Uncertainty modelling within an End-to-end framework for Food Image Analysis

Petia Radeva

Collaboration with:
Eduardo Aguilar, Marc Bolaños,
Bhalaji Nagarajan, Rupali Khatun

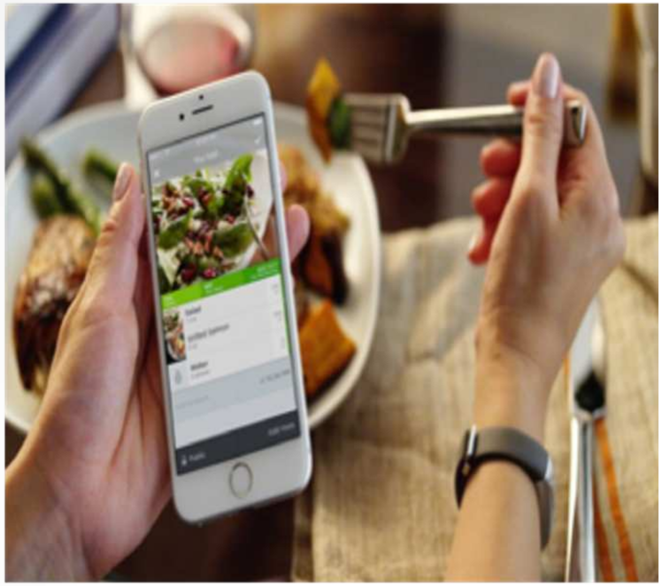
**University of Barcelona &
Computer Vision Center**

petia.ivanova@ub.edu

Contents

- The food image problem
- Multi-task food learning with aleatoric uncertainty
- Food recognition with epistemic uncertainty
- Conclusions

Why food recognition?



"Camera eats first"

180M #food

90/minute

54% take picture

39% post it

Why is the food recognition a challenge?

Motivation

Food Analysis Problems

Ingredients

- Intra-class variability
- Inter-class similarity

Intra-class variability example: Apple. Image source: Recipes5k

Inter-class similarity example: Tomato sauce and Curry sauce. Image source: Recipes5k

Decreasement in Precision

Are we able to recognize thousands of dishes?

- 79% on UECFOOD
- 44% on ChinaFood1000
- How to achieve scalability?

Contents

- The food image problem
- Multi-task food learning with aleatoric uncertainty
- Food recognition with epistemic uncertainty
- Conclusions

Food Analysis as a Multi-task Problem

Cuisine: French.

Categories: Meat.

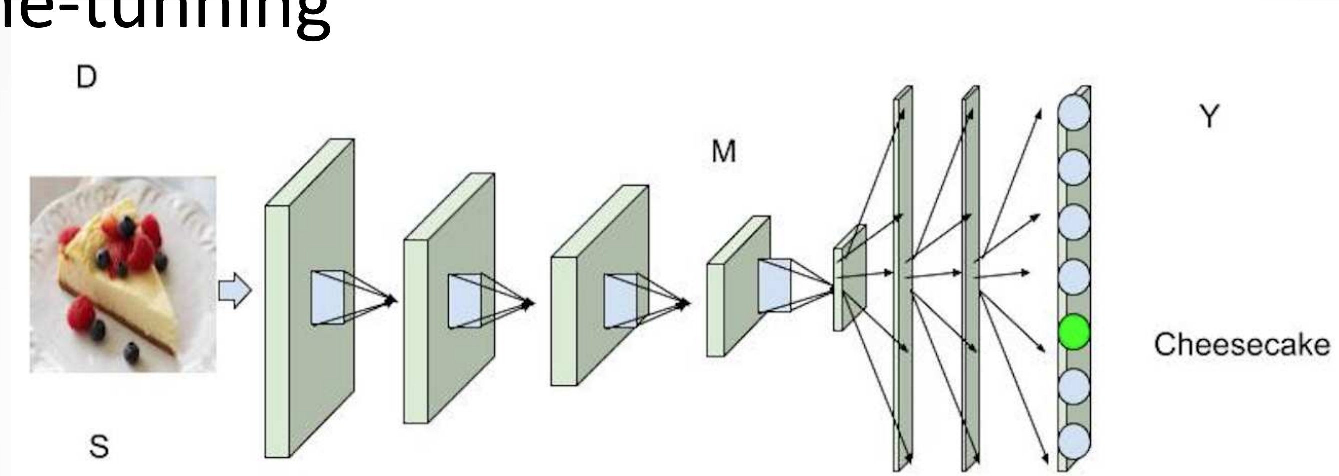
Ingredients: salt, oil, onion, garlic, black pepper, tomato, cloves, parsley, thyme, bay, white wine, clove, duck, fat, mutton.

Dish: Confit de canard.

- Learning **multiple objectives** from a shared representation
  - *Efficiency* and prediction *accuracy*.
- Crucial importance in systems where **long computation** run-time is prohibitive
  - Combining all tasks *reduces computation*.
- Inductive **knowledge transfer**
  - *Generalization* by sharing the domain information between complimentary tasks.

Transfer Learning

Fine-tunning



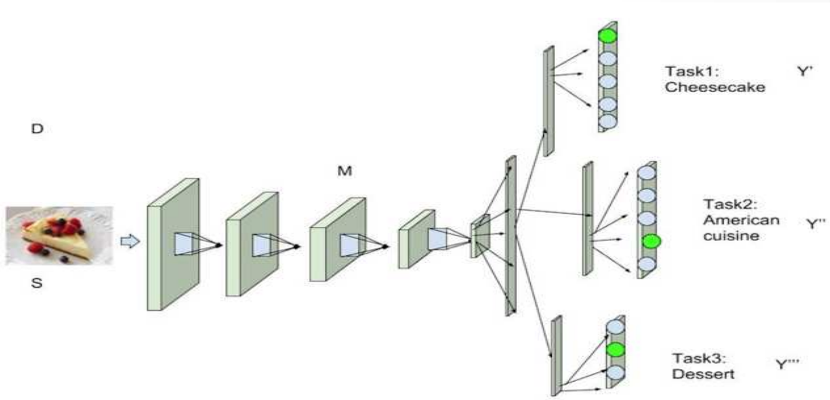
Multi-task learning

Cuisine: French.

Categories: Meat.

Ingredients: salt, oil, onion, garlic, black pepper, tomato, cloves, parsley, thyme, bay, white wine, clove, duck, fat, mutton.

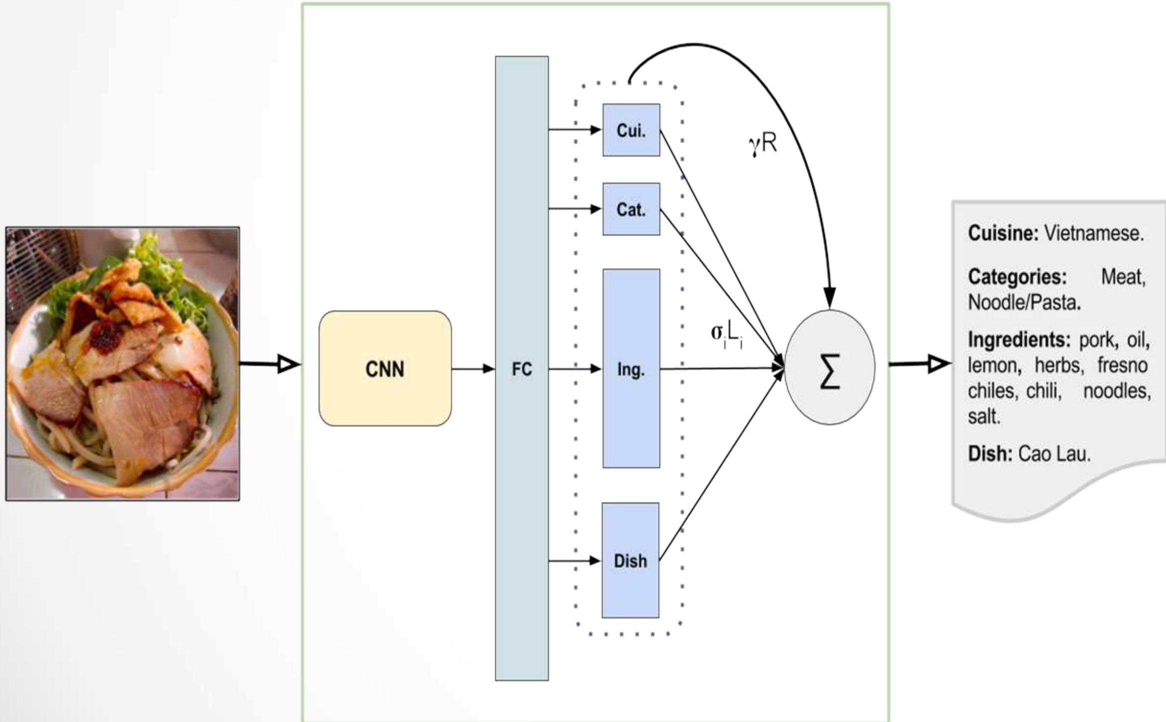
Dish: Confit de canard.



Multi-task FAQ

- How should one pick the right architecture for multi-task learning?
- Does it depend on the final tasks?
- Should we have a completely shared representation between tasks?
- Or should we have a combination of shared and task-specific representations?
- Is there a principled way of answering these questions?

