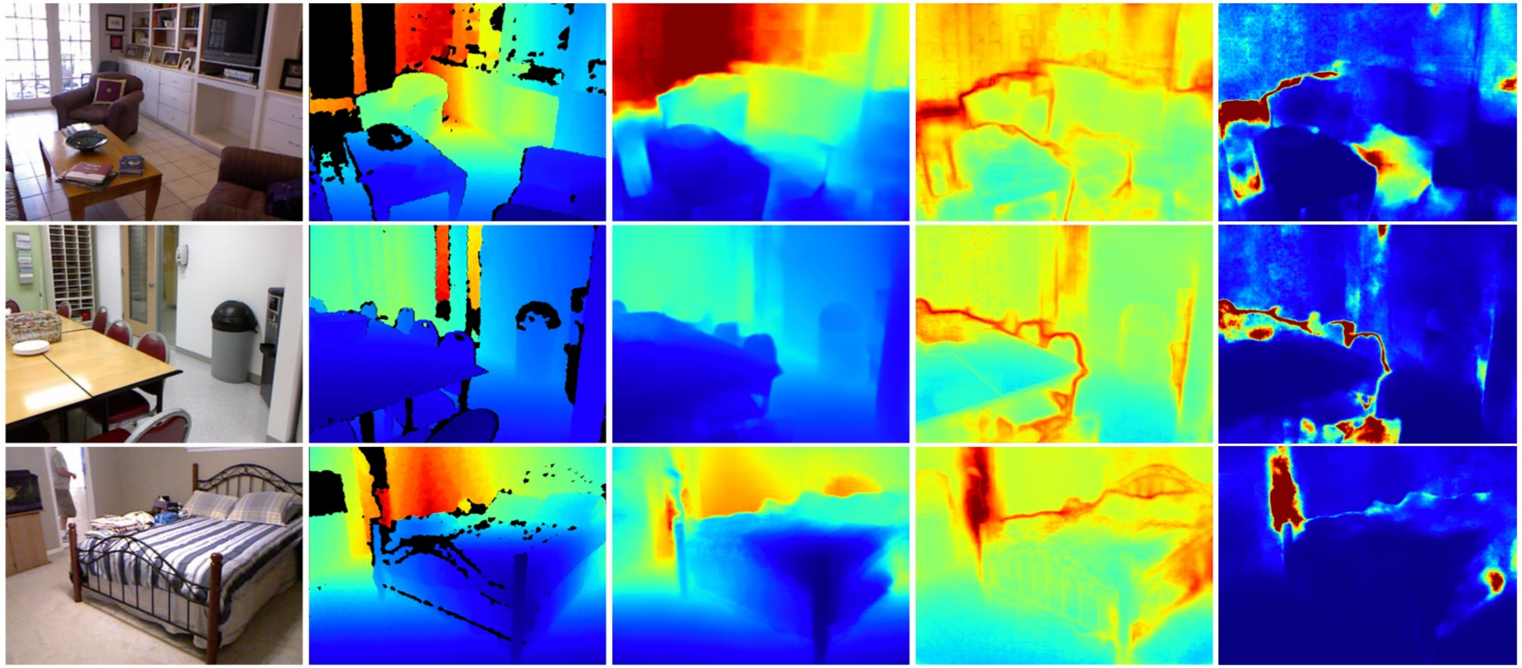
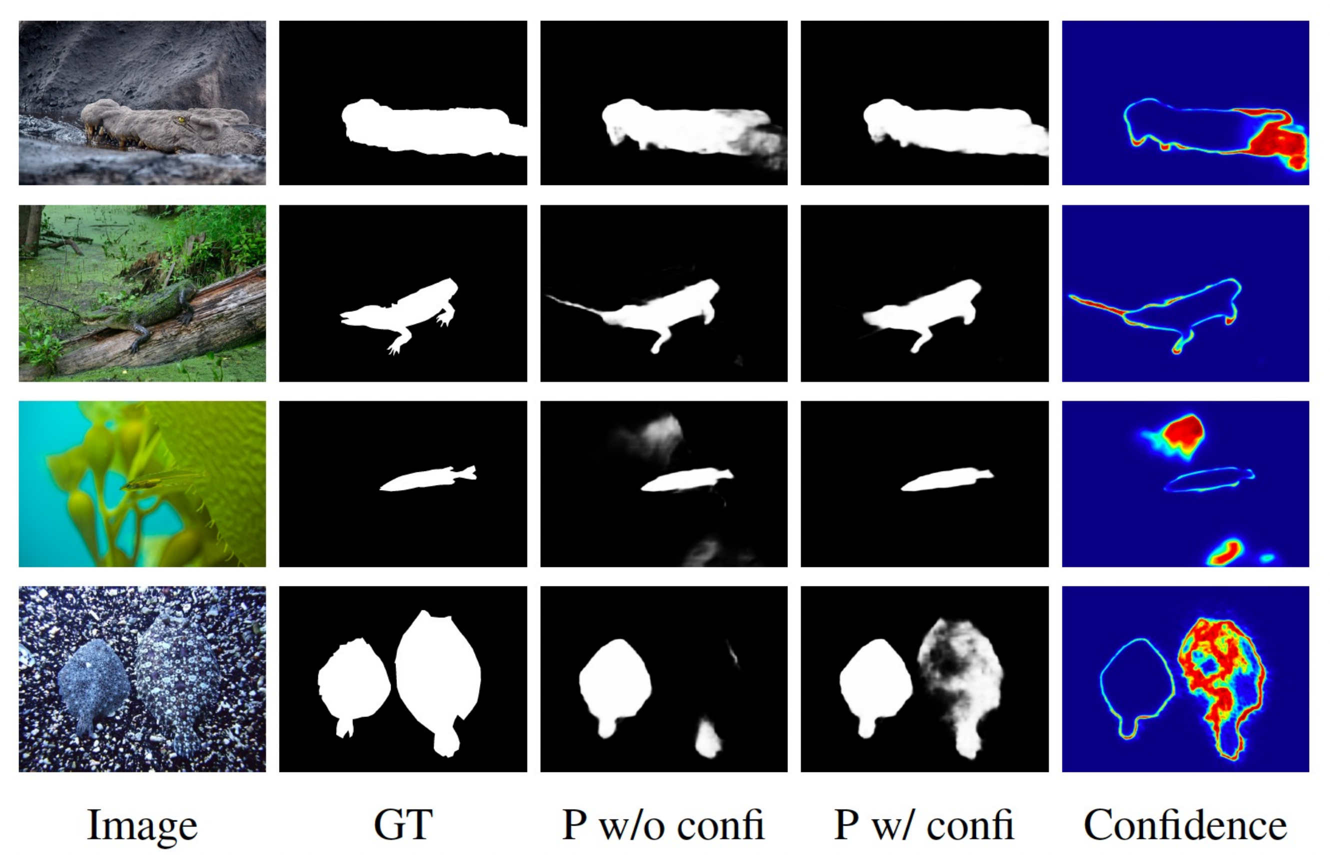


Figure 5: NYUv2 Depth results. From left: input image, ground truth, depth regression, aleatoric uncertainty, and epistemic uncertainty.

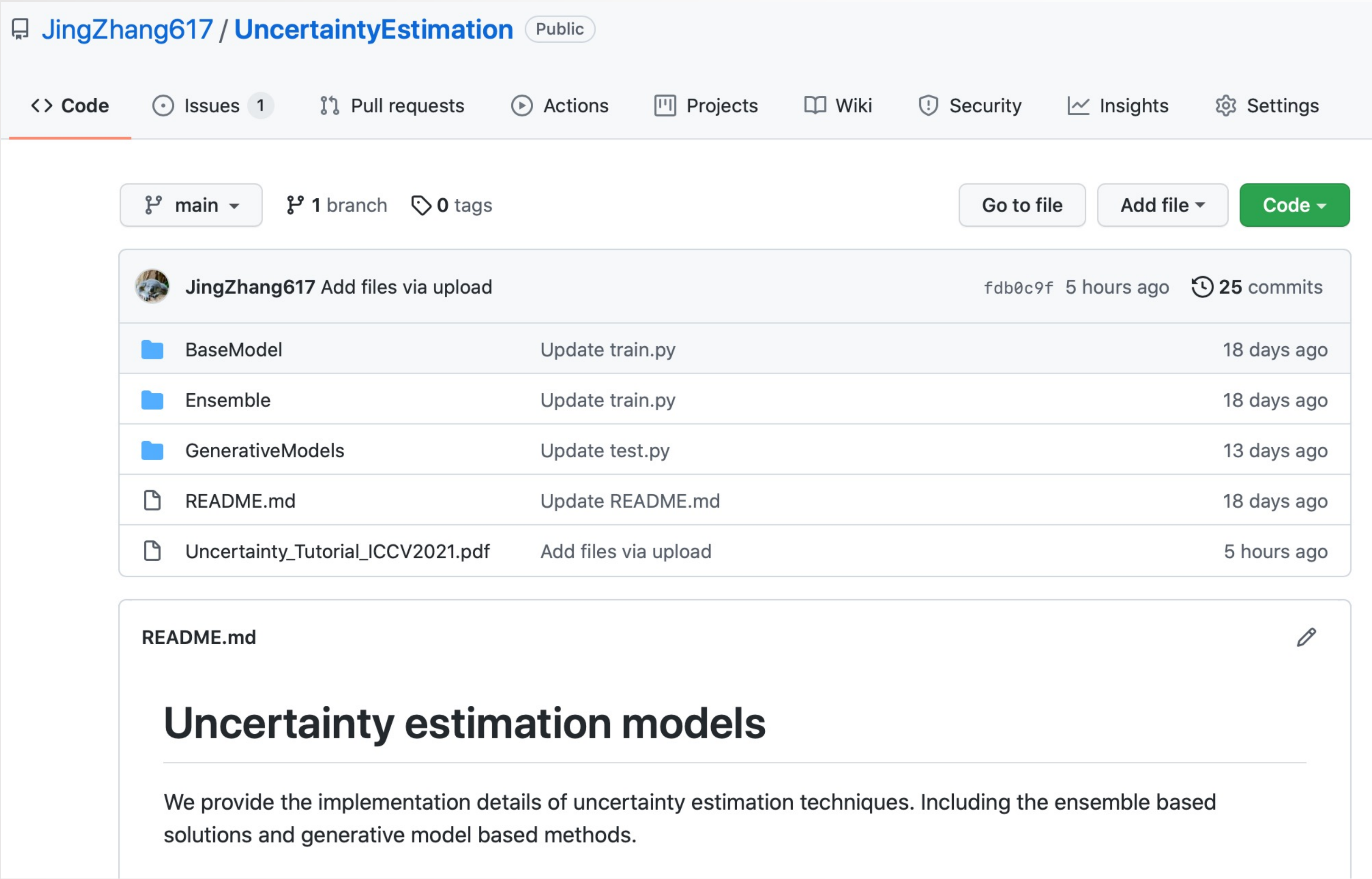
Figure from “What uncertainties do we need in bayesiandeep learning for computer vision?” by A. Kendall and Y. Gal.

Motivation

- Well-calibrated model
- Hard-negative mining
- Dynamic confidence supervision



Confidence-Aware Learning for Camouflaged Object Detection, Liu, Zhang, Barnes, WACV 2022



Code and tutorial material are available

Outline

- Motivation
- Background
- Aleatoric uncertainty and Epistemic Uncertainty
- Uncertainty Approximation
  - ✓Ensemble solutions
  - ✓Generative model solutions
  - ✓Bayesian latent variable model solutions
- Experiments
- Discussion and Conclusion

Background

Given training dataset $D = \{x_i, y_i\}_{i=1}^N$, the goal of machine learning methods:

$$\min_\theta \mathbb{E}_{x,y}[\mathcal{L}(f(x,\theta), y)] = \int \mathcal{L}(f(x,\theta), y) dp(x,y) \approx \frac{1}{N}\sum_{i=1}^N \mathcal{L}(f(x_i,\theta), y_i), (x_i, y_i) \sim p(x,y)$$

Loss over a continuous distribution is approximated by empirical loss over a dataset.

Background

Given training dataset $D = \{x_i, y_i\}_{i=1}^{N}$, the goal of machine learning methods:

$$\min_{\theta} \mathbb{E}_{x,y}[\mathcal{L}(f(x,\theta), y)] = \int \mathcal{L}(f(x,\theta), y) dp(x,y) \approx \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(f(x_i,\theta), y_i), (x_i, y_i) \sim p(x,y)$$

$\mathbb{E}_{x,y}$: Expectation over input
$f(x,\theta)$: model hypothesis for x given model parameters θ
$\mathcal{L}$: (.,.) loss
$(x_i, y_i) \sim p(x,y)$: (x_i, y_i) input/output sampled from true distribution

Ambiguity comes from θ and the representativeness of the sampled dataset D.

Background

Given training dataset $D = \{x_i, y_i\}_{i=1}^{N}$, the predictive distribution is defined as:

$$p(y|x) = \int p(y|x,\theta)p(\theta)d\theta$$

where the likelihood of prediction is defined as:

$p(y|x,\theta) = \mathcal{N}(f(x,\theta),\Sigma)$ (Regression) Or $p(y|x,\theta) = Softmax(\frac{f(x,\theta)}{\exp(\sigma^2)})$ (Classification)

Background

Given training dataset $D = \{x_i, y_i\}_{i=1}^{N}$, the predictive distribution is defined as:

$$p(y|x) = \int p(y|x,\theta)p(\theta)d\theta$$

where the likelihood of prediction is defined as:

$p(y|x,\theta) = \mathcal{N}(f(x,\theta),\Sigma)$ (Regression) Or $p(y|x,\theta) = Softmax(\frac{f(x,\theta)}{\exp(\sigma^2)})$ (Classification)

Uncertainty origin: (ϵ / Σ), $p(\theta)$

Aleatoric uncertainty
(Intrinsic randomness of a phenomenon [1,2])
e.g., sensor noise, labeling noise

Epistemic uncertainty
(Lack of knowledge of data [1,2])
Model noise

1. "Aleatory or epistemic? does it matter?, A. D. Kiureghian and O. Ditlevsen, Structural Safety, 2009.
2. What uncertainties do we need in Bayesian deep learning for computer vision? A. Kendall and Y. Gal. NeurIPS 2017

Background

Uncertainty origin: (ϵ / Σ), $p(\theta)$

Aleatoric uncertainty: data related uncertainty

(Intrinsic randomness of a phenomenon [1,2])

Epistemic uncertainty: model related uncertainty

(Lack of knowledge of data [1,2])

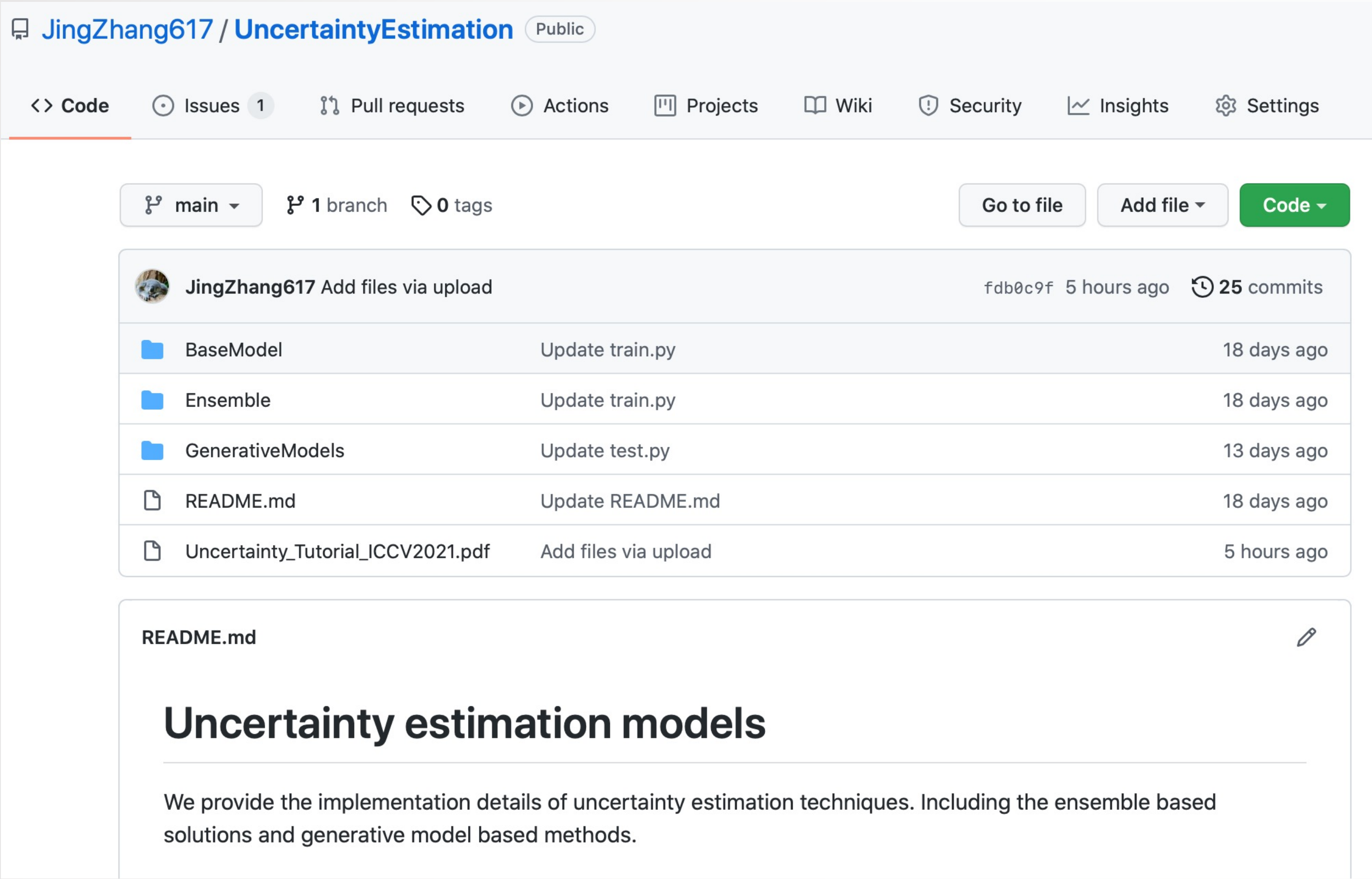
Point estimation system vs Self-awareness of machine learning system

Decision making: Medical diagnosis, automatous driving, …

1. "Aleatory or epistemic? does it matter?, A. D. Kiureghian and O. Ditlevsen, Structural Safety, 2009.
2. What uncertainties do we need in Bayesian deep learning for computer vision? A. Kendall and Y. Gal. NeurIPS 2017

Examples

Example and image from: [Dealing with Overconfidence in Neural Networks: Bayesian Approach – Jonathan Ramkissoon (jramkiss.github.io)]



Code and tutorial material are available

Outline

- Motivation
- Background
- Aleatoric uncertainty and Epistemic Uncertainty
- Uncertainty Approximation
    - ✓Ensemble solutions
    - ✓Generative model solutions
    - ✓Bayesian latent variable model solutions
- Experiments
- Discussion and Conclusion

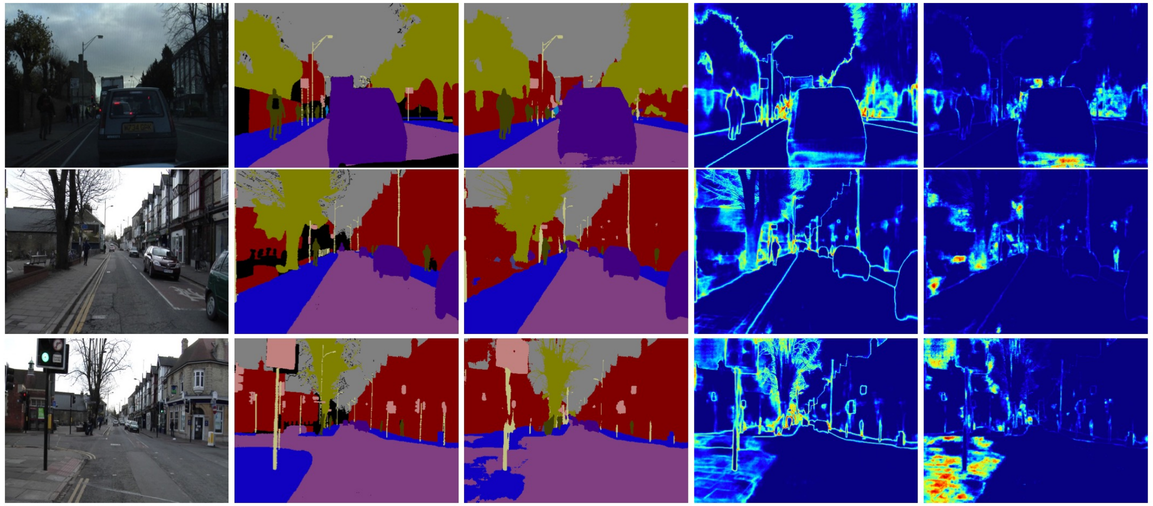
Aleatoric Uncertainty and Epistemic Uncertainty

- Aleatoric uncertainty: randomness, inherent noise
- Epistemic uncertainty: lack of knowledge, can be explained away with enough data, while aleatoric uncertainty cannot.

Aleatoric uncertainty: function of input, model it over outputs
Epistemic uncertainty: function of model, model it over network parameters

Examples

- What is aleatoric uncertainty? What is epistemic uncertainty?



(a) Input Image (b) Ground Truth (c) Semantic Segmentation (d) Aleatoric Uncertainty (e) Epistemic Uncertainty

Figure 1: **Illustrating the difference between aleatoric and epistemic uncertainty** for semantic segmentation on the CamVid dataset [8]. *Aleatoric* uncertainty captures noise inherent in the observations. In (d) our model exhibits increased aleatoric uncertainty on object boundaries and for objects far from the camera. *Epistemic* uncertainty accounts for our ignorance about which model generated our collected data. This is a notably different measure of uncertainty and in (e) our model exhibits increased epistemic uncertainty for semantically and visually challenging pixels. The bottom row shows a failure case of the segmentation model when the model fails to segment the footpath due to increased epistemic uncertainty, but not aleatoric uncertainty.

Figure from “What uncertainties do we need in bayesiandeep learning for computer vision?” by A. Kendall and Y. Gal.

Aleatoric uncertainty modeling

- Image-level classification
  - Image-independent aleatoric uncertainty—homoscedastic uncertainty
  - Image-conditional aleatoric uncertainty---heteroscedastic uncertainty
- Pixel-level Classification/Regression
  - Pixel-conditional uncertainty
  - Image-level uncertainty
  - Dataset-level uncertainty

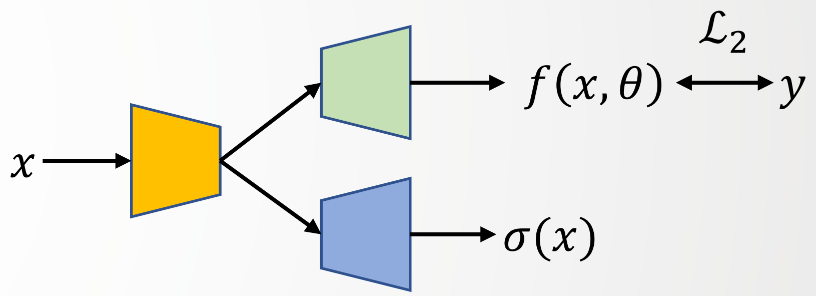
Constant vs Learned

Aleatoric Uncertainty--Regression:

For Gaussian likelihood: $p(y|x,\theta) = \mathcal{N}(f(x,\theta), \Sigma(x))$

$\Sigma(x) = diag((\sigma(x))^2)$ is the inherent label noise with the basic assumption:

$y = f(x,\theta) + n(x), n(x) \sim \mathcal{N}(0, \Sigma(x))$

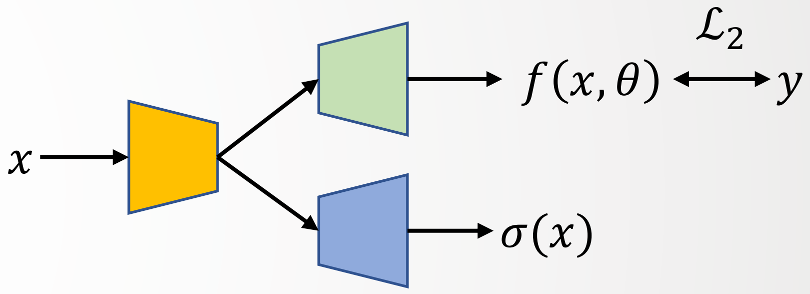


Uncertainty-aware loss: $\mathcal{L}(\theta) = \frac{1}{N}\sum_{i=1}^{N}(\frac{1}{2(\sigma(x))^2}\mathcal{L}_2 + \frac{1}{2}\log((\sigma(x))^2))$

Aleatoric Uncertainty--Regression:

For Gaussian likelihood: $p(y|x,\theta) = \mathcal{N}(f(x,\theta), \Sigma(x))$

$$\mathcal{L}(\theta) = \frac{1}{N}\sum_{i=1}^{N} (\frac{1}{2(\sigma(x))^2}\mathcal{L}_2 + \frac{1}{2}\log((\sigma(x))^2))$$



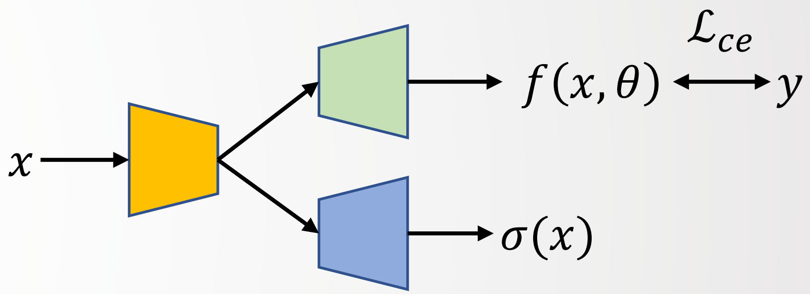
For numerical stability, define $s_i = \log\left((\sigma(x))^2\right)$: $\mathcal{L}(\theta) = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{1}{2}\exp(-s_i)\,\mathcal{L}_2 + \frac{1}{2}s_i\right)$

Aleatoric uncertainty: $U_a = (\sigma(x))^2$

Aleatoric Uncertainty--Classification:

For SoftMax likelihood where $p(y|x,\theta) = Softmax(\frac{f(x,\theta)}{\exp(\sigma^2)})$

$$\mathcal{L}(\theta) = \frac{1}{N}\sum_{i=1}^{N}(\frac{1}{\exp(\sigma^2)}\mathcal{L}_{ce} + \frac{1}{2}\sigma^2)$$



For numerical stability, define $\mathrm{T} = \exp(\sigma^2)$ (the temperature):

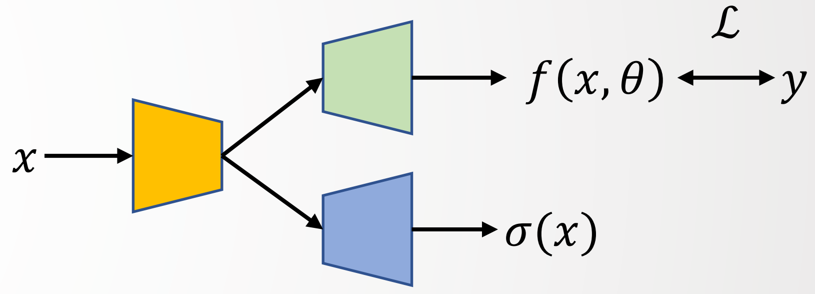
$$\mathcal{L}(\theta) = \frac{1}{N}\sum_{i=1}^{N}(\frac{1}{T}\mathcal{L}_{ce} + \frac{1}{2}\log(T))$$

$$U_a = (\sigma(x))^2$$

Trivial solution:

For Gaussian likelihood: $p(y|x,\theta) = \mathcal{N}(f(x,\theta), \Sigma(x))$

$$\mathcal{L}(\theta) = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{1}{2\sigma^2}\mathcal{L}_2 + \frac{1}{2}\log(\sigma^2)\right)$$

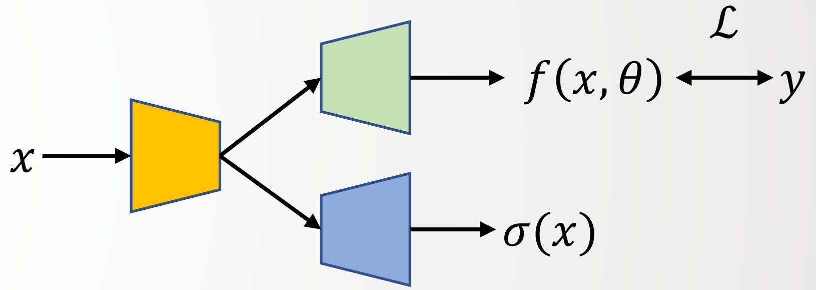


$\sigma(x)$=1 !!, constant! Image-independent!!

Trivial solution:

For SoftMax likelihood where $p(y|x,\theta) = Softmax(\frac{f(x,\theta)}{\exp(\sigma^2)})$

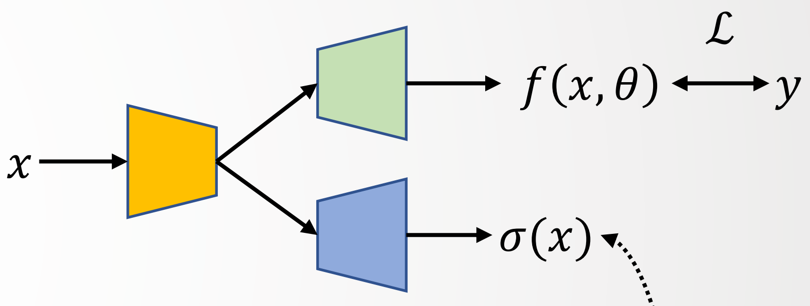
$$\mathcal{L}(\theta) = \frac{1}{N}\sum_{i=1}^{N}(\frac{1}{\exp(\sigma^2)}\mathcal{L}_{ce} + \frac{1}{2}\sigma^2)$$



$\sigma(x)$=0 !! Temperature=1!!

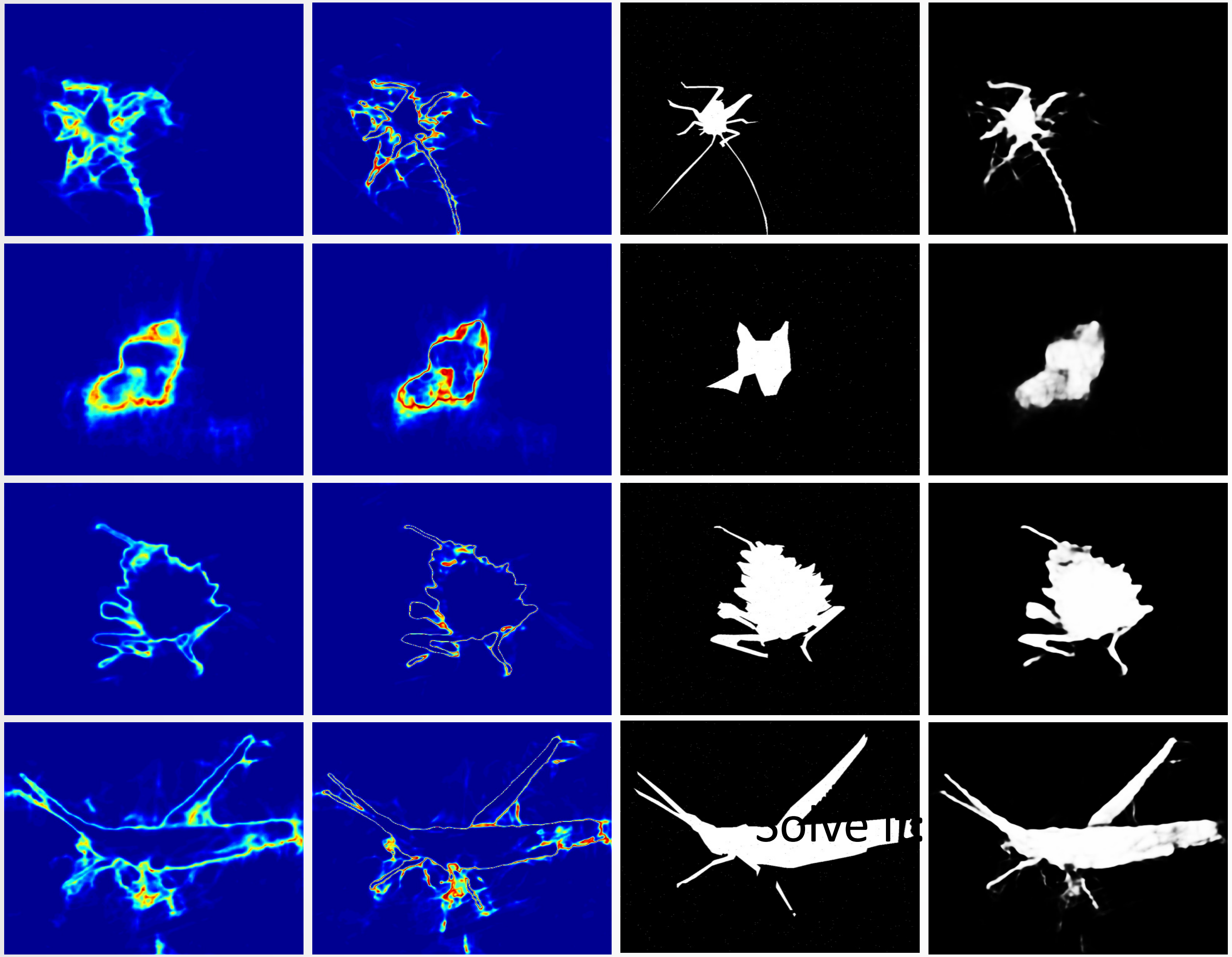
Avoid the Trivial solution:

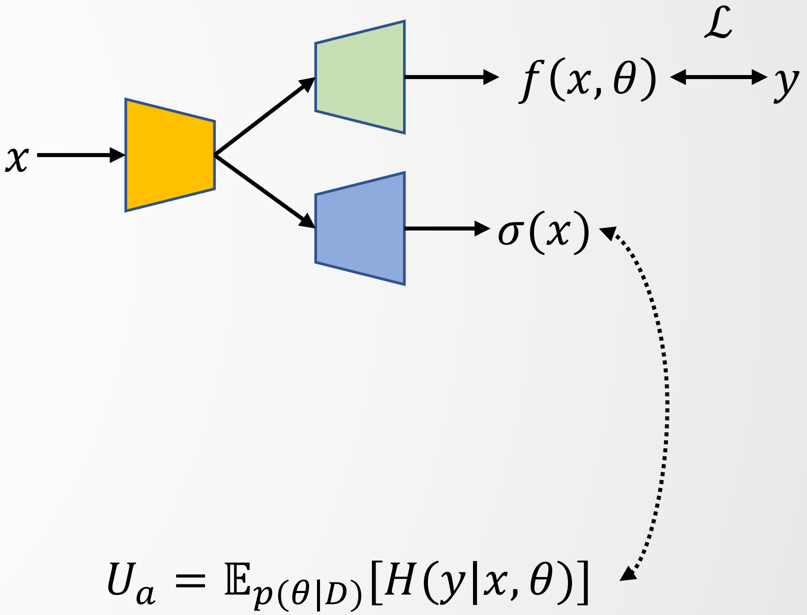
- Two types of aleatoric uncertainty definition
  - ➢Multi-head
  - ➢Mean entropy within BNN
- Uncertainty consistency loss for sampling free aleatoric uncertainty estimation



Solve it: $$U_a = \mathbb{E}_{p(\theta|D)}[H(y|x,\theta)]$$

Stefan Depeweg et. al. Decomposition of Uncertainty in Bayesian Deep Learning for Efficient and Risk-sensitive Learning. ICML. 2018


Dual-head
Entropy
GT
Prediction


x
$f(x, \theta)$
$\mathcal{L}$
y
$\sigma(x)$
$U_a = \mathbb{E}_{p(\theta|D)}[H(y|x, \theta)]$

Epistemic Uncertainty

- Not directly estimated
- Defined as residual of predictive uncertainty and aleatoric uncertainty
- Mutual information of model prediction and model parameters within Bayesian Neural Network (BNN).

Given predictive uncertainty $U_p = H(y|x)$

Epistemic uncertainty: $U_e = U_p - U_a = H(y|x) - \mathbb{E}_{p(\theta|D)}[H(y|x,\theta)] = I(y,\theta|x)$

$$I(y,\theta|x) = H(y|x) - H(y|x,\theta) = H(y|x) - \mathbb{E}_{p(\theta|D)}[H(y|x,\theta)]$$

Stefan Depeweg et. al. Decomposition of Uncertainty in Bayesian Deep Learning for Efficient and Risk-sensitive Learning. ICML. 2018

Epistemic Uncertainty

For Gaussian likelihood with aleatoric uncertainty σ^2 from multi-head, the entropy based uncertainty is reduced to a function of variance, leading to epistemic uncertainty:

$$U_e = \mathbb{E}_{p(\theta|D)}[p(y|x,\theta)^2] - (\mathbb{E}_{p(\theta|D)}[p(y|x,\theta)])^2$$

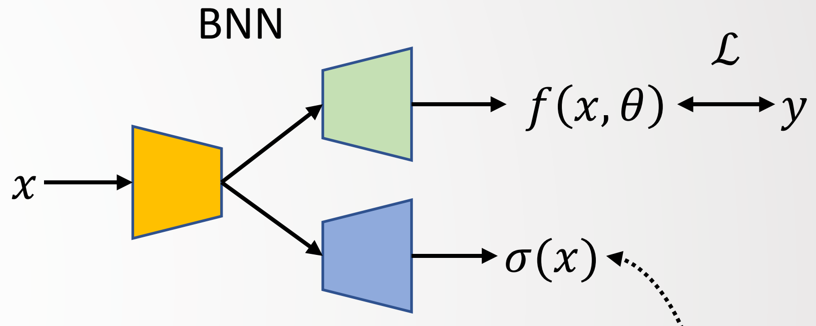
The predictive uncertainty is then:

$$U_p = \underbrace{\mathbb{E}_{p(\theta|D)}[p(y|x,\theta)^2] - (\mathbb{E}_{p(\theta|D)}[p(y|x,\theta)])^2}_{\text{epistemic uncertainty}} + \underbrace{\mathbb{E}_{p(\theta|D)}[\sigma(x)^2]}_{\text{aleatoric uncertainty}}$$

1. What uncertainties do we need in Bayesian deep learning for computer vision? A. Kendall and Y. Gal. NeurIPS 2017
2. Stefan Depeweg et. al. Decomposition of Uncertainty in Bayesian Deep Learning for Efficient and Risk-sensitive Learning. ICML. 2018

Epistemic Uncertainty

SoftMax likelihood for classification:



$$U_p = H(y|x)$$

$$U_e = U_p - U_a = H(y|x) - \mathbb{E}_{p(\theta|D)}[H(y|x,\theta)] = I(y,\theta|x)$$

$$U_a = \sigma(x)^2 \longleftrightarrow U_a = \mathbb{E}_{p(\theta|D)}[H(y|x,\theta)]$$

Break

Outline

- Motivation
- Background
- Aleatoric uncertainty and Epistemic Uncertainty
- Uncertainty Approximation
  - ✓Ensemble solutions
  - ✓Generative model solutions
  - ✓Bayesian latent variable model solutions
- Experiments
- Discussion and Conclusion

Epistemic Uncertainty

- Not directly estimated
- Defined as residual of predictive uncertainty and aleatoric uncertainty
- Mutual information of model prediction and model parameters within Bayesian Neural Network (BNN).

Given predictive uncertainty $U_p = H(y|x)$

Epistemic uncertainty: $U_e = U_p - U_a = H(y|x) - \mathbb{E}_{p(\theta|D)}[H(y|x,\theta)] = I(y,\theta|x)$

$$I(y,\theta|x) = H(y|x) - H(y|x,\theta) = H(y|x) - \mathbb{E}_{p(\theta|D)}[H(y|x,\theta)]$$

Stefan Depeweg et. al. Decomposition of Uncertainty in Bayesian Deep Learning for Efficient and Risk-sensitive Learning. ICML. 2018

Uncertainty Approximation

- Recall that epistemic uncertainty is usually directly estimated, which is defined as residual of predictive uncertainty and aleatoric uncertainty or the mutual information of model prediction and model parameters within Bayesian Neural Network (BNN).

Given predictive uncertainty $U_p = H(y|x)$

Epistemic uncertainty: $U_e = U_p - U_a = H(y|x) - \mathbb{E}_{p(\theta|D)}[H(y|x,\theta)] = I(y,\theta|x)$

$$I(y,\theta|x) = H(y|x) - H(y|x,\theta) = H(y|x) - \mathbb{E}_{p(\theta|D)}[H(y|x,\theta)]$$

Stefan Depeweg et. al. Decomposition of Uncertainty in Bayesian Deep Learning for Efficient and Risk-sensitive Learning. ICML. 2018

Uncertainty Approximation

- Recall: $U_e = U_p - U_a = H(y|x) - \mathbb{E}_{p(\theta|D)}[H(y|x,\theta)] = I(y,\theta|x)$
- no close form solution for $p(\theta|D)$

$p(\theta|D) = p(\theta|x,y) = p(y|x,\theta)p(\theta)/p(y|x)$, where $p(y|x)$ cannot be evaluated analytically, thus no close form solution for $p(\theta|D)$.

- Approximation for Bayesian posterior inference
  - Variational inference

  Approximate $p(\theta|D)$ with easy-controlled distribution $q_\gamma(\theta)$, γ: variational parameters. i.e. MC-dropout

  - Markov chain Monte Carlo (MCMC) methods

  Sampling based solution, correlated sequence of $\theta_t \sim p(\theta|D)$. MC average is used as approximation of expectation----Generative model based solutions

Uncertainty Approximation

- Ensemble solutions
  1. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. Yarin Gal, Zoubin Ghahramani. ICML. 2016
  2. Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles. Balaji Lakshminarayanan et. al. NeurIPS 2017
- Generative model solutions
  1. Adversarial distillation of Bayesian neural network posteriors. Kuan-Chieh Wang et. al. ICML. 2018
- Bayesian latent variable model solutions
  1. What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? Alex Kendall, Yarin Gal. NeurIPS. 2017
  2. Decomposition of Uncertainty in Bayesian Deep Learning for Efficient and Risk-sensitive Learning. Stefan Depeweg et. al. ICML. 2018

Outline

- Motivation
- Background
- Aleatoric uncertainty and Epistemic Uncertainty
- Uncertainty Approximation
  - ✓ Ensemble solutions
    - ✓ MC-dropout
    - ✓ Deep ensemble
    - ✓ Snapshots ensemble
  - ✓ Generative model solutions
  - ✓ Bayesian latent variable model solutions
- Experiments
- Discussion and Conclusion

Ensemble Solutions

- MC dropout:

True predictive distribution:

$$p(y|x) = \int p(y|x,\theta)p(\theta)d\theta$$

MC average as approximation: $p(y|x) \approx \frac{1}{T}\sum_{t=1}^{T} p(y_t|x,\theta_t)$

Where θ_t is sampled from the approximate posterior distribution $q_\gamma(\theta)$

Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. Yarin Gal, Zoubin Ghahramani. ICML. 2016

Ensemble Solutions

- MC dropout:

  Implementation details: add dropout before every weighted layer during both training and testing.

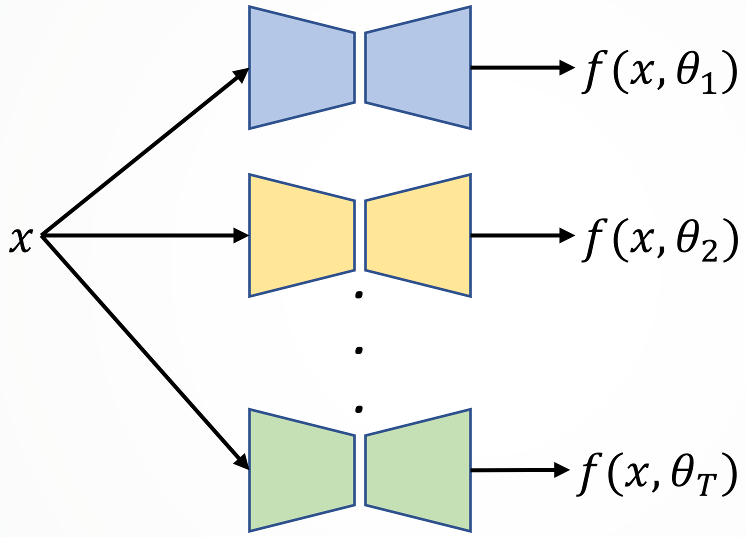
  Pros: easy to implement, no additional parameters

  Cons: cannot control the dropout mask, mode collapse issue

Masksembles for Uncertainty Estimation. Nikita Durasov et. al. CVPR. 2021.

Ensemble Solutions

- Deep ensemble



Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles.
Lakshminarayanan, Balaji et. al. NeurIPS. 2017

Ensemble Solutions

- Deep ensemble

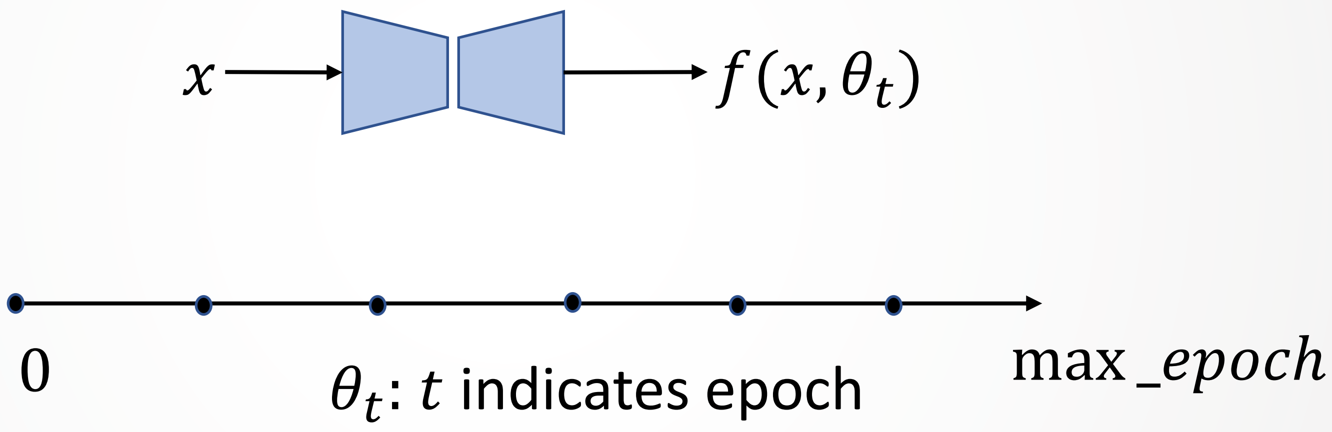
  Implementation details: random initialization, multi-network, multi-head.

  Pros: easy to implement, usually suffer no mode collapse issue

  Cons: extra parameters leading to longer training time, fixed number of predictions, not very flexible.