Food Recognition as a MTL



$$L_{total} = \sum_i \omega_i L_i$$

How to define the importance of each task?

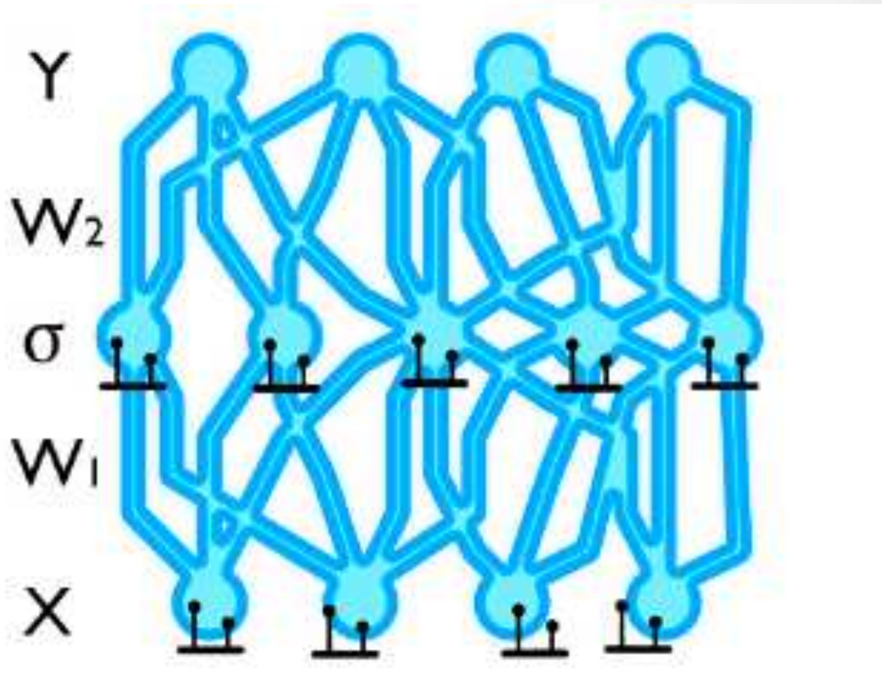
- Weighted uniformly the losses.
- Manually tuned the losses.
- Dynamic weighted of the losses.
    - The main task is fixed and weights are learned for each side-task ([1]).
    - Weight the tasks according to the homoscedastic uncertainty ([2]).

[1] X. Yin and X. Liu. Multi-task convolutional neural network for face recognition.
[2] A. Kendall, Y. Gal, and R. Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics.

Let’s talk about uncertainty

But many unanswered questions...

- Why doesn’t my model work?
- -> Why does my model work?
- We don’t understand many of the tools that we use...
    - E.g. stochastic reg. techniques (dropout) are used in most deep learning models to avoid over-fitting. Why do they work?
- What does my model know?



From Gal’16

But many unanswered questions...

- Why does my model work?
- What does my model know?
- Why does my model predict this and not that?

- **Our models are black boxes and not interpretable...**
- Physicians and others need to understand why a model predicts an output.

Gal'16

Uncertainty in ML

- For Computer scientists, computers and algorithms are **deterministic**.

"Many branches of computer science deal mostly with entities that are entirely deterministic and certain.
Given that many computer scientists work in a relatively clean and certain environment, it can be surprising that machine learning makes heavy use of probability theory."

- The reason that the **answers are unknown** is because of uncertainty.

- The solution is to **systematically evaluate different solutions** until a good or good-enough set of features and/or algorithm is discovered for a specific prediction problem.

https://machinelearningmastery.com/uncertainty-in-machine-learning/

Noise in observations

- Noise refers to **variability or randomness** in the observation.

- The real world, and in turn, real data, is **messy or imperfect**.

    - As practitioners, we must remain skeptical of the data and **develop systems to expect and even harness this uncertainty**.

https://machinelearningmastery.com/uncertainty-in-machine-learning/

Incomplete Coverage of the Domain

- In statistics, a random sample refers to a collection of observations chosen from the domain without systematic bias.
  - **However, there will always be some bias.**

- A suitable level of **variance and bias** in the sample is **required** such that the **sample is representative** of the task or project for which the data or model will be used.

  - Often, we have little control over the sampling process.

https://machinelearningmastery.com/uncertainty-in-machine-learning/

Incomplete Coverage of the Domain

- In all cases, we will never have all of the observations. If we did, a predictive model would not be required.

- This is why we **split a dataset into train and test** sets or use resampling methods like k-fold cross-validation.

    - We do this to handle the uncertainty in the representativeness of our dataset and estimate the performance of a modelling procedure on data not used in that procedure.

https://machinelearningmastery.com/uncertainty-in-machine-learning/

Imperfect Model of the Problem

- This is often summarized as "all models are wrong," or more completely in an aphorism by George Box:

"All models are wrong but some are useful"

- This does not **apply** just to the model, the artifact, but the **whole procedure** used to prepare it, including the choice and preparation of data, choice of training hyperparameters, and the interpretation of model predictions.

https://machinelearningmastery.com/uncertainty-in-machine-learning/

Imperfect Model of the Problem

- Another type of error is an error of omission.

"In many cases, it is more practical to use a simple but uncertain rule rather than a complex but certain one, even if the true rule is deterministic and our modeling system has the fidelity to accommodate a complex rule."

- Given we know that the models will make errors, we handle this uncertainty by seeking a model that is good enough.
  - This often is interpreted as selecting a model that is skillful as compared to a naive method or other established learning models, e.g. good relative performance.

https://machinelearningmastery.com/uncertainty-in-machine-learning/

How to manage Uncertainty

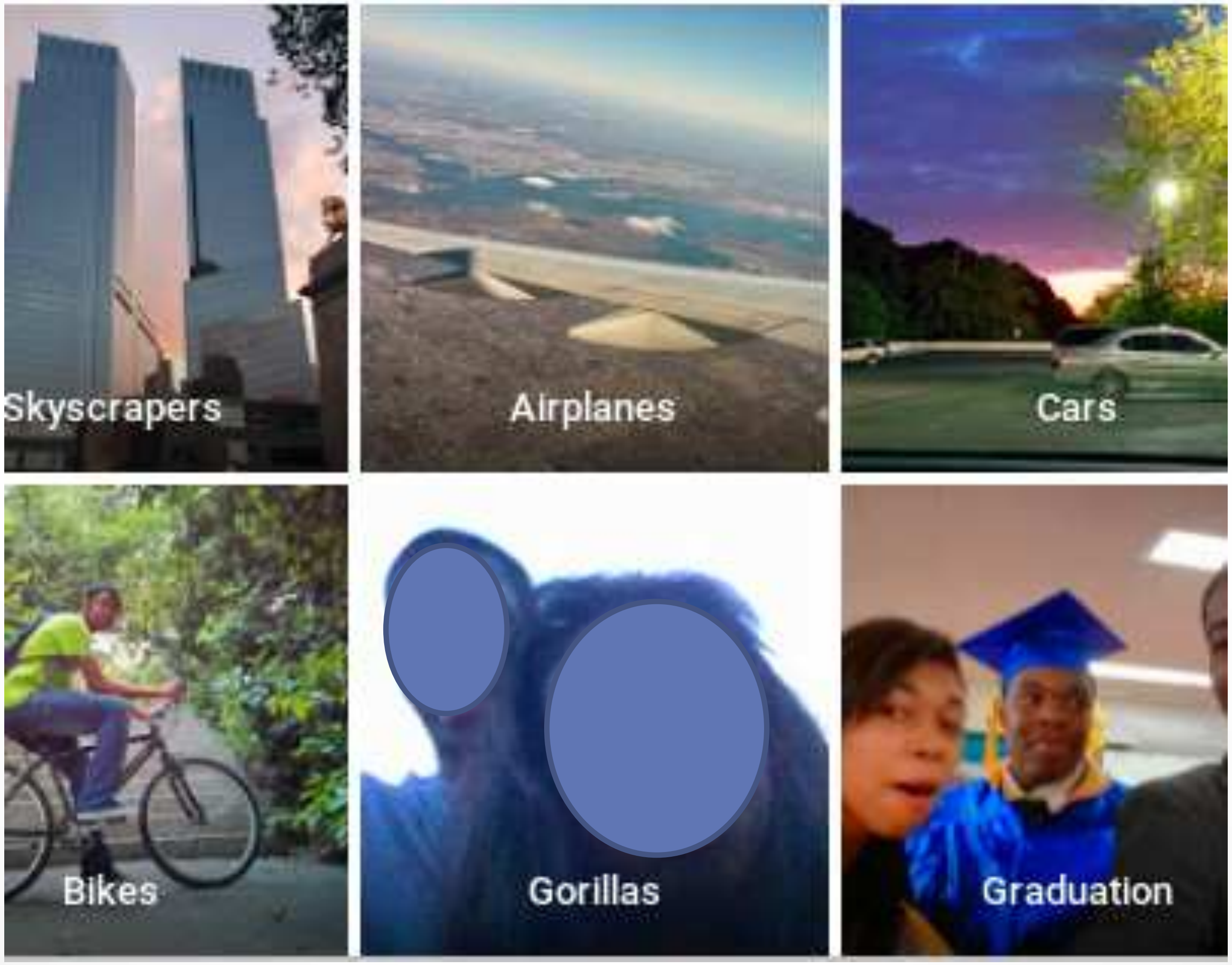
- Probability is the field of mathematics designed to handle, manipulate, and harness uncertainty.
- **In terms of noisy observations**, probability and statistics help us to understand and quantify the expected value, the variability of variables in our observations from the domain.
- **In terms of the incomplete coverage of the domain**, probability helps to understand and quantify the expected distribution and density of observations in the domain.
- **In terms of model error**, probability helps to understand and quantify the expected capability and variance in performance of our predictive models when applied to new data.

https://machinelearningmastery.com/uncertainty-in-machine-learning/

Why uncertainty is important?

Fatal accident of Tesla, May, 2016.

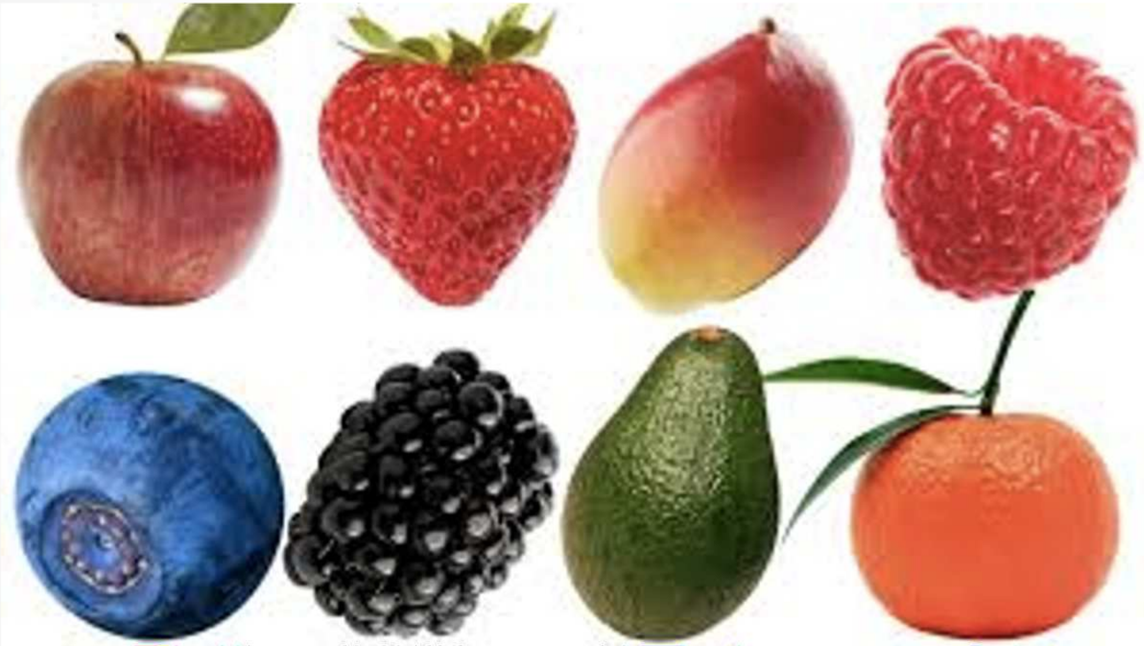
Why uncertainty is important?



Google Photos

Model uncertainty

1. Given a model trained with several pictures of fruits, a user asks the model to decide what is the object using a photo of a chocolate cake.



Who is the guilty for this?

$$S(y_i) = \frac{e^{y_i}}{\sum_j e^{y_j}}$$

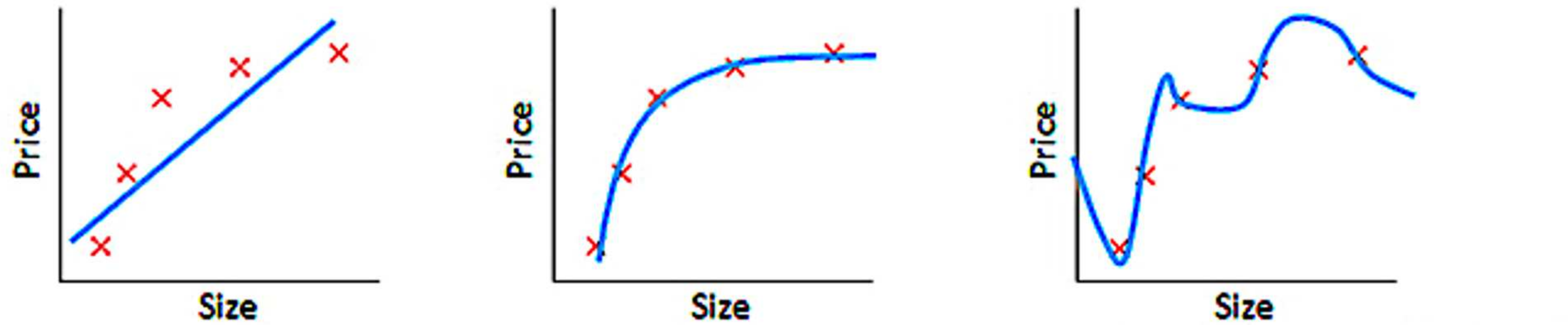
Model uncertainty

2. We have different types of images to classify fruits, where one of the category comes with a lot of clutter/noise/occlusions.

Adapted from Gal (2016)

Model uncertainty

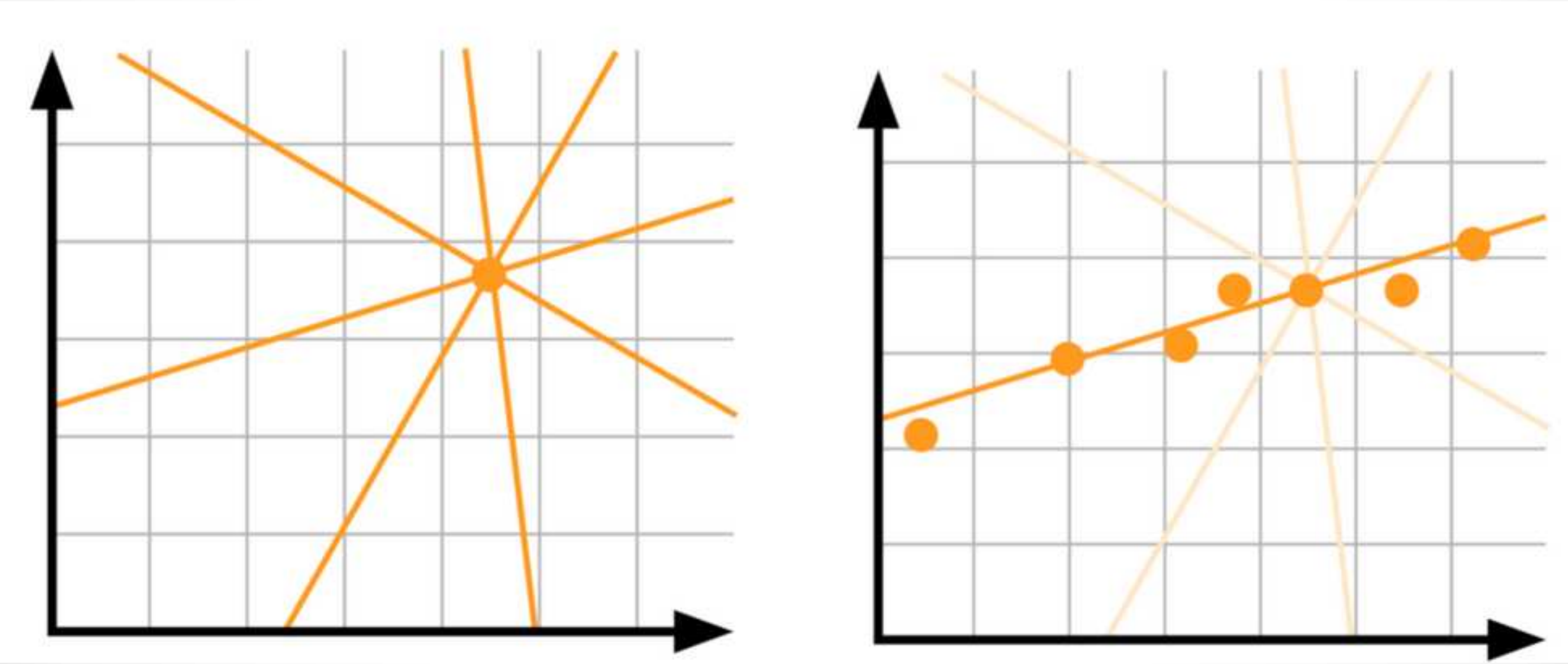
3. What is the best model parameters that best explain a given dataset? What model structure should we use?