Ensemble Solutions

- Snapshot ensemble



Snapshot Ensembles: Train 1, Get M for Free. Gao Huang et. al. ICLR. 2017

Ensemble Solutions

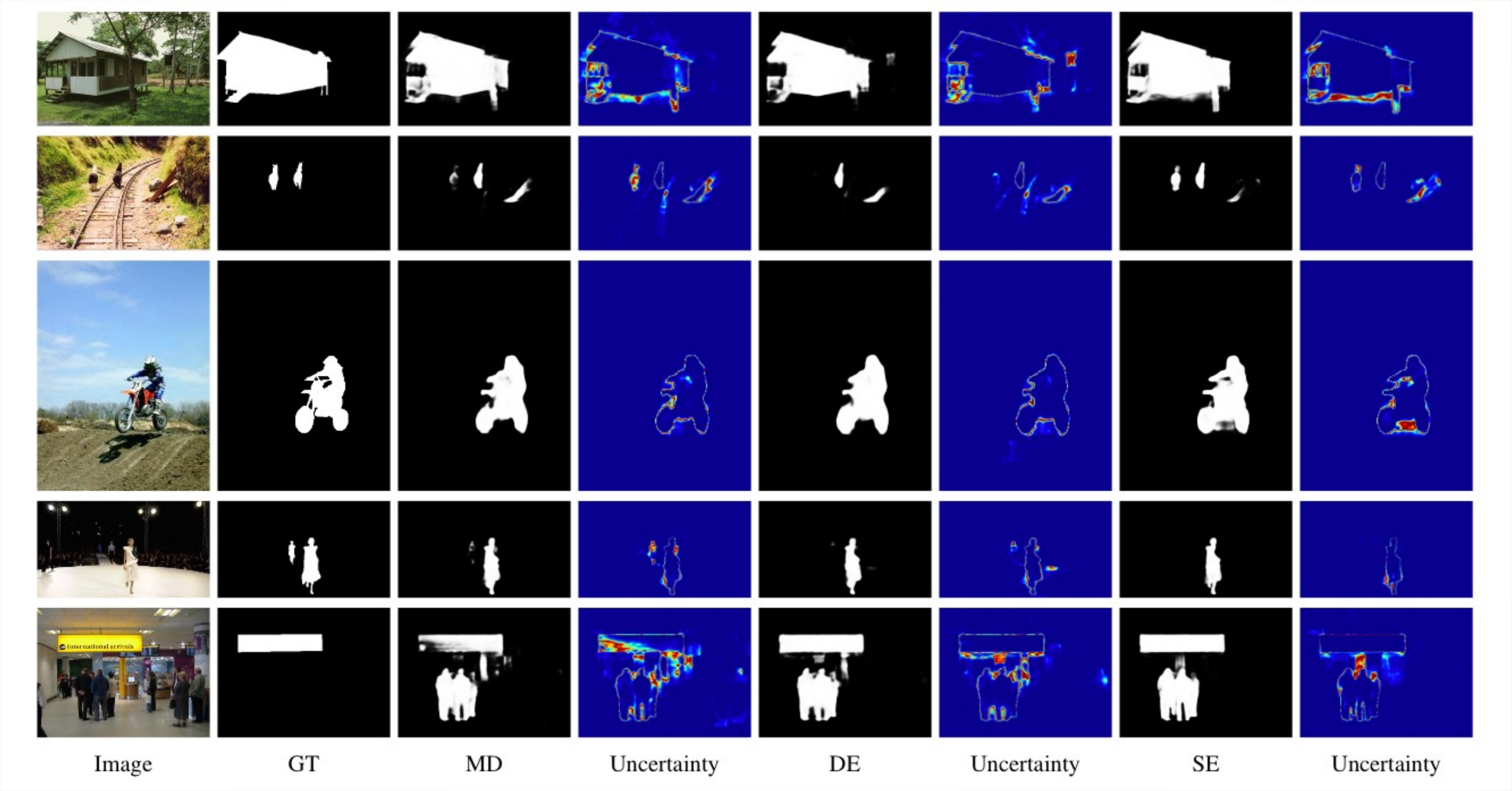
- Snapshot ensemble

  Implementation details: save multiple snapshots for multiple predictions

  Pros: no extra parameters, easy to implement

  Cons: hard to determine the snapshots point

Examples



MD: MC-dropout, DE: deep ensemble, SE: snapshot ensemble

Dense Uncertainty Estimation. Zhang et. al. 2021. https://github.com/JingZhang617/UncertaintyEstimatin

Ensemble based Uncertainty computation—regression

- Aleatoric uncertainty $U_a = \frac{1}{T}\sum_{t=1}^{T} H(y|x,\theta_t)$, or $U_a = \sigma(x)^2$
- Epistemic uncertainty

$$U_e = \mathbb{E}_{p(\theta|D)}[p(y|x,\theta)^2] - (\mathbb{E}_{p(\theta|D)}[p(y|x,\theta)])^2$$

or

$$U_e = H\left(\frac{1}{T}\sum_{t=1}^{T} p(y|x,\theta_t)\right) - \frac{1}{T}\sum_{t=1}^{T} H(y|x,\theta_t)$$

- Predictive uncertainty

$$U_p = U_a + U_e$$

Ensemble based Uncertainty computation--classification

- Aleatoric uncertainty

$$U_a = \frac{1}{T}\sum_{t=1}^{T} H(y|x,\theta_t), \text{ or } \; U_a = \sigma(x)^2$$

- Predictive uncertainty

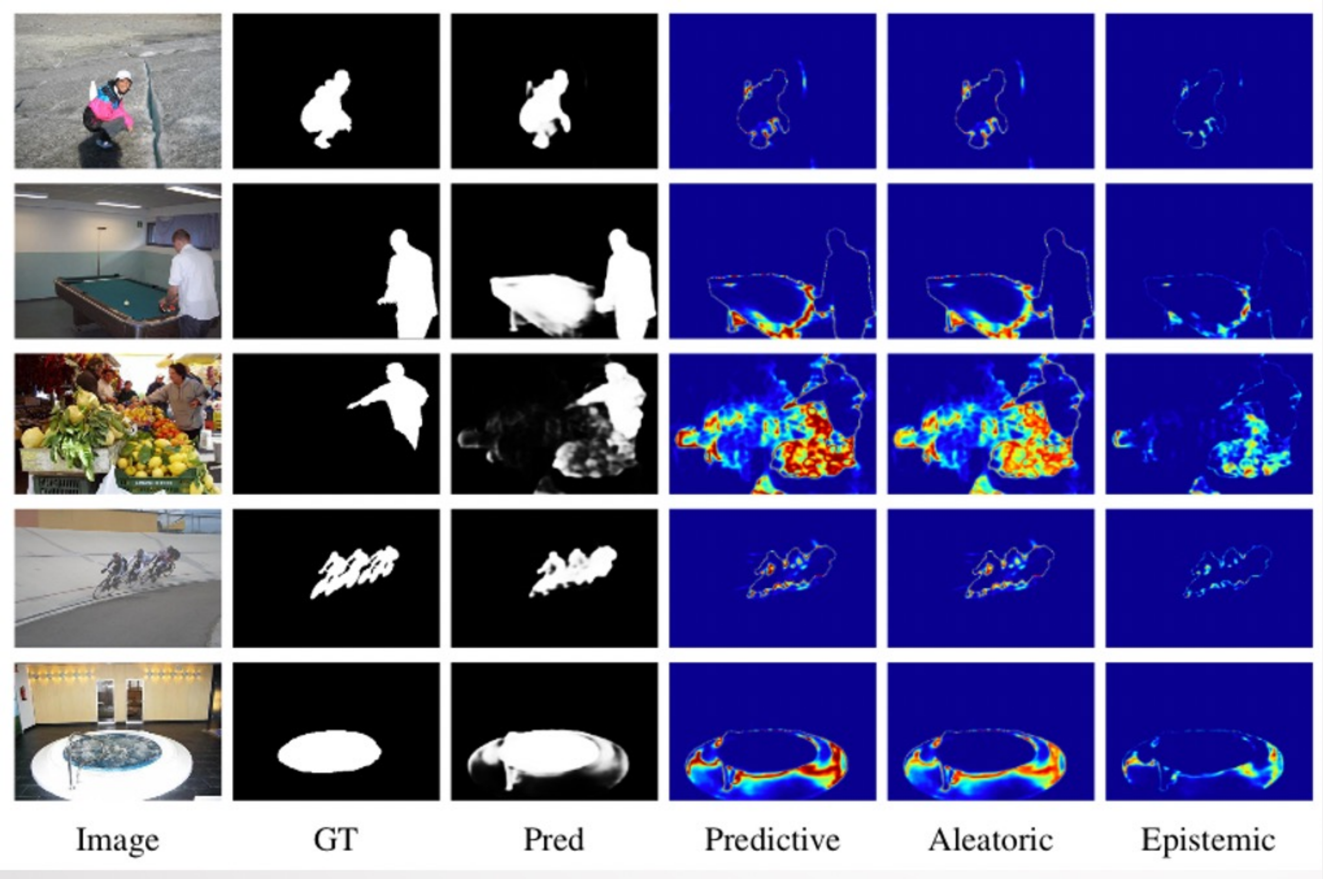
$$U_p = H(\frac{1}{T}\sum_{t=1}^{T} p(y|x,\theta_t))$$

- Epistemic uncertainty

$$U_e = H\left(\frac{1}{T}\sum_{t=1}^{T} p(y|x,\theta_t)\right) - \frac{1}{T}\sum_{t=1}^{T} H(y|x,\theta_t)$$

Examples

Aleatoric uncertainty: inherent noisy region
Epistemic uncertainty: hard samples
Predictive uncertainty: both



Dense Uncertainty Estimation. Zhang et. al. 2021. https://github.com/JingZhang617/UncertaintyEstimation

Outline

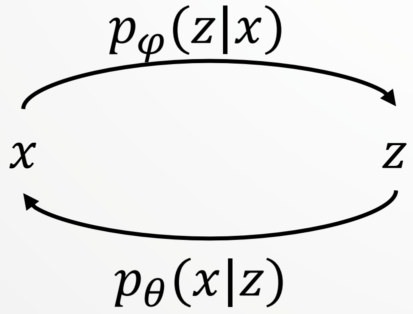
- Motivation
- Background
- Aleatoric uncertainty and Epistemic Uncertainty
- Uncertainty Approximation
  - ✓Ensemble solutions
  - ✓Generative model solutions
  - ✓Bayesian latent variable model solutions
- Experiments
- Discussion and Conclusion

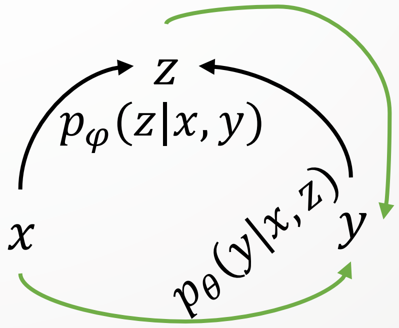
Generative model solutions

- Latent variable models
- Energy-based model

Latent variable models

- A latent variable is a variable which is not directly observable and is assumed to affect the observable variables (manifest variables)
- A latent variable model is a statistical model that relates a set of observable variables to a set of latent variables (Latent Variable Models and Factor Analysis, David J. Bartholomew et. al, 2011)
- More formally, a latent variable model (LVM) is a probability distribution over two sets of variables x and z as $p_\theta(x, z)$, or over three sets of variables x, y and z as $p_\theta(y|x, z)$ for the conditional version of the latent variable models with the conditional variable x.





Latent variable model solutions

- Predictive distribution with extra latent variable z

$$p(y|x) = \int p_\theta(y|x,z)p(\theta)p(z)d\theta dz$$

Regression Likelihood:

$$p_\theta(y|x,z) = \mathcal{N}(f_\theta(x,z),\Sigma)$$

Classification Likelihood:

$$p_\theta(y|x,z) = Softmax(f_\theta(x,z)/\exp(\sigma^2))$$

Uncertainty origin: $(\epsilon \; / \; \Sigma)$, z, $p(\theta)$

Aleatoric uncertainty

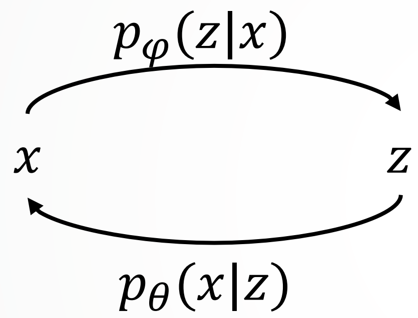
Epistemic uncertainty

Latent variable model solutions

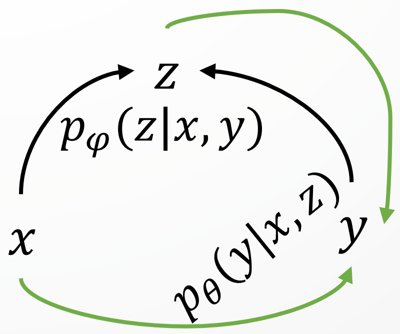
- CVAE—Conditional Variational Auto-encoder
  - Auto-Encoding Variational Bayes. Kingma, Diederik et. al. ICLR. 2014
  - Learning Structured Output Representation using Deep Conditional Generative Models. Sohn, Kihyuk et. al. NeurIPS. 2015
- CGAN—Conditional Generative Adversarial Nets
  - Generative Adversarial Nets. Goodfellow, Ian et. al. NeurIPS. 2014
  - Conditional Generative Adversarial Nets. Mehdi Mirza, Simon Osindero. arXiv. 2014
- ABP—Alternating Back-Propagation
  - Alternating Back-Propagation for Generator Network. Tian Han et. al. AAAI. 2016

Latent variable model solutions—VAE/CVAE

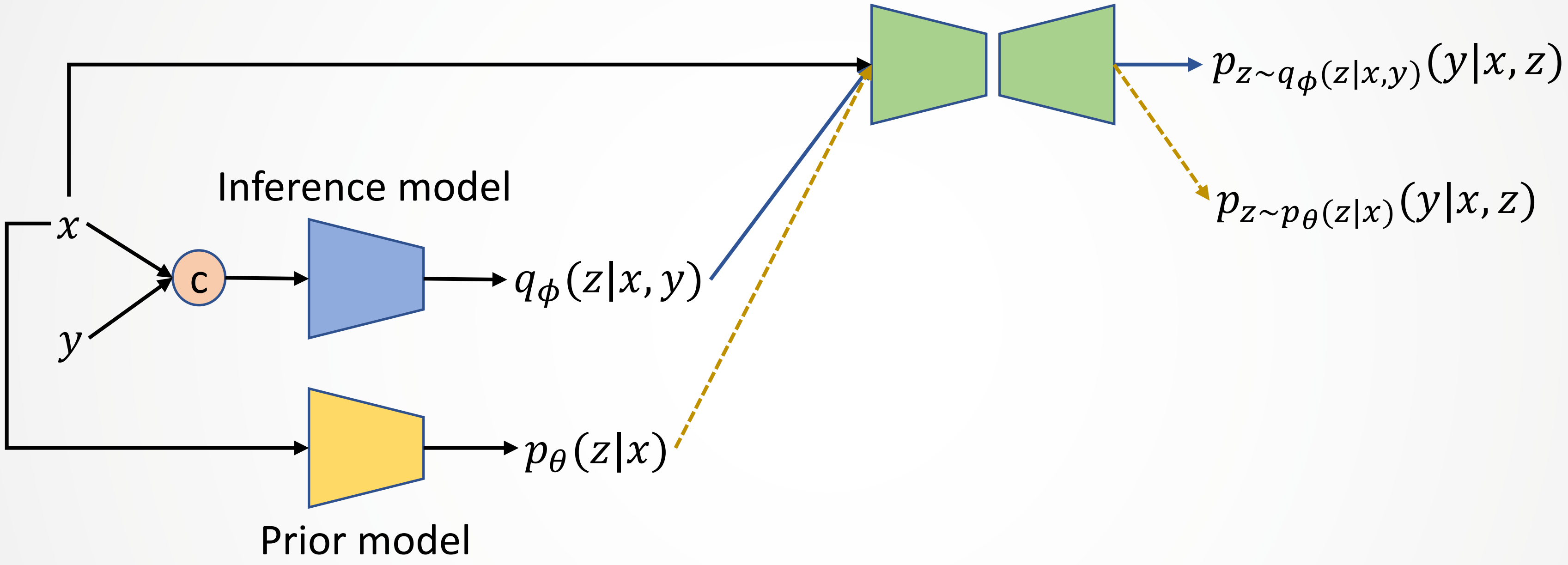
- VAE: unsupervised feature representation



- CVAE: latent feature exploring



- CVAE: conditional directed graph model



$$\mathcal{L}(\theta,\phi;x) = \mathbb{E}_{q_\phi(z|x,y)}\left[\log\left(p_\theta(y|x,z)\right)\right] - D_{KL}(q_\phi(z|x,y)||p_\theta(z|x))$$

1. A Probabilistic U-Net for Segmentation of Ambiguous Images. Simon Kohl et. al. NeurIPS. 2018
2. UC-Net: Uncertainty Inspired RGB-D Saliency Detection via Conditional Variational Autoencoders. Jing Zhang et. al. CVPR. 2020

Latent variable model solutions—GAN/CGAN

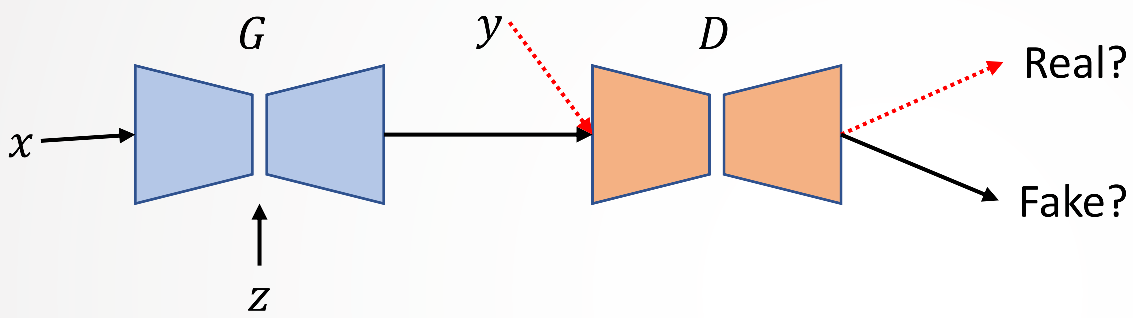
- GAN: min-max game
  - discriminator seeks to maximize the probability assigned to real and fake images

$$\max(\log\big(D(x)\big) + \log(1 - D(G(z)))) \text{ or } \min(\mathcal{L}_{ce}(D(x), 1) + \mathcal{L}_{ce}(D(G(z)), 0))$$