$\theta_0 + \theta_1 x$ | $\theta_0 + \theta_1 x + \theta_2 x^2$ | $\theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$

Gal (2016)

Model uncertainty

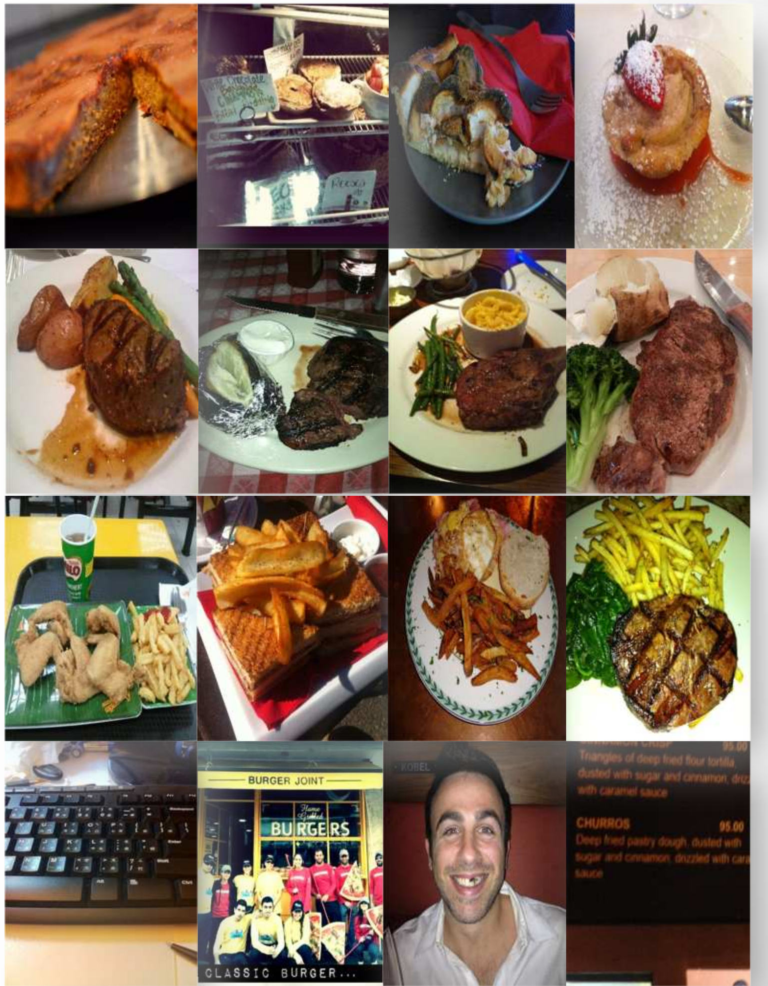


https://engineering.taboola.com/using-uncertainty-interpret-model/

Noisy labels

Noisy labels: with supervised learning we use labels to train the models.

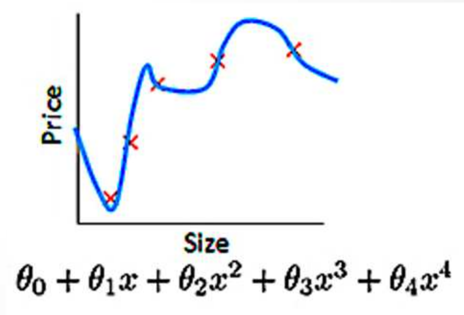
If the labels are noisy, the uncertainty increases.



Types of uncertainty in Bayesian modeling

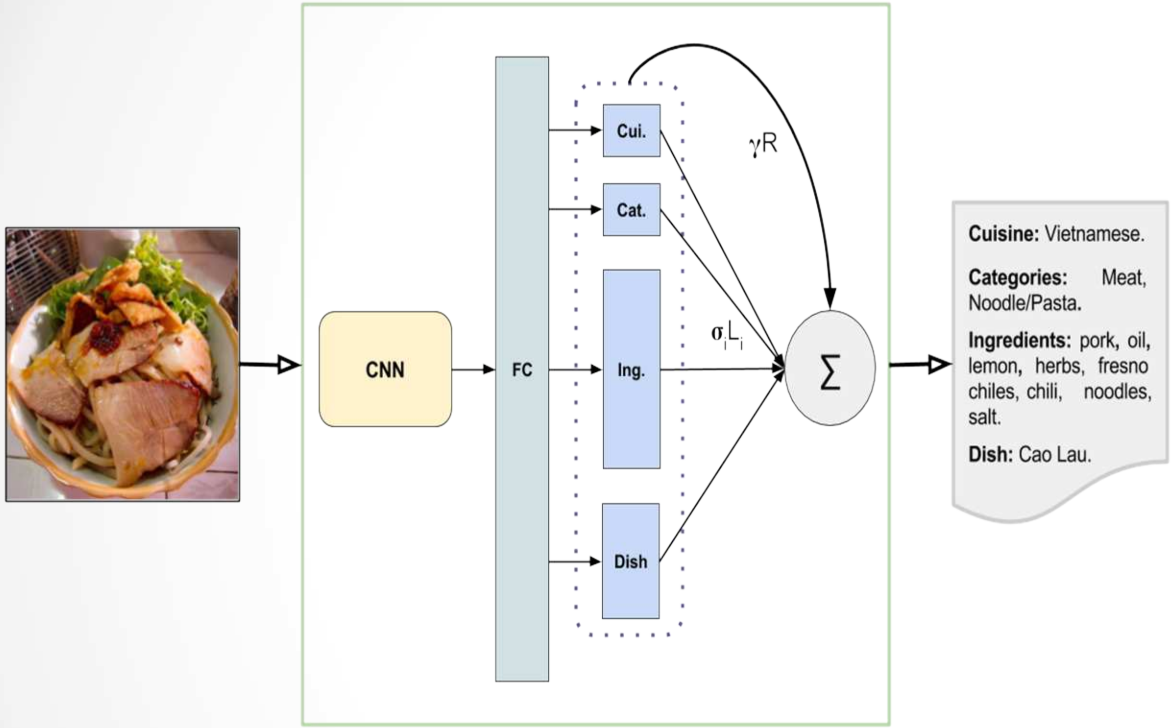
Aleatoric – captures the noise inherent in the observations

- heteroscedastic – data-dependent
- homoscedastic – constant for different data points,
  - but can be task-dependent.
- **Epistemic** – model uncertainty
  - Can be explained away given enough data
  - Uncertainty about the model parameters
  - Uncertainty about the model structure



Food Recognition as a MTL

Aleatoric uncertainty – How to model it?



$$L_{total} = \sum_i \omega_i L_i$$

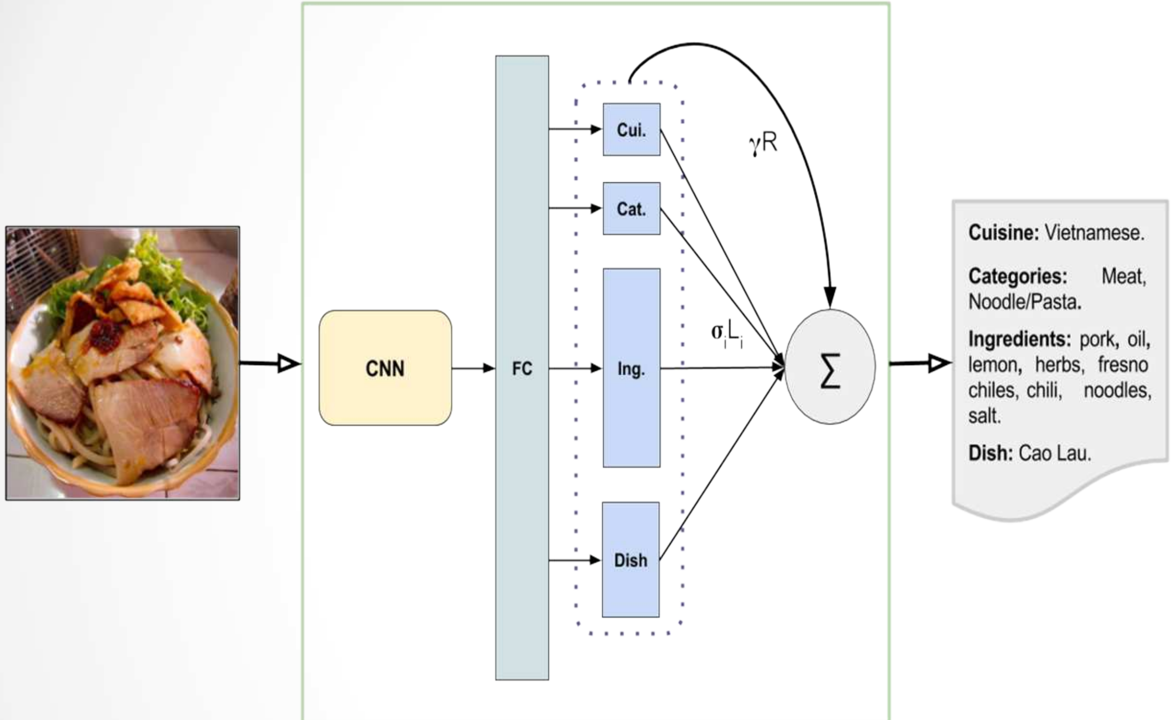
How to determine the total loss of the MTF?

- Expensive to learn & Affects the performance and the efficiency.

Use aleatoric uncertainty modeling to make the model smarter!