  - generator learns to generate samples that have a low possibility of being fake by minimizing the log of the inverse probability predicted by the discriminator for fake images.

$$\min(\,\log(1 - D(G(z))))$$

- CGAN for dense prediction



$$\mathcal{L}_G = \mathcal{L}_{CE}(f_G(x,z),y) + \lambda \mathcal{L}_{CE}(D(f_G(x,z),1))$$

$$\mathcal{L}_D = \mathcal{L}_{CE}(D(y),1) + \mathcal{L}_{CE}(D(f_G(x,z),0))$$

Adversarial Learning for Semi-Supervised Semantic Segmentation. Wei-Chih Hung et. al. BMVC 2018

Explain ABP

- VAE—Inference model is needed
- GAN–Discriminator is needed with no inference model

- ABP—Alternating Back-Propagation
  - ➢Infer the latent variable directly from the true posterior distribution
  - ➢No extra parameters

Alternating Back-Propagation

- Define the generative model as:

$$z \sim p(z) = \mathcal{N}(0,1)$$
$$y = f_\theta(x,z) + \epsilon, \epsilon \sim \mathcal{N}(0, \Sigma)$$

- The conditional distribution of y is defined as:

$$p_\theta(y|x) = \int p(z) p_\theta(y|x,z) dz$$

- Define the observed data log-likelihood as $\log(p_\theta(y|x))$, it's gradient is then:

$$\frac{\partial}{\partial\theta} \log\big(p_\theta(y|x)\big) = \mathbb{E}_{p_\theta(z|x,y)}\left[\frac{\partial}{\partial\theta} \log(p_\theta(y,z|x)\right]$$

Alternating Back-Propagation

- The observed data log-likelihood:

$$\frac{\partial}{\partial\theta}\log\big(p_\theta(y|x)\big) = \mathbb{E}_{p_\theta(z|x,y)}\left[\frac{\partial}{\partial\theta}\log(p_\theta(y,z|x)\right]$$

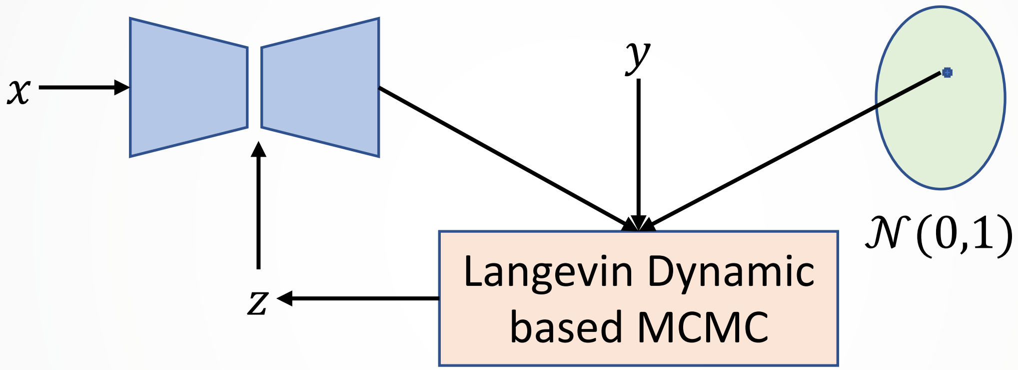
where the expectation term can be approximated with Langevin dynamics based MCMC (a gradient based MCMC) to sample z from it's true posterior distribution via:

$$z_{t+1} = z_t + \frac{s^2}{2}\left[\frac{\partial}{\partial z}\log p_\theta(y, z_t|x)\right] + s\mathcal{N}(0,1)$$

With: $$\frac{\partial}{\partial z}\log p_\theta(y, z_t|x) = \frac{1}{\sigma^2}(y - f_\theta(x,z))\frac{\partial}{\partial z} f_\theta(x,z) - z$$

t : time step for Langevin sampling $\quad$ s : step size

- ABP for dense prediction



Different from VAE or GAN that involves extra modules (inference model for VAE and discriminator for GAN), ABP sample directly from the true posterior distribution via gradient based MCMC.

Energy-based model

- Latent variable model
  - ➢estimate the distribution of the latent variable
  - ➢Estimate the predictive distribution
- Energy-based model
  - ➢Estimate predictive distribution directly.

Energy-based model solution

- EBM: energy-based model, learns an energy function to assign low energy to in-distribution samples and high energy for others.

- Energy-based model:

$$p_\gamma(y|x) = \frac{p_\gamma(y,x)}{\int p_\gamma(y,x)dy} = \frac{1}{Z(x;\gamma)}\exp\left[-U_\gamma(y,x)\right]$$

$U_\gamma(y,x)$: the energy function

$Z(x;\gamma) = \int \exp\left[-U_\gamma(y,x)\right]dy$: the normalizing constant

A tutorial on energy-based learning. Yann LeCun. 2006

Energy-based model solution

- Energy-based model:

$$p_\gamma(y|x) = \frac{p_\gamma(y,x)}{\int p_\gamma(y,x)dy} = \frac{1}{Z(x;\gamma)} \exp\left[-U_\gamma(y,x)\right]$$

When the energy function U_γ is learned and input image x is given, prediction can be achieved via Langevin sampling: $y \sim p_\gamma(y|x)$:

$$y_{t+1} = y_t - \frac{\sigma^2}{2} \frac{\partial U_\gamma(y_t, x)}{\partial y} + \delta\Delta_t, \Delta_t \sim \mathcal{N}(0,1)$$

Cooperative Training of Descriptor and Generator Networks. Xie, Jianwen et. al. TPAMI. 2018

Energy-based model solution

- EBM: 1) start point of Langevin sampling, 2) train the energy function U_γ

Start point:

1) Any deterministic model $f_\theta(x)$
2) Any latent variable model $f_\theta(x, z)$

Learn U_γ : maximum likelihood estimation

$$\Delta\gamma \approx \frac{1}{N}\sum\nolimits_{i=1}^{N} \frac{\partial}{\partial\gamma} U_\gamma(f_\theta(x_i), x_i) - \frac{1}{N}\sum\nolimits_{i=1}^{N} \frac{\partial}{\partial\gamma} U_\gamma(y_i, x_i)$$

Outline

- Motivation
- Background
- Aleatoric uncertainty and Epistemic Uncertainty
- Uncertainty Approximation
  - ✓ Ensemble solutions
  - ✓ Generative model solutions
  - ✓ Bayesian latent variable model solutions
- Experiments
- Discussion and Conclusion

Bayesian latent variable model solutions

- Bayesian Neural Network
- Latent variable model

Predictive distribution:

$$p(y|x) = \int p_\theta(y|x,z)p(\theta)p(z)d\theta dz$$

Regression Likelihood:

$$p_\theta(y|x,z) = \mathcal{N}(f_\theta(x,z),\Sigma)$$

Classification Likelihood:

$$p_\theta(y|x,z) = Softmax(f_\theta(x,z)/\exp(\sigma^2))$$

Uncertainty origin: $\underbrace{(\epsilon, \Sigma), z}$, $p(\theta)$

Aleatoric uncertainty

Epistemic uncertainty

Generative model based Uncertainty computation—regression

- Aleatoric uncertainty $U_a = \frac{1}{T}\sum_{t=1}^{T} H(y|x,\theta_t)$, or $U_a = \sigma(x)^2$
- Epistemic uncertainty

$$U_e = \mathbb{E}_{p(\theta|D)}[p(y|x,\theta)^2] - (\mathbb{E}_{p(\theta|D)}[p(y|x,\theta)])^2$$

or

$$U_e = H\left(\frac{1}{T}\sum_{t=1}^{T} p(y|x,\theta_t)\right) - \frac{1}{T}\sum_{t=1}^{T} H(y|x,\theta_t)$$

- Predictive uncertainty

$$U_p = U_a + U_e$$

Generative model based Uncertainty computation--classification

- Aleatoric uncertainty

$$U_a = \frac{1}{T}\sum_{t=1}^{T} H(y|x,\theta_t), \text{ or } \; U_a = \sigma(x)^2$$

- Predictive uncertainty

$$U_p = H(\frac{1}{T}\sum_{t=1}^{T} p(y|x,\theta_t))$$

- Epistemic uncertainty

$$U_e = H\left(\frac{1}{T}\sum_{t=1}^{T} p(y|x,\theta_t)\right) - \frac{1}{T}\sum_{t=1}^{T} H(y|x,\theta_t)$$

Break

Outline

- Motivation
- Background
- Aleatoric uncertainty and Epistemic Uncertainty
- Uncertainty Approximation
  - ✓Ensemble solutions
  - ✓Generative model solutions
  - ✓Bayesian latent variable model solutions
- Experiments
- Discussion and Conclusion

Experiments--Uncertainty quality measure?

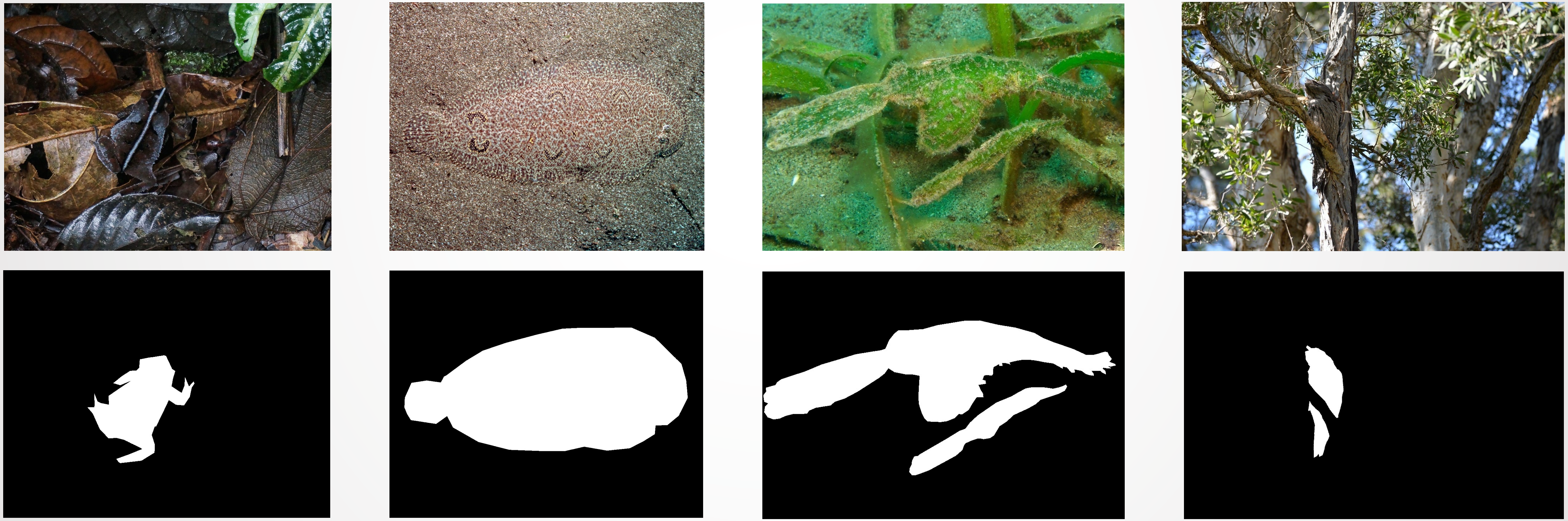
- Expected calibration error
  - ➢On Calibration of Modern Neural Networks. Chuan Guo et. al. ICML. 2017
- Patch accuracy vs patch uncertainty
  - ➢Evaluating Bayesian Deep Learning Methods for Semantic Segmentation. Jishnu Mukhoti and Yarin Gal. Arxiv. 2018
- Evaluation on out-of-distribution samples

Experiments--Tasks

- Camouflaged object detection
- Salient object detection

COD

- Where is the camouflaged object



Camouflaged object detection

TABLE 1
Ensemble based solutions for **camouflaged object detection**. ↑ indicates the higher the score the better, and vice versa for ↓.

Method	CAMO [82]		CHAMELEON [83]		COD10K [71]		NC4K [84]	
	$F_\beta \uparrow$	$\mathcal{M} \downarrow$	$F_\beta \uparrow$	$\mathcal{M} \downarrow$	$F_\beta \uparrow$	$\mathcal{M} \downarrow$	$F_\beta \uparrow$	$\mathcal{M} \downarrow$
Base	.757	.079	.848	.029	.731	.035	.803	.048
MD	.767	.080	.842	.028	.731	.035	.803	.048
DE	.729	.088	.846	.030	.718	.037	.796	.051

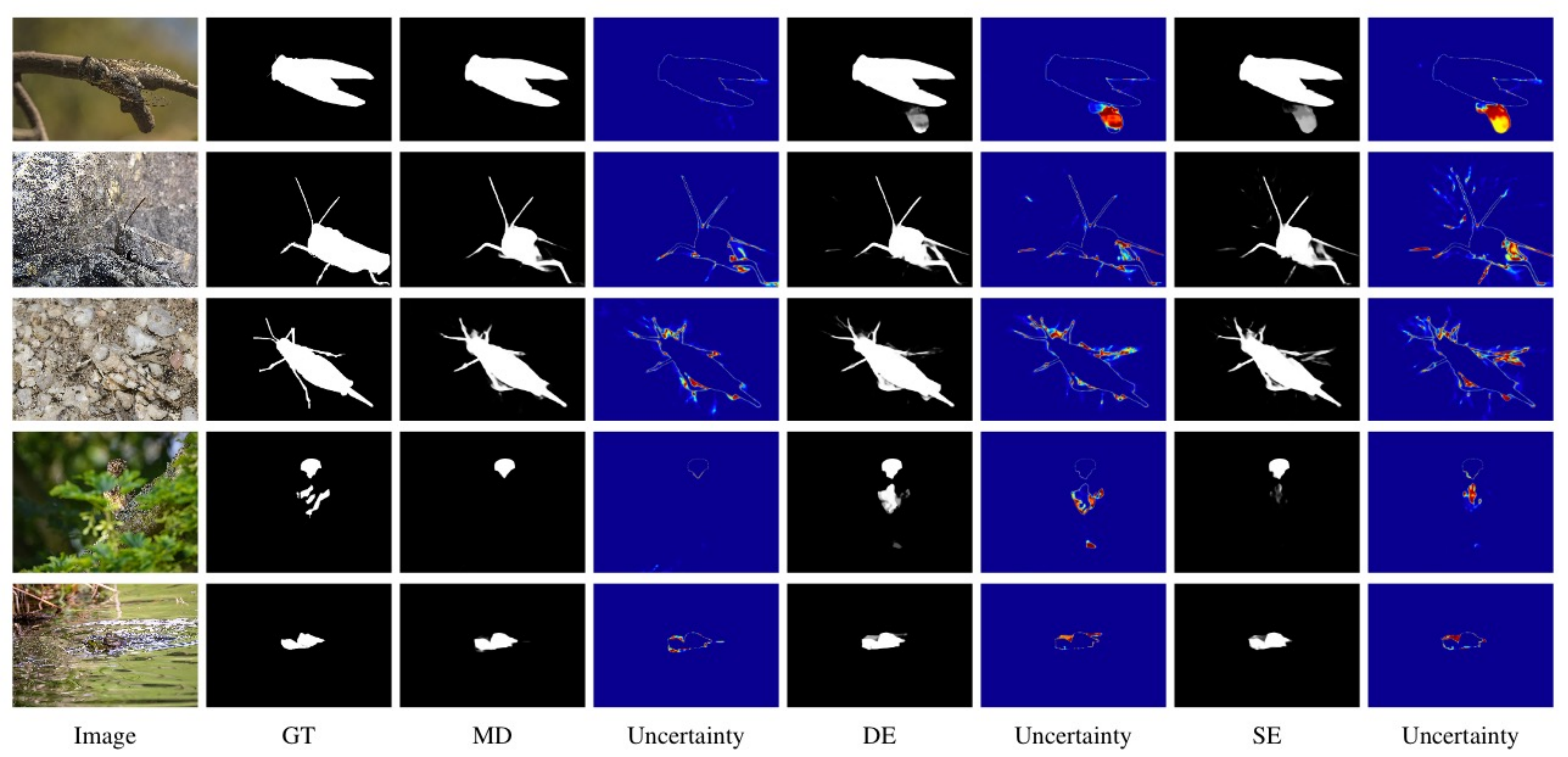
1. Base: the base model
2. MD: MC-dropout
3. DE: deep ensemble

TABLE 3
Generative model based solutions for **camouflaged object detection**. ↑ indicates the higher the score the better, and vice versa for ↓.