Food Recognition as a MTL

Aleatoric uncertainty – How to model it?



$$L_{total} = \sum_i \omega_i L_i$$

- Let us consider a neural network defined on T tasks with model output y and parameters W. Factorizing the output and assuming a Gaussian distribution, we get:

$$p(y_1 \ldots y_T | f^W(x))) = \prod_{i=1}^{T} p(y_i | f^W(x)) = \prod_{i=1}^{T} N(y_i; f^W(x), \sigma_i^2)$$

- Note that σ is a model's observation noise parameter

Multi-task uncertainty-based likelihood

In maximum likelihood inference, we maximize the log likelihood of the model:

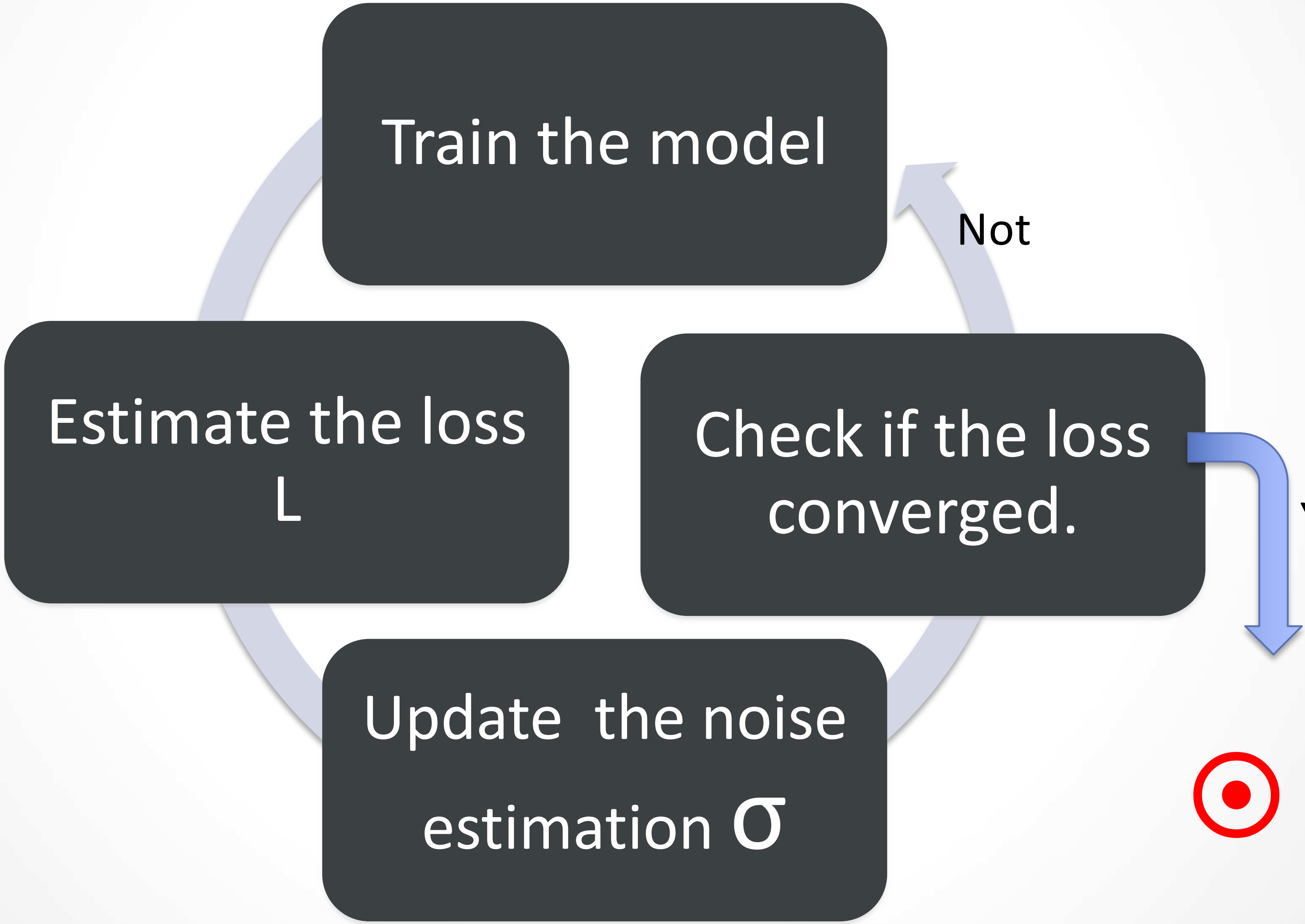
$$L(W,\sigma,...,\sigma) = -\log p(y_1,...y_T|f^W(x))$$

Kendal et.al. (Kendal'2016) showed that:

$$L(W,\sigma,...,\sigma) = -\log p(y_1,...y_T|f^W(x)) \approx \sum_{i=1}^{T}(\frac{1}{2\sigma_i^2}L_i(W) + \log\sigma_i^2)$$

- **Proved that the formula can be extended for the binary cross entropy too (multi-label problems).**

The MTL algorithm



Eduardo Aguilar, Marc Bolaños, Petia Radeva: **Regularized uncertainty-based multi-task learning model for food analysis.** J. Visual Communication and Image Representation 60: 360-370 (2019)

Validation

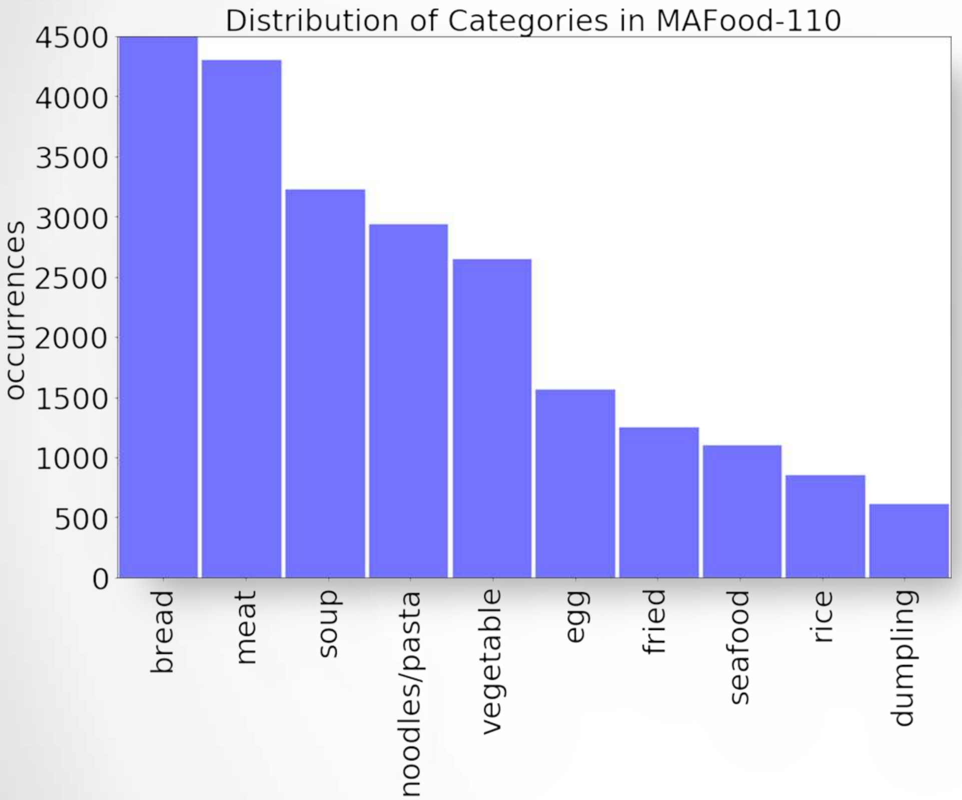
Our Food Dataset

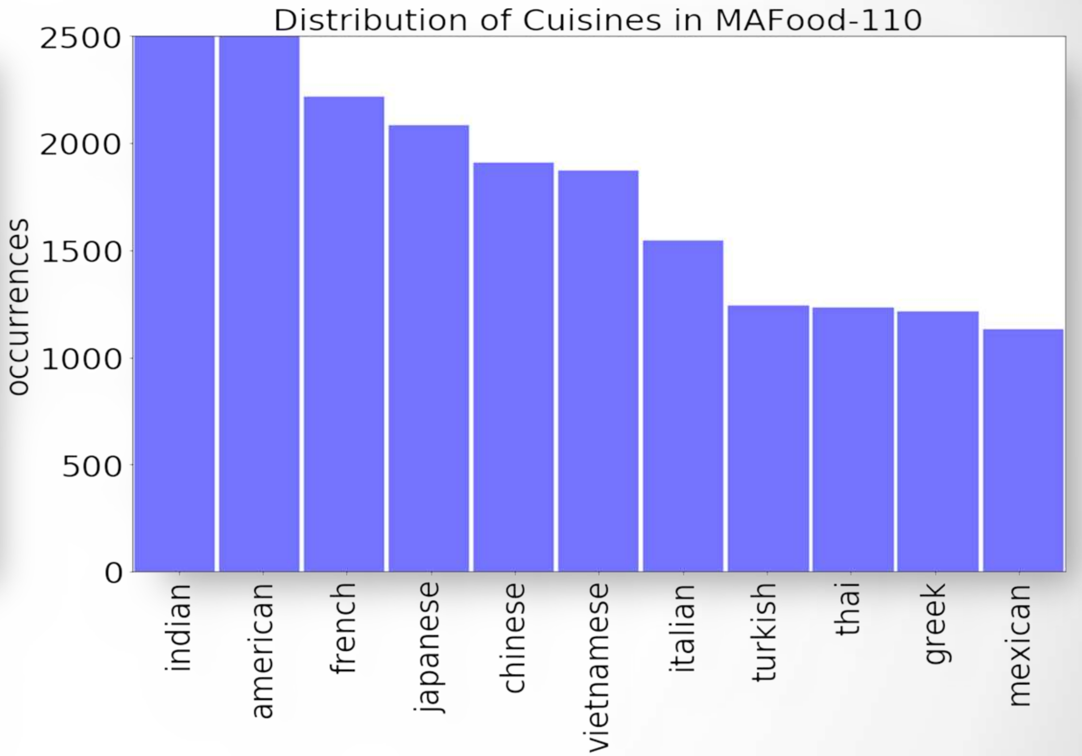
- Food – 550 dishes, 11 categories, 11 cuisines
- Ingredients – 65
- Drinks – 40

In total:
more than
550.000 images

Eduardo Aguilar, Marc Bolaños, Petia Radeva: **Regularized uncertainty-based multi-task learning model for food analysis.** J. Visual Communication and Image Representation 60: 360-370 (2019)

MAFood Data





Dataset available at: www.ub.edu/cvub/dataset

Eduardo Aguilar, Marc Bolaños, Petia Radeva: **Regularized uncertainty-based multi-task learning model for food analysis.** J. Visual Communication and Image Representation 60: 360-370 (2019)

Results

	GT	RUMTL	Single-task
	Dish: tacos	**Dish:** tacos	**Dish:** prime_rib
	Cuisine: mexican	**Cuisine:** mexican	**Cuisine:** american
	Categories: vegetable, meat, bread	**Categories:** vegetable, bread	**Categories:** vegetable, meat

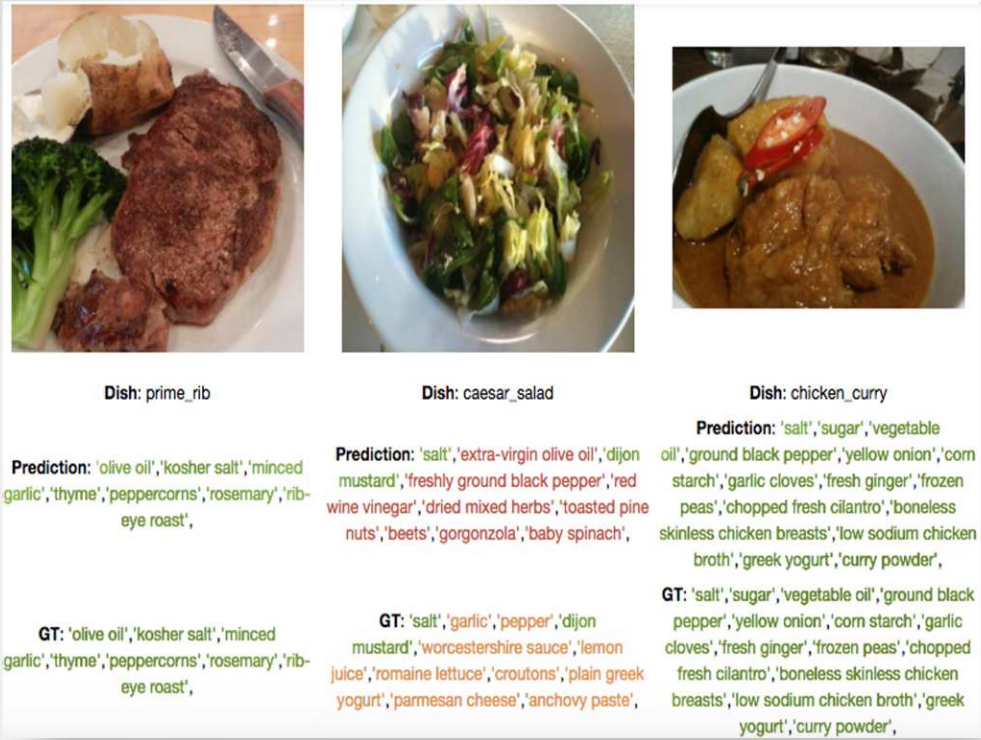
	GT	RUMTL	Single-task
	Dish: eggs_benedict	**Dish:** eggs_benedict	**Dish:** ravioli
	Cuisine: american	**Cuisine:** american	**Cuisine:** italian
	Categories: vegetable, bread, egg	**Categories:** vegetable, bread, egg	**Categories:** vegetable, egg

	GT	RUMTL	Single-task
	Dish: sushi	**Dish:** sushi	**Dish:** cha_ca
	Cuisine: japanese	**Cuisine:** japanese	**Cuisine:** japanese
	Categories: vegetable, seafood, rice	**Categories:** seafood, rice	**Categories:** fried_food

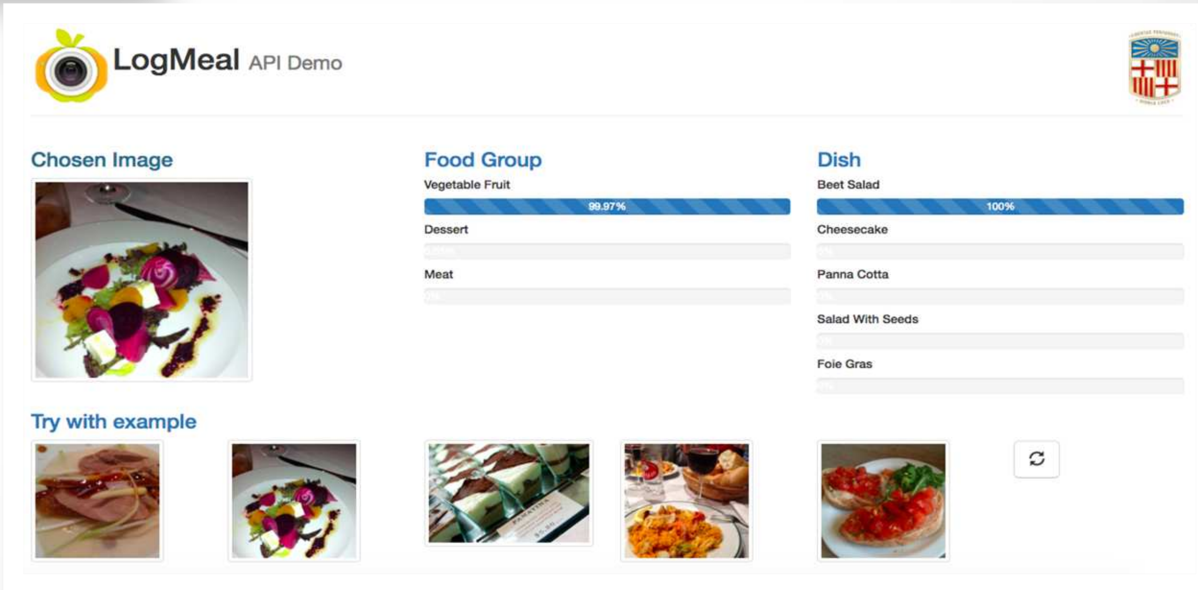
	GT	RUMTL	Single-task
	Dish: ravioli	**Dish:** bruschetta	**Dish:** lobster_roll_sandwich
	Cuisine: italian	**Cuisine:** italian	**Cuisine:** italian
	Categories: dumpling	**Categories:** vegetable, bread	**Categories:** vegetable, meat, bread

Eduardo Aguilar, Marc Bolaños, Petia Radeva: **Regularized uncertainty-based multi-task learning model for food analysis.** J. Visual Communication and Image Representation 60: 360-370 (2019)