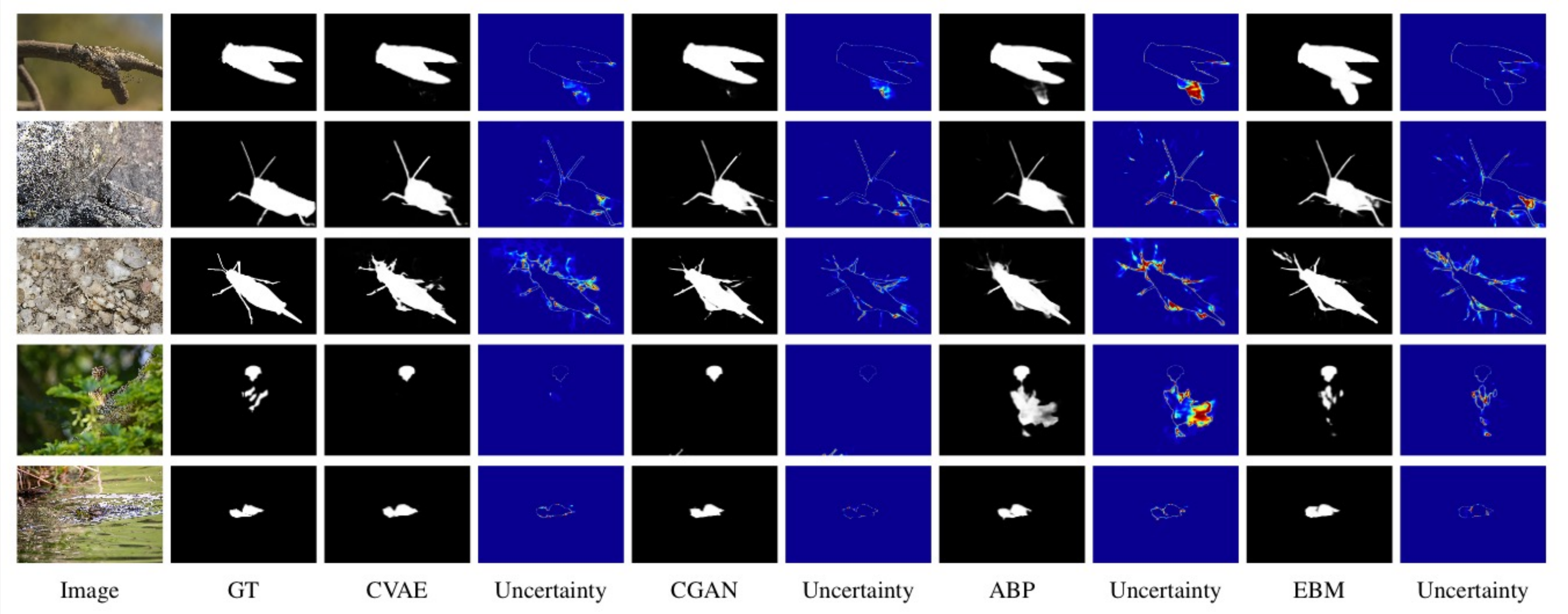
Method	CAMO [82]		CHAMELEON [83]		COD10K [71]		NC4K [84]	
	$F_\beta \uparrow$	$\mathcal{M} \downarrow$	$F_\beta \uparrow$	$\mathcal{M} \downarrow$	$F_\beta \uparrow$	$\mathcal{M} \downarrow$	$F_\beta \uparrow$	$\mathcal{M} \downarrow$
Base	.757	.079	.848	.029	.731	.035	.803	.048
CVAE	.758	.081	.848	.030	.731	.034	.802	.048
CGAN	.762	.080	.852	.026	.730	.034	.807	.048
ABP	.756	.081	.846	.030	.729	.034	.801	.047
EBM	.777	.076	.844	.031	.721	.038	.796	.050

1. Base: the base model
2. CVAE: the CVAE based framework
3. CGAN: the CGAN based framework
4. ABP: the ABP based framework
5. EBM: the EBM based framework