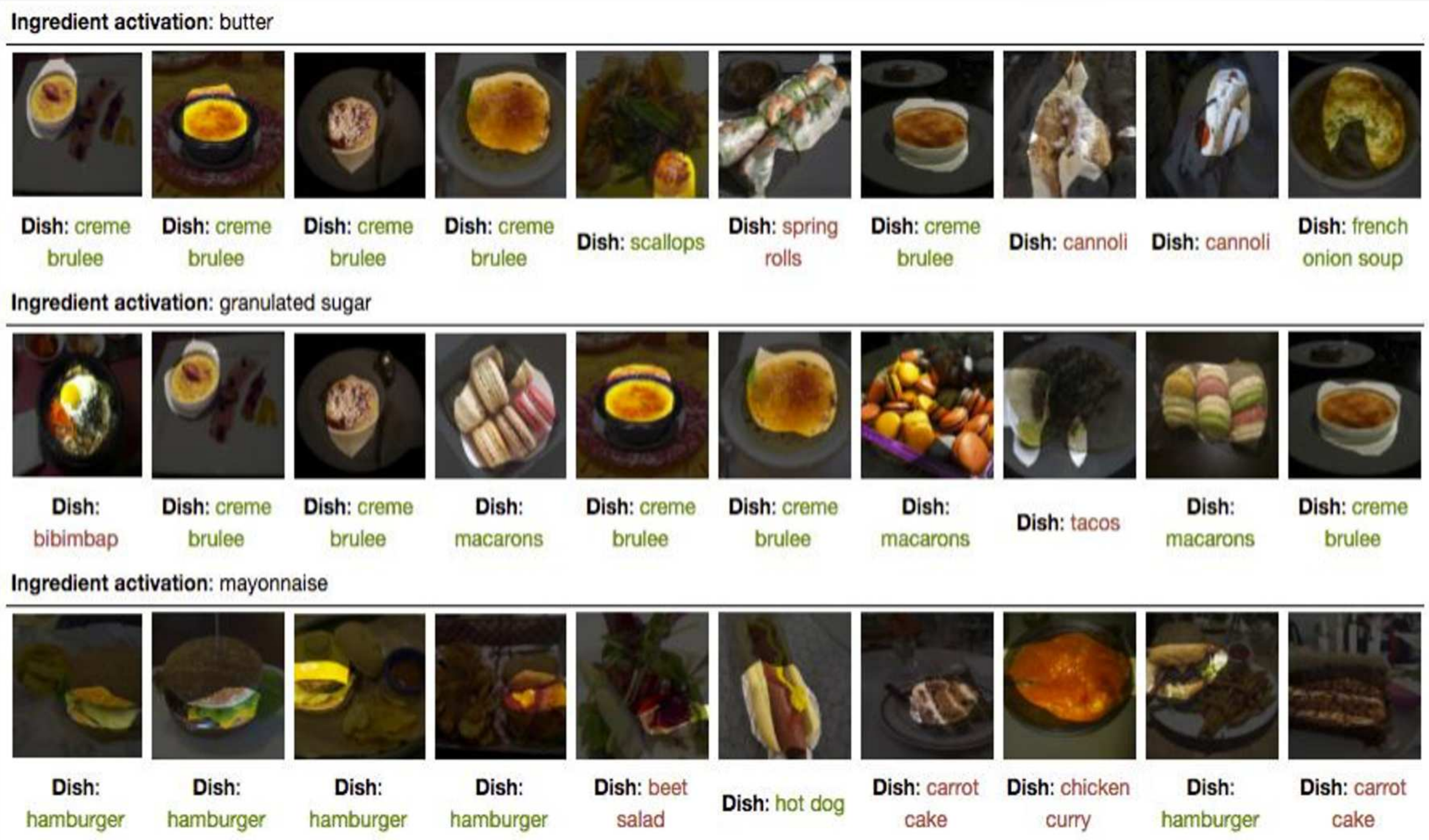
Food ingredients recognition



Food category and class recognition



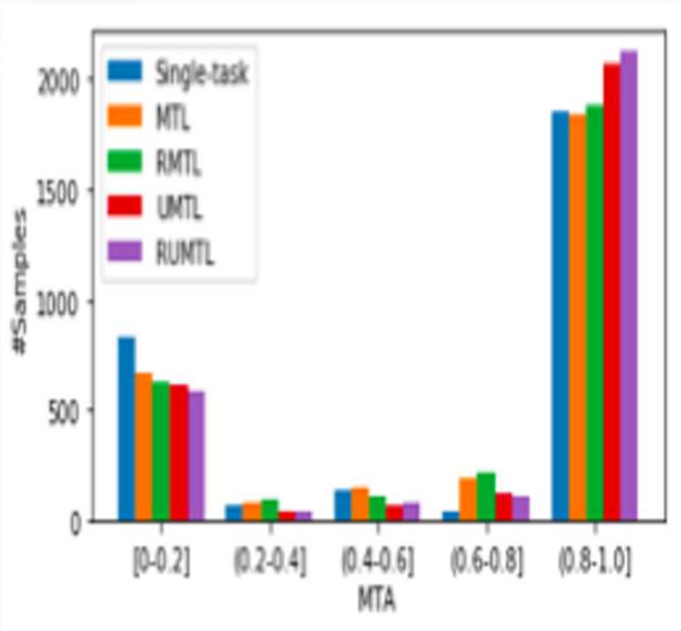
Neurons' Activations

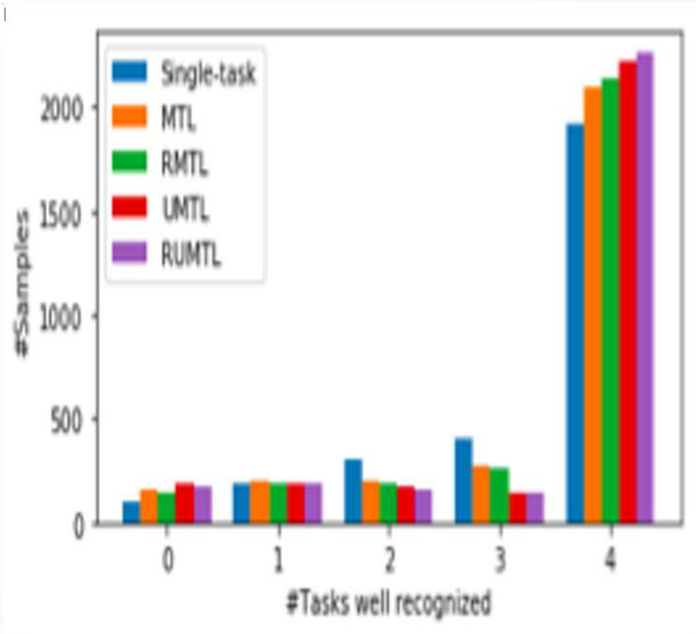


Food ingredients recognition

	Dish	Cuisine	Categories			Ingredients			
	Acc	*Acc*	F_1	*Pre*	*Rec*	F_1	*Pre*	*Rec*	*MTA*
Single-task	0.8334	0.8649	0.8709	0.8944	0.8485	**0.8992**	0.9143	0.8846	0.6713
MTL	0.8303	**0.8958**	0.8811	0.9042	0.8592	0.8780	0.8972	0.8596	0.6927
RMTL	0.8351	0.8917	0.8834	0.8789	0.8880	0.8809	0.8613	0.9014	0.7061
UMTL	0.8221	0.8944	0.8925	0.9067	0.8788	0.8943	0.9095	0.8795	0.7478
RUMTL	**0.8358**	0.8934	**0.8944**	0.9041	0.8848	0.8988	0.9084	0.8893	**0.7600**

Multi-task Accuracy: encourage errors to concentrate on the same data.



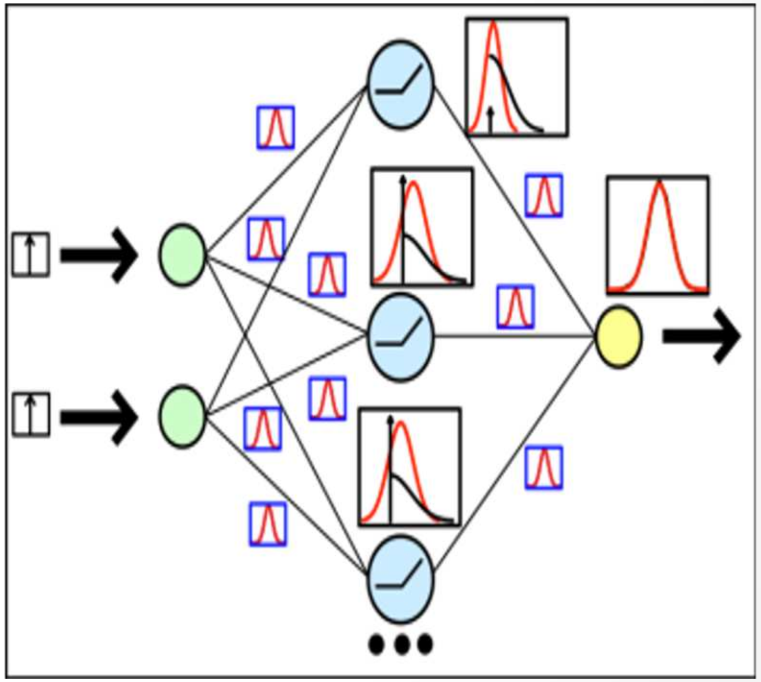


Contents

- The food image problem
- Multi-task food learning with aleatoric uncertainty
- Food recognition with epistemic uncertainty
- - GAN
- - Hierarchical classifier with epistemic ucnertainty
- Conclusions

Bayesian neural networks

Instead of learning the model's weights,
learn a distribution over the weights



- => estimate uncertainty over the weights.
- So how do we do that?

[Financial forecasting with probabilistic programming and Pyro]

Bayesian Neural Networks

At inference, instead of taking the single set of weights that maximized the posterior distribution, we consider all possible weights, weighted by their probability.

$$p(y|x, X, Y) = \int p(y|x, w)p(w|X, Y)dw$$

- p(y|x,w) is the likelihood,
- p(w|X,Y) is the posterior probability of the model's weights given the data.