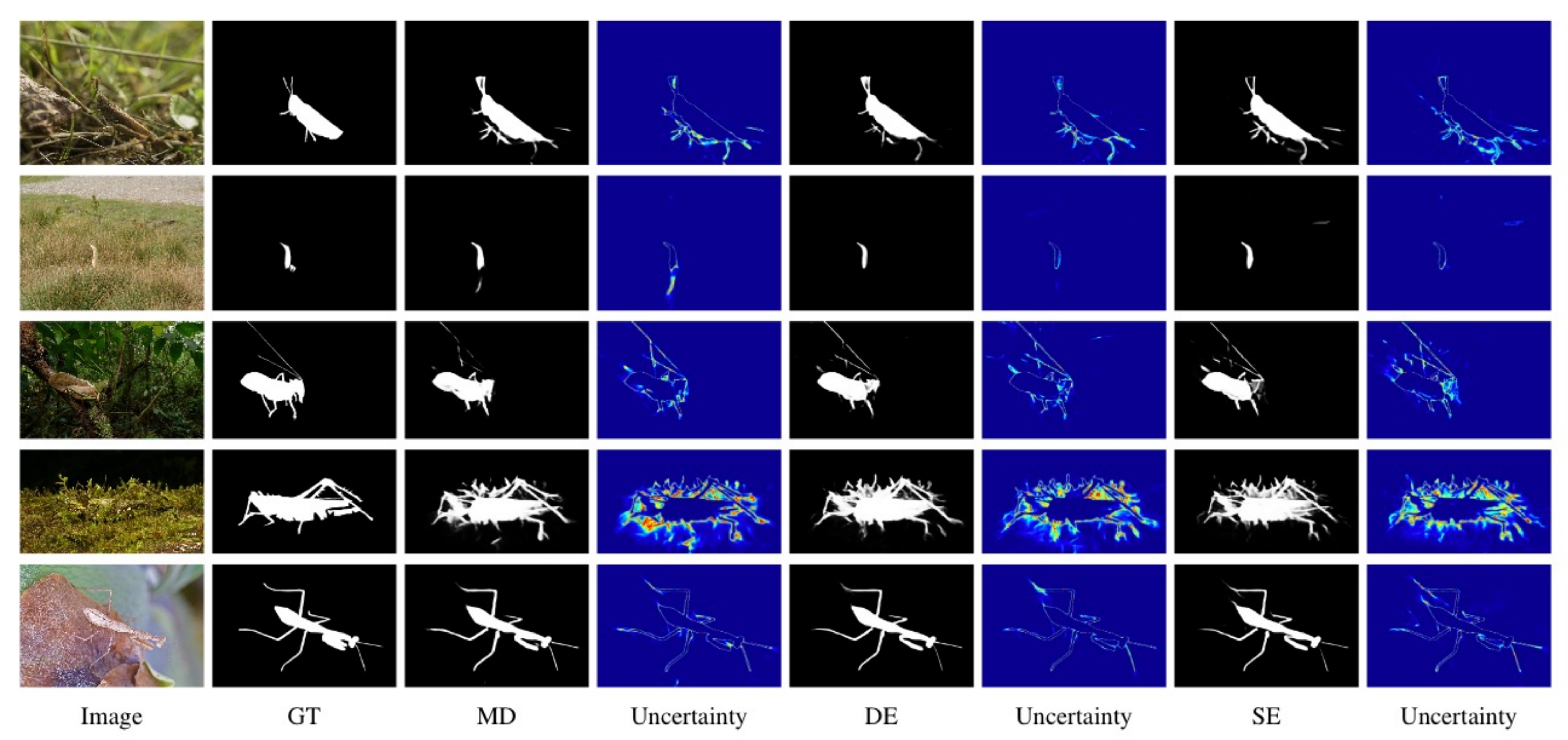
Predictive Uncertainty-Ensemble



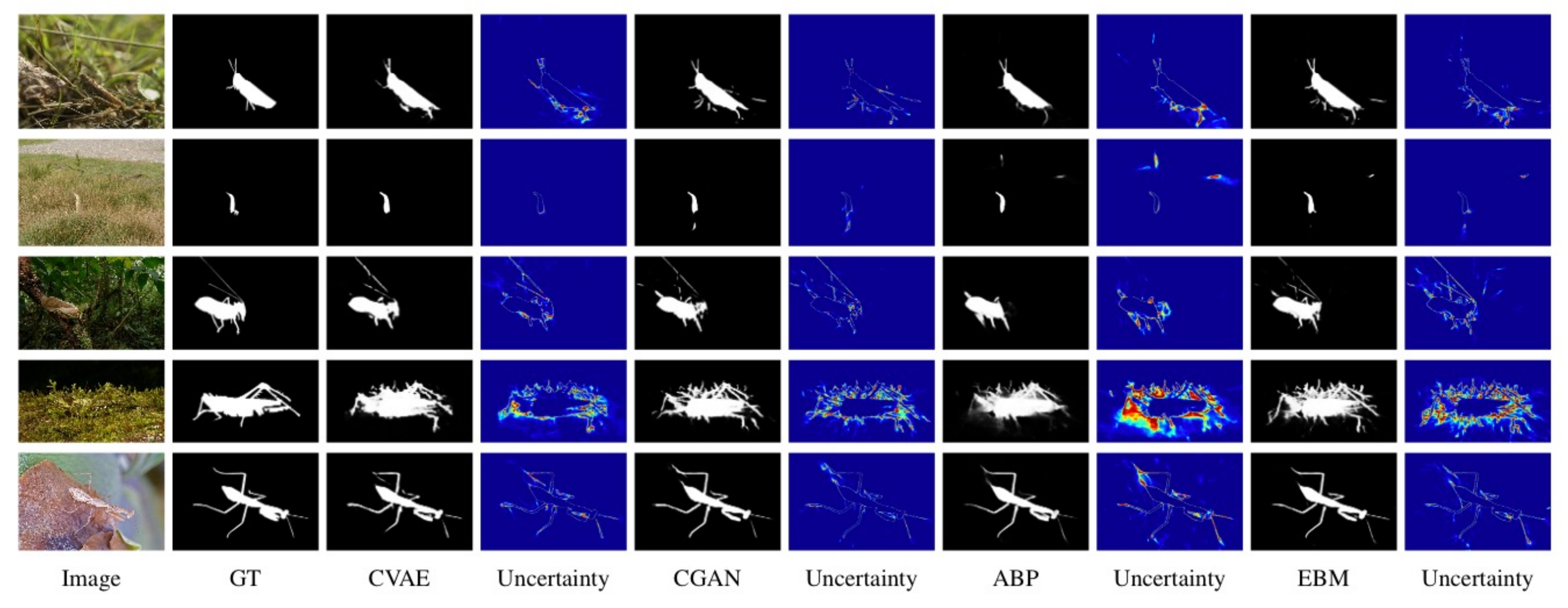
Predictive Uncertainty-Generative Model



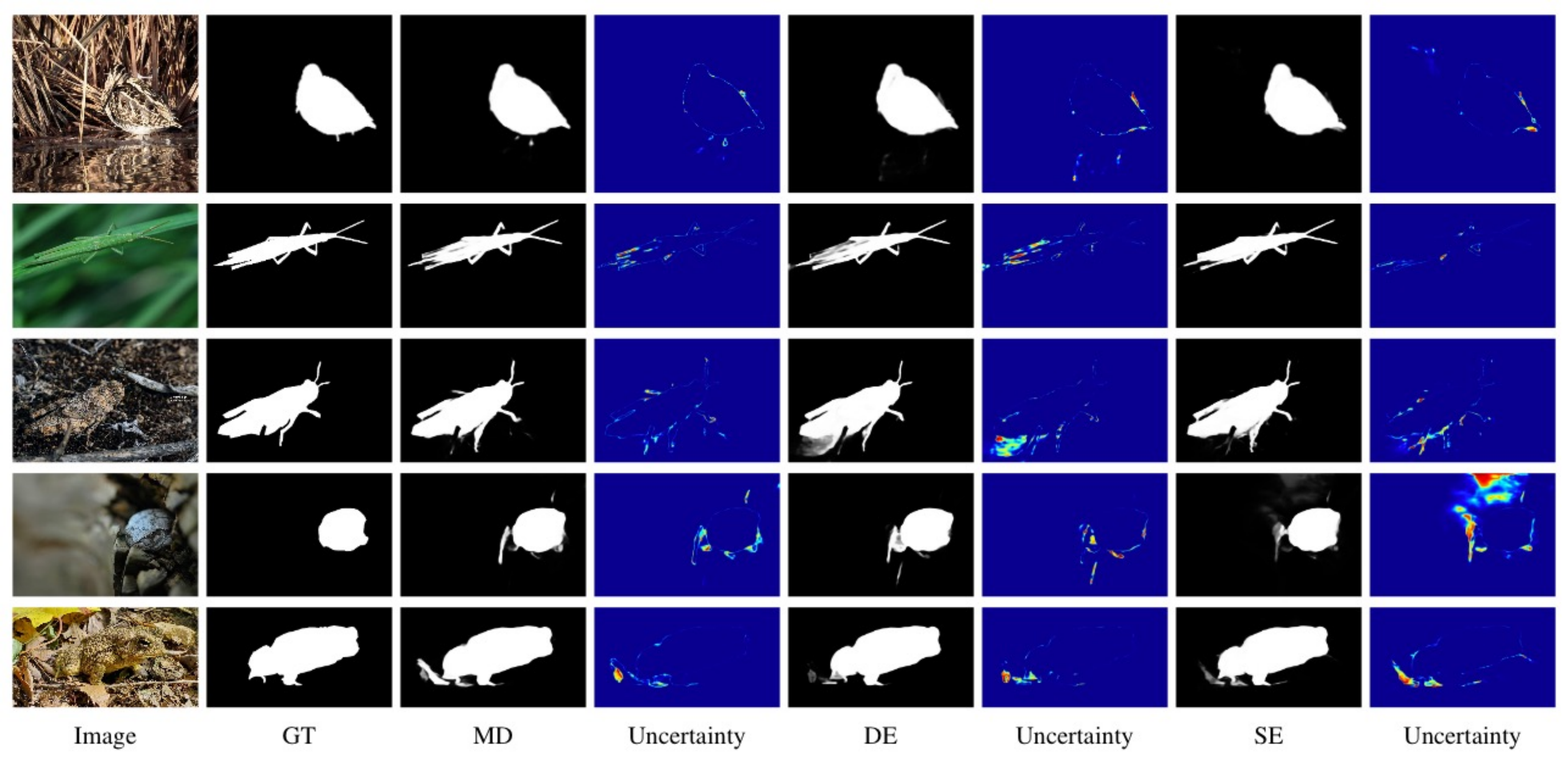
Aleatoric Uncertainty-Ensemble



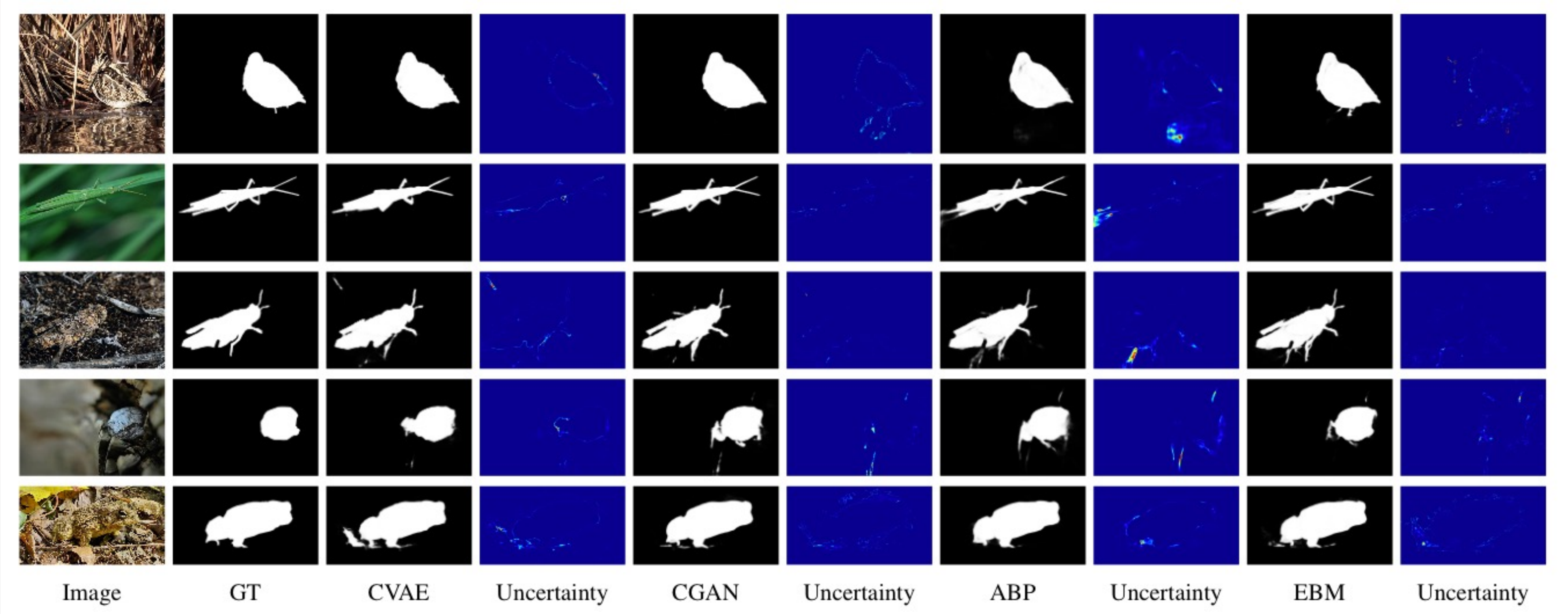
Aleatoric Uncertainty-Generative Model



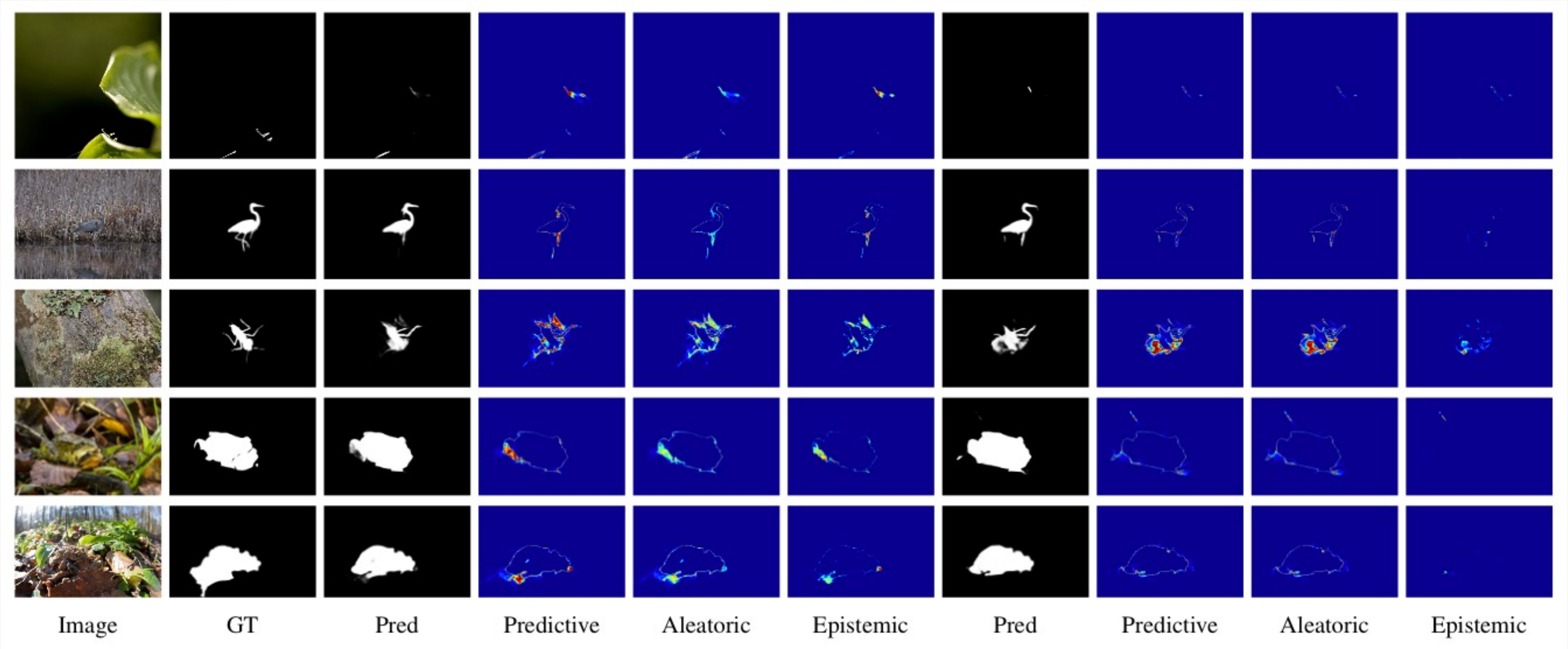
Epistemic Uncertainty-Ensemble



Epistemic Uncertainty-Generative Model



Three types of uncertainty



Observations

- Aleatoric uncertainty focus on inherent noise, i.e. object boundaries
- Epistemic uncertainty highlight challenging pixels, representing our ignorance about which model generated the given dataset.
- COD: aleatoric uncertainty vs epistemic uncertainty

SOD

- Which one is salient?

Salient object detection

TABLE 2
Ensemble based solutions for **salient object detection**. ↑ indicates the higher the score the better, and vice versa for ↓.

Method	DUTS [78]		DUT [79]		HKU-IS [80]		PASCAL [81]	
	$F_\beta \uparrow$	$\mathcal{M} \downarrow$	$F_\beta \uparrow$	$\mathcal{M} \downarrow$	$F_\beta \uparrow$	$\mathcal{M} \downarrow$	$F_\beta \uparrow$	$\mathcal{M} \downarrow$
Base	.842	.037	.760	.055	.904	.030	.828	.064
MD	.854	.036	.763	.056	.911	.028	.840	.061
DE	.828	.040	.738	.061	.897	.031	.825	.065

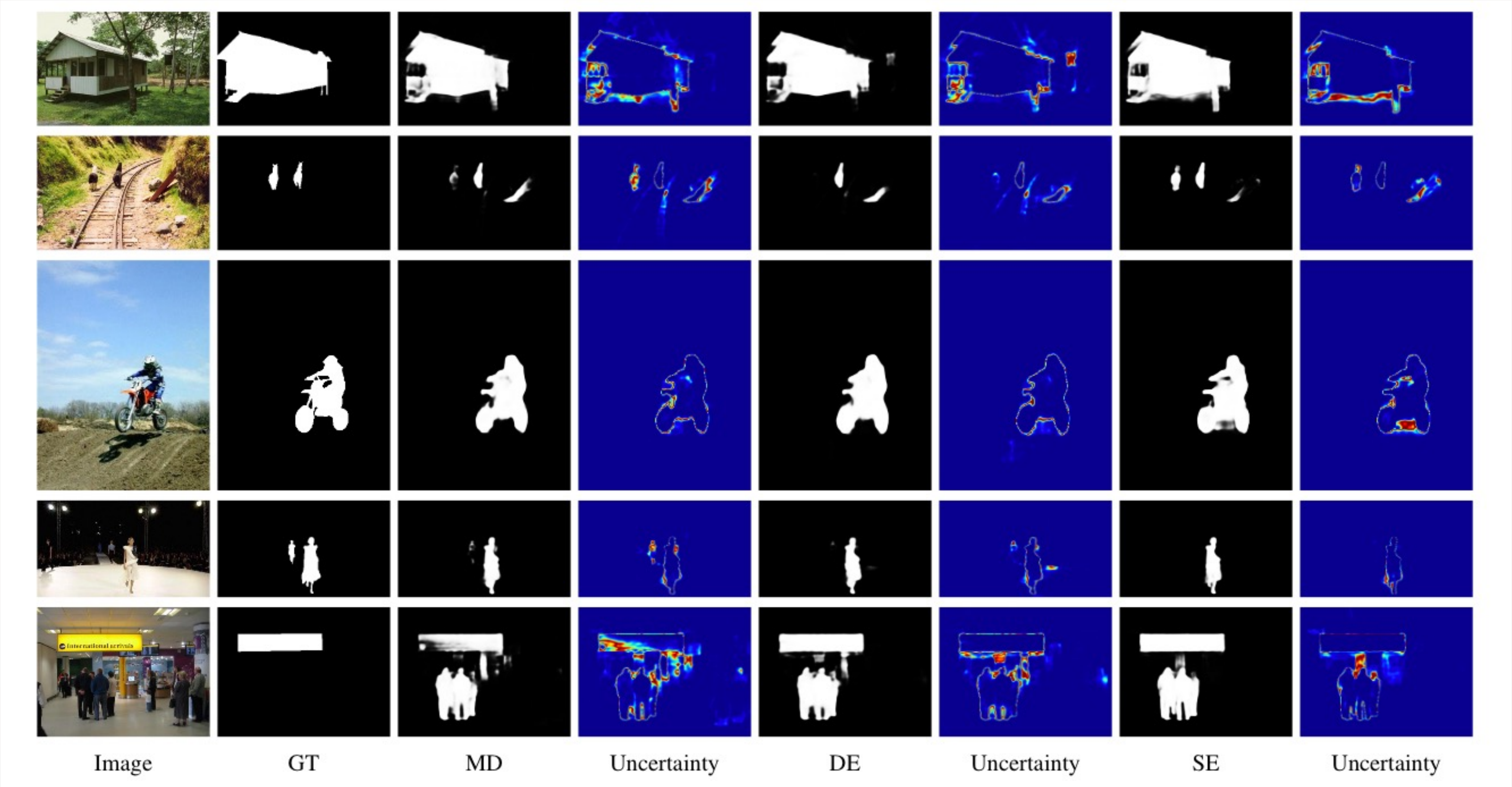
1. Base: the base model
2. MD: MC-dropout
3. DE: deep ensemble

TABLE 4
Generative model based solutions for **salient object detection**, ↑ indicates the higher the score the better, and vice versa for ↓.

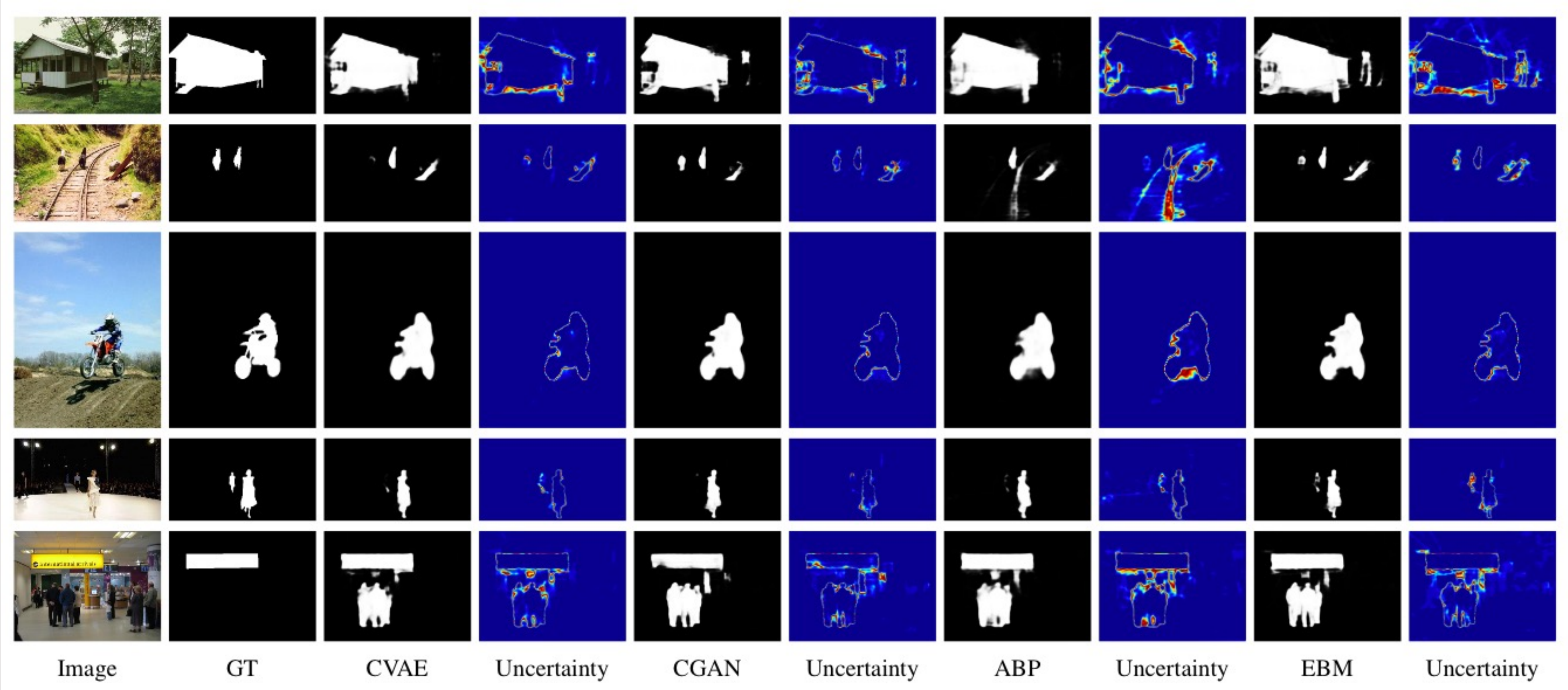
Method	DUTS [79]		DUT [80]		HKU-IS [81]		PASCAL [82]	
	$F_\beta \uparrow$	$\mathcal{M} \downarrow$	$F_\beta \uparrow$	$\mathcal{M} \downarrow$	$F_\beta \uparrow$	$\mathcal{M} \downarrow$	$F_\beta \uparrow$	$\mathcal{M} \downarrow$
Base	.842	.037	.760	.055	.904	.030	.828	.064
CVAE	.836	.037	.748	.055	.901	.030	.826	.063
CGAN	.846	.035	.752	.054	.905	.029	.828	.063
ABP	.829	.040	.740	.059	.889	.034	.818	.068
EBM	.834	.040	.744	.062	.900	.031	.829	.064

1. Base: the base model
2. CVAE: the CVAE based framework
3. CGAN: the CGAN based framework
4. ABP: the ABP based framework
5. EBM: the EBM based framework

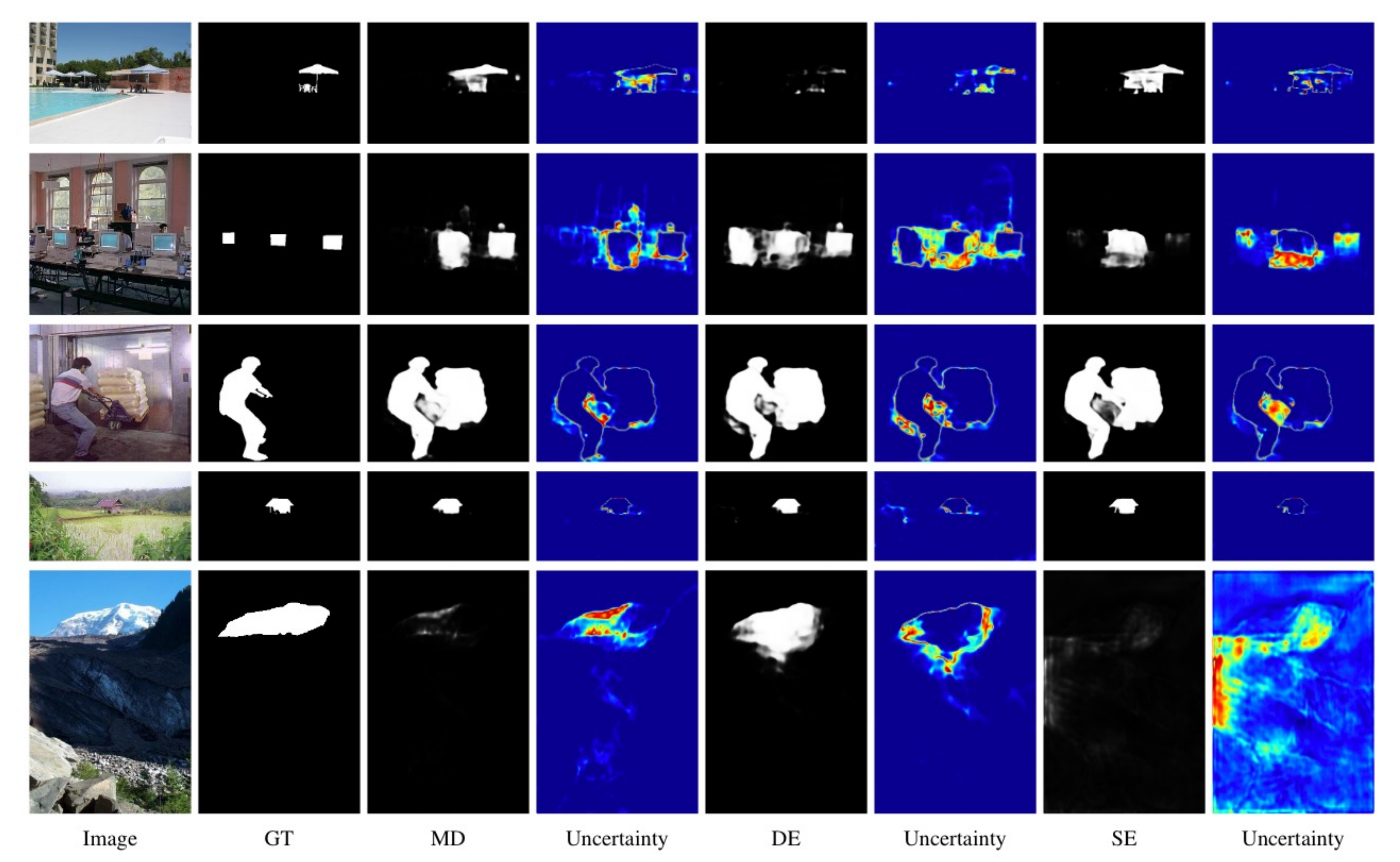
Predictive Uncertainty-Ensemble



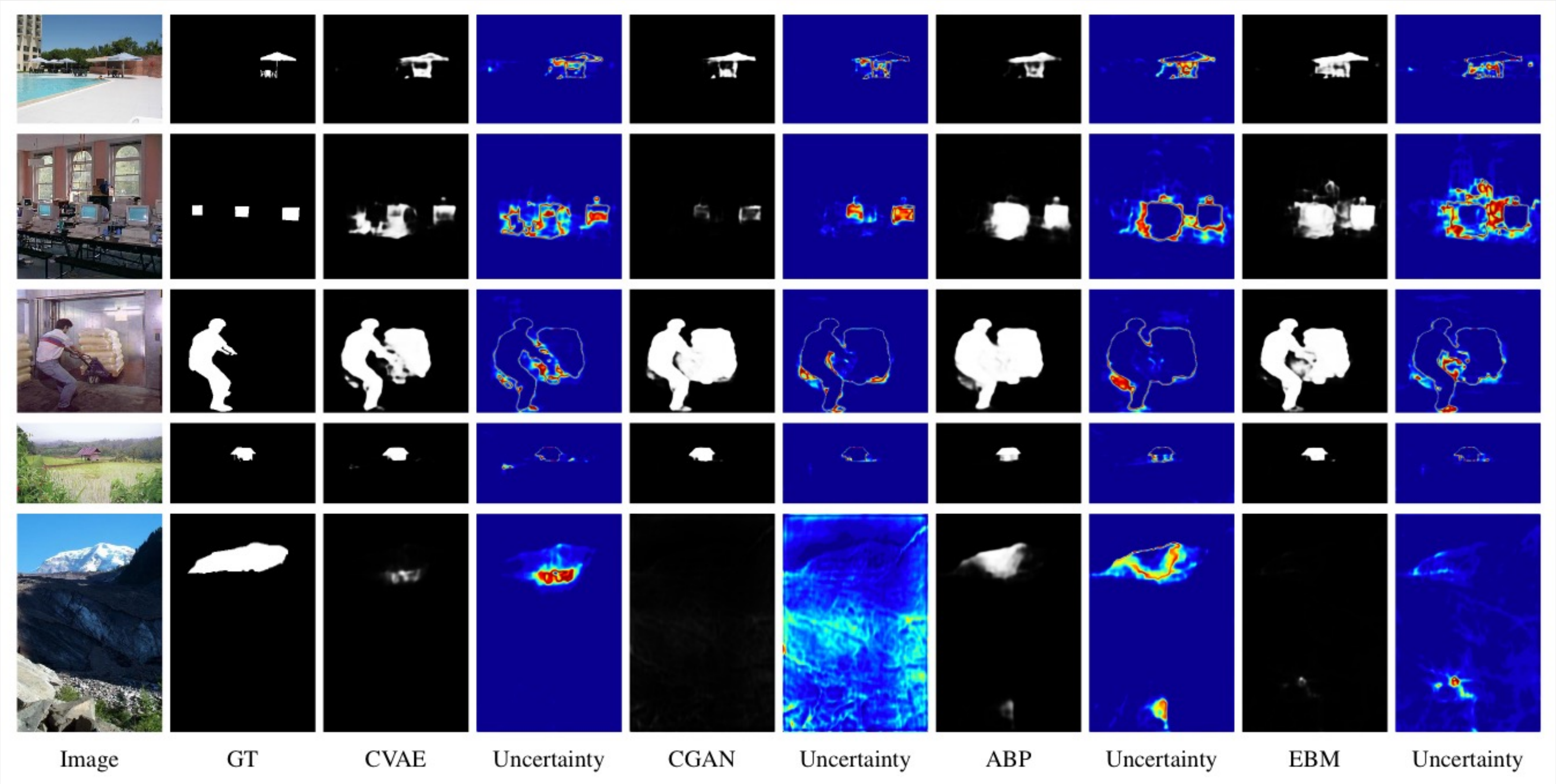
Predictive Uncertainty-Generative Model



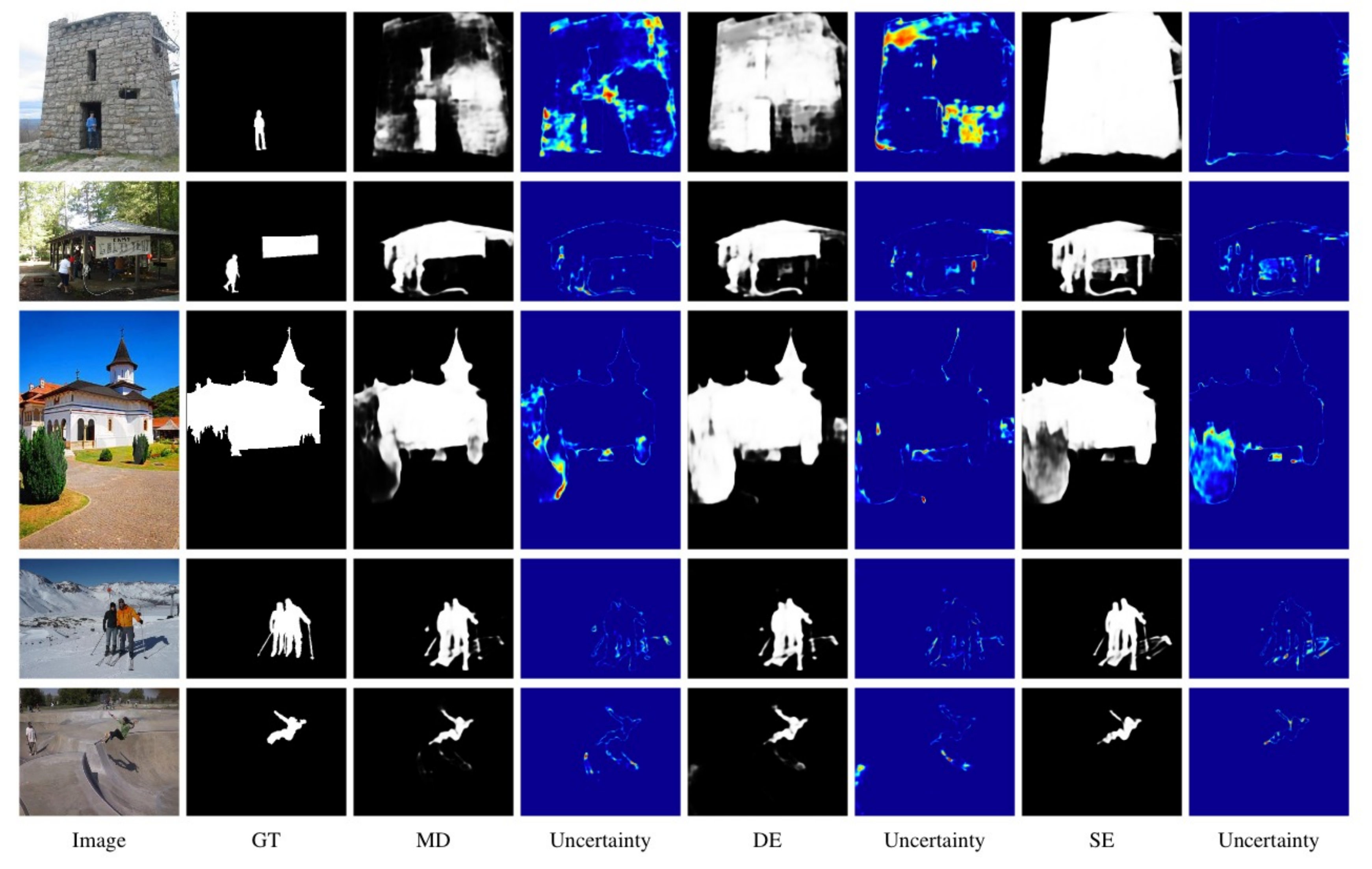
Aleatoric Uncertainty-Ensemble



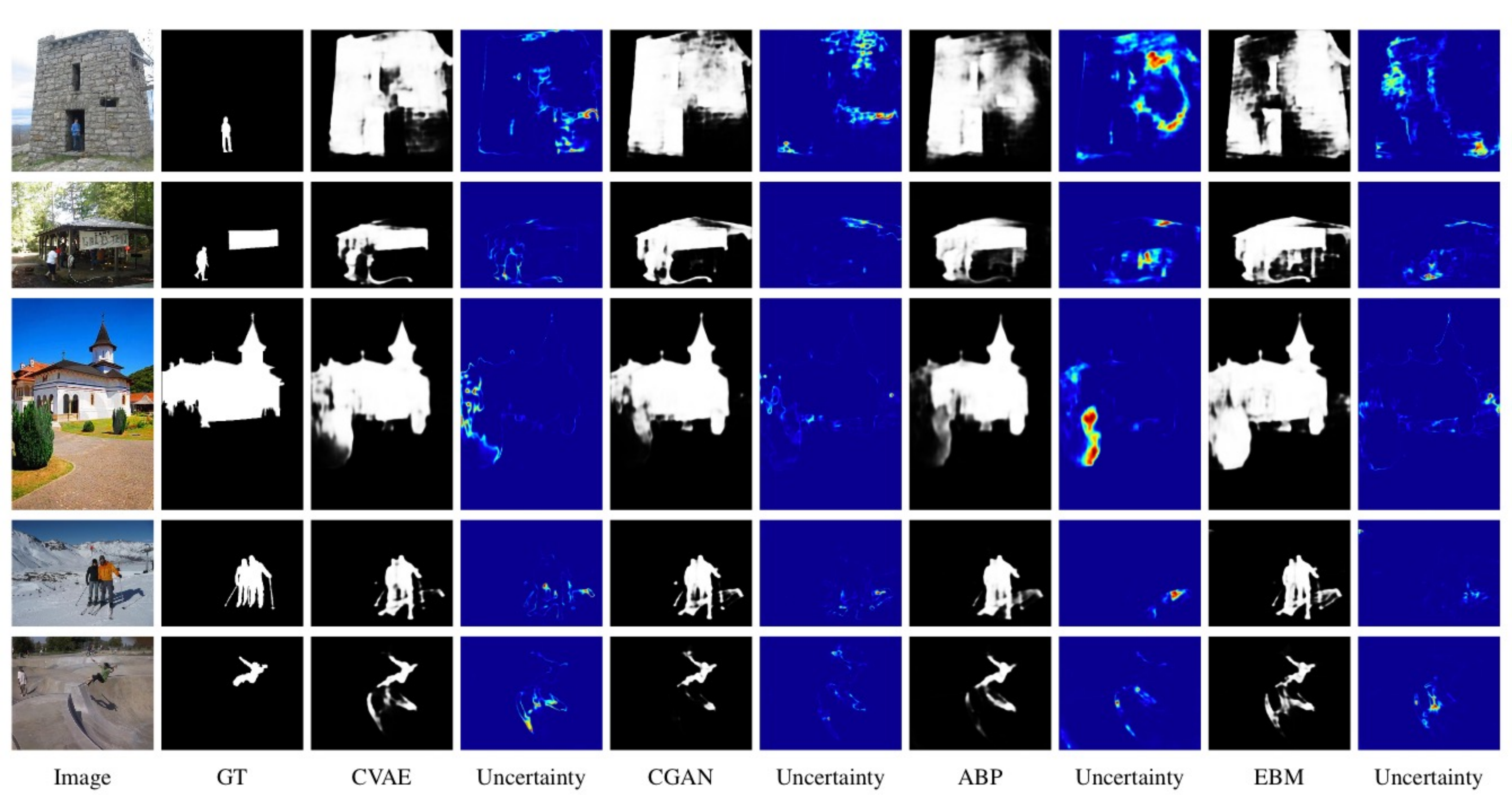
Aleatoric Uncertainty-Generative Model



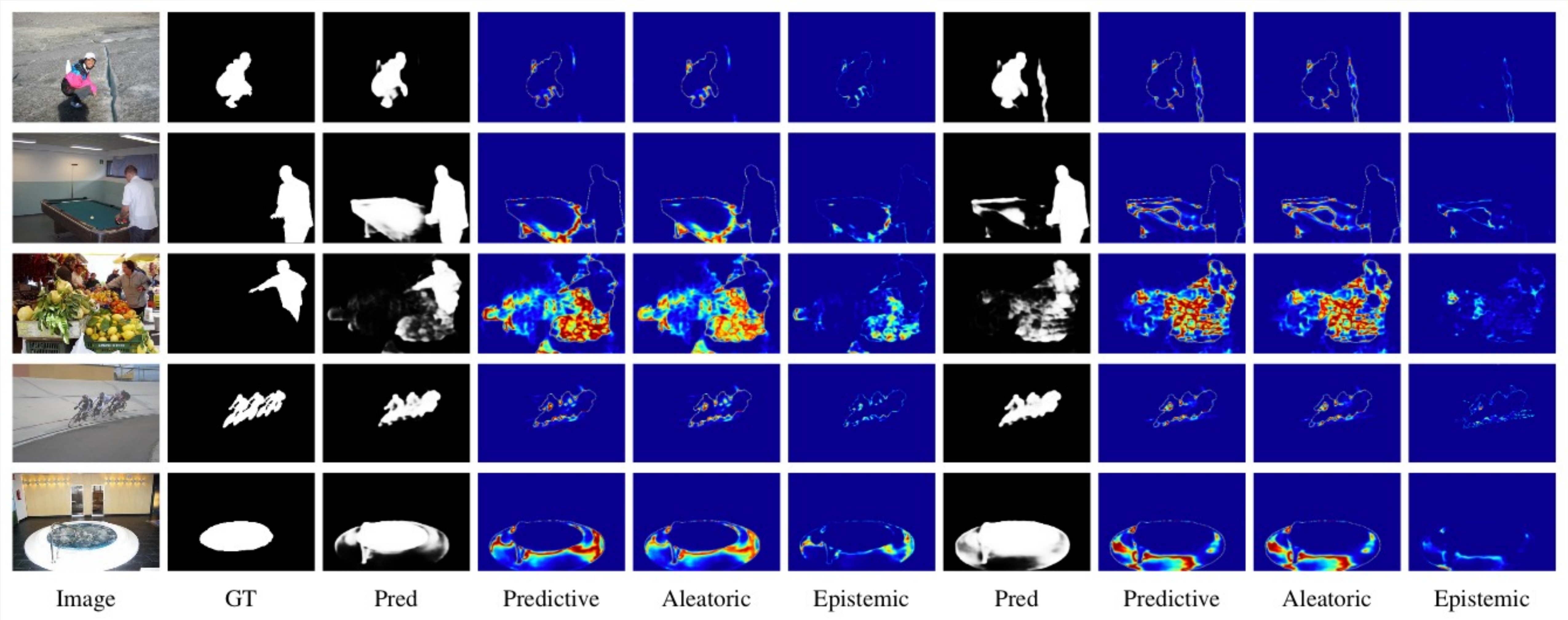
Epistemic Uncertainty-Ensemble



Epistemic Uncertainty-Generative Model



Three types of uncertainty



MC-dropout

CGAN

Observations

- Aleatoric uncertainty focus on inherent noise, i.e. object boundaries
- Epistemic uncertainty highlight challenging pixels, representing our ignorance about which model generated the given dataset.
- SOD: aleatoric uncertainty vs epistemic uncertainty

Discussion

- Sampling-free uncertainty estimation
- Pixel-level uncertainty vs Instance-level uncertainty
- How to effectively use the produced uncertainty
- Model calibration and uncertainty estimation
- Out-of-distribution detection and uncertainty estimation
- Multi-modal/multi-task learning
- Semi-/weakly-supervised learning
- Effectiveness measure

Thanks

Thanks Bushfire CoE for the support.

Contact: zjnwpu@gmail.com

Code and tutorial material are available