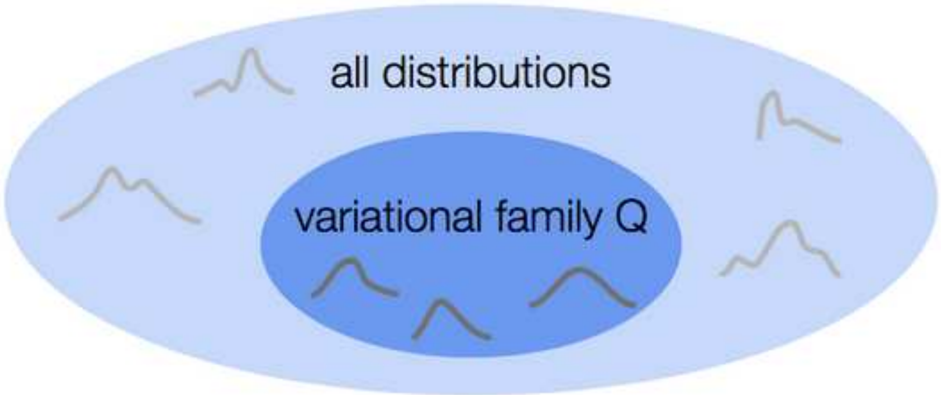
Bayesian Neural Networks

But, how to compute the posterior probability of the model's weights, p(w|X,Y)?

$$p(y|x, X, Y) = \int p(y|x, w)p(w|X, Y)dw$$

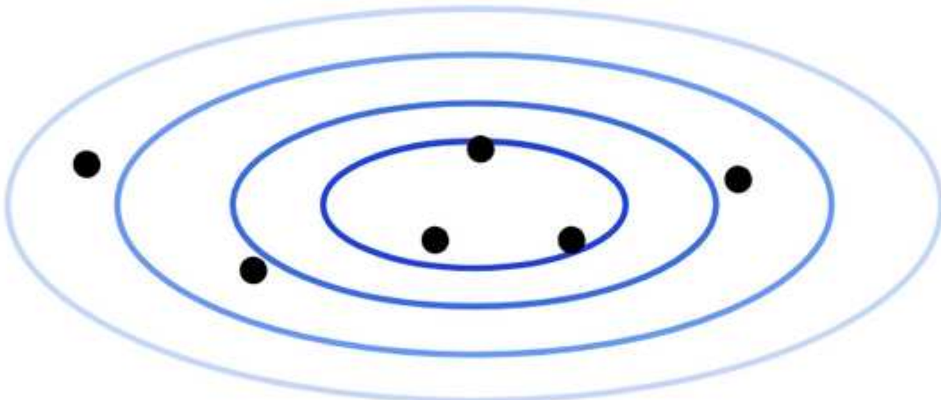
VARIATIONAL INFERENCE

HIGH BIAS - LOW VARIANCE



SAMPLING METHODS

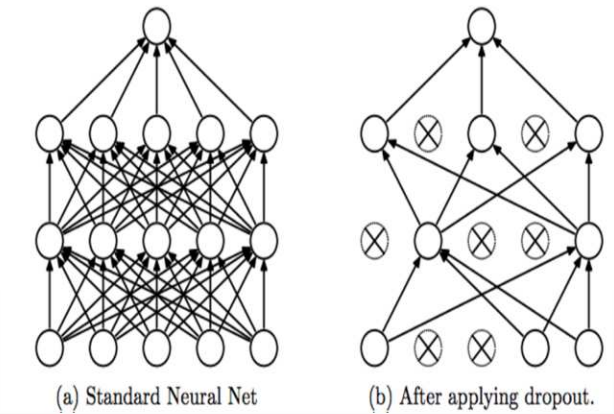
HIGH VARIANCE - LOW BIAS



How to estimate the Epistemic Uncertainty?

Gal and Ghahramani showed that dropout at inference time gives an uncertainty estimator:

1. Infer y|x multiple times, each time sample a different set of nodes to drop out.
2. Average the predictions to get the final prediction E(y|x).
3. Calculate the sample variance of the predictions.



How to estimate the Epistemic Uncertainty?

The Epistemic Uncertainty (EU) can be expressed as follows:

where

$$EU(x_t) = -\sum_{c=1}^{C} \overline{p(y_c = \hat{y}_c | x_t)} \ln(\overline{p(y_c = \hat{y}_c | x_t)}),$$

K Monte Carlo dropout simulations

$$\overline{p(y_c = \hat{y}_c | x)} = \frac{1}{K} \sum_{k=1}^{K} p(y_c^k = \hat{y_c^k} | x).$$

Class uncertainty

What to do with the difficult clases?

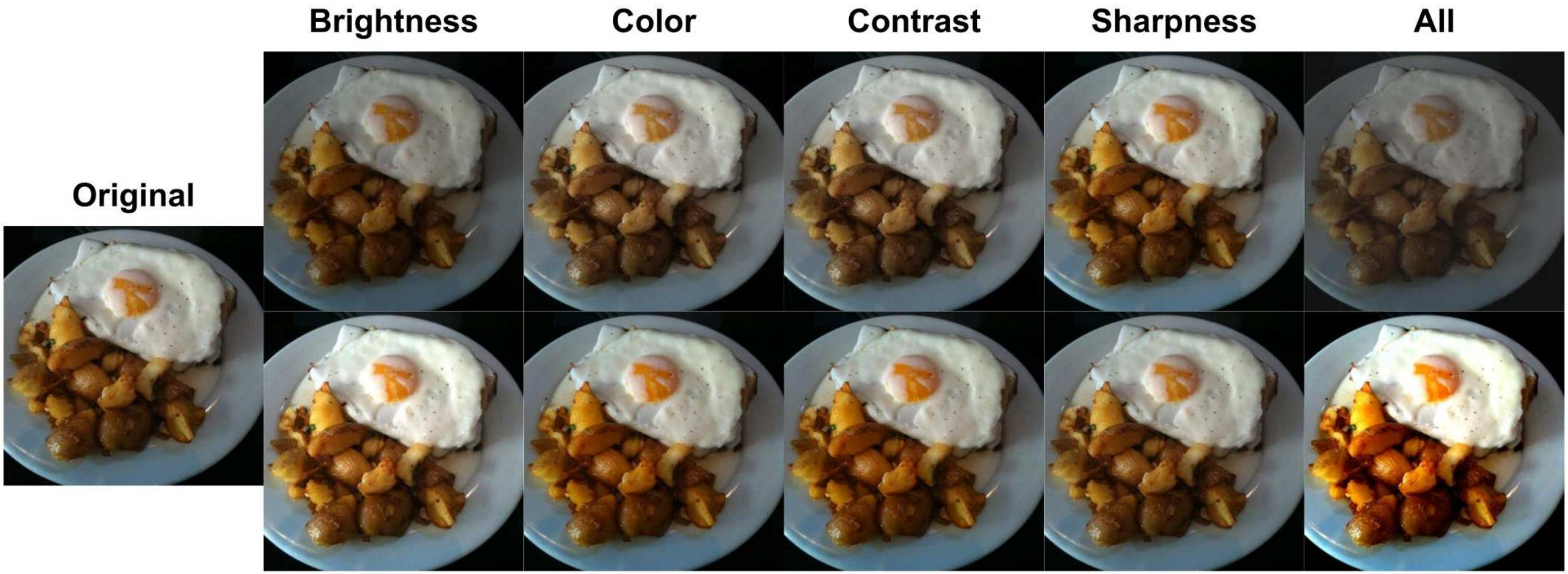
Are all clases well represented/easily discriminable?

Adapted from Gal (2016)

How to augment difficult clases?

- data augmentation

Use Uncertainty for Data Augmentation



Use Uncertainty for Data Augmentation

Sample of the synthetic images from the generator applied.

Class uncertainty

What to do with the difficult clases?

Are all clases well represented/easily discriminable?

Adapted from Gal (2016)

How to augment difficult clases?

- classic data augmentation, or
- creating synthetic images. How?

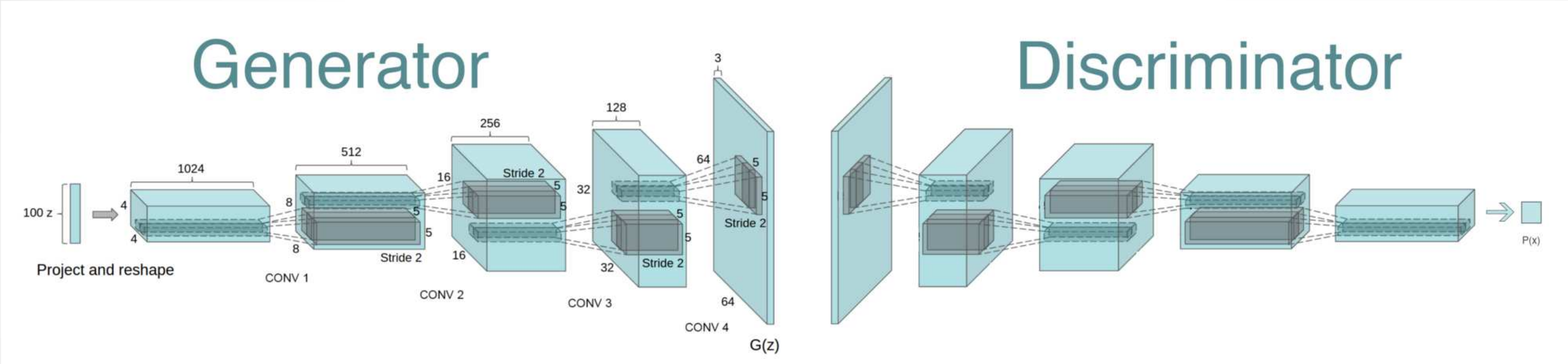
The Biggest Breakthrough In The History Of AI

- Celebrated computer scientist Yann Lecun observed:

“GANs and the variations that are now being proposed is the most interesting idea in the last 10 years in ML, in my opinion.”

VP and Chief AI Scientist, Facebook
Silver Professor of Computer Science, Data Science, Neural Science, and Electrical and Computer Engineering, New York University.
ACM Turing Award Laureate, (sounds like I'm bragging, but a condition of accepting the award is to write this next to you name)
Member, National Academy of Engineering

Generative Adversarial Network (GAN)



New components:
Transposed convolution,
Batch Normalization

Binary Classifier:
Conv, Leaky ReLU,
FC, Sigmoid

https://github.com/PramodShenoy/GANerations

GAN

Loss function for D

- If x is real, D(x) = 1; otherwise, D(x) = 0
- Minimize the error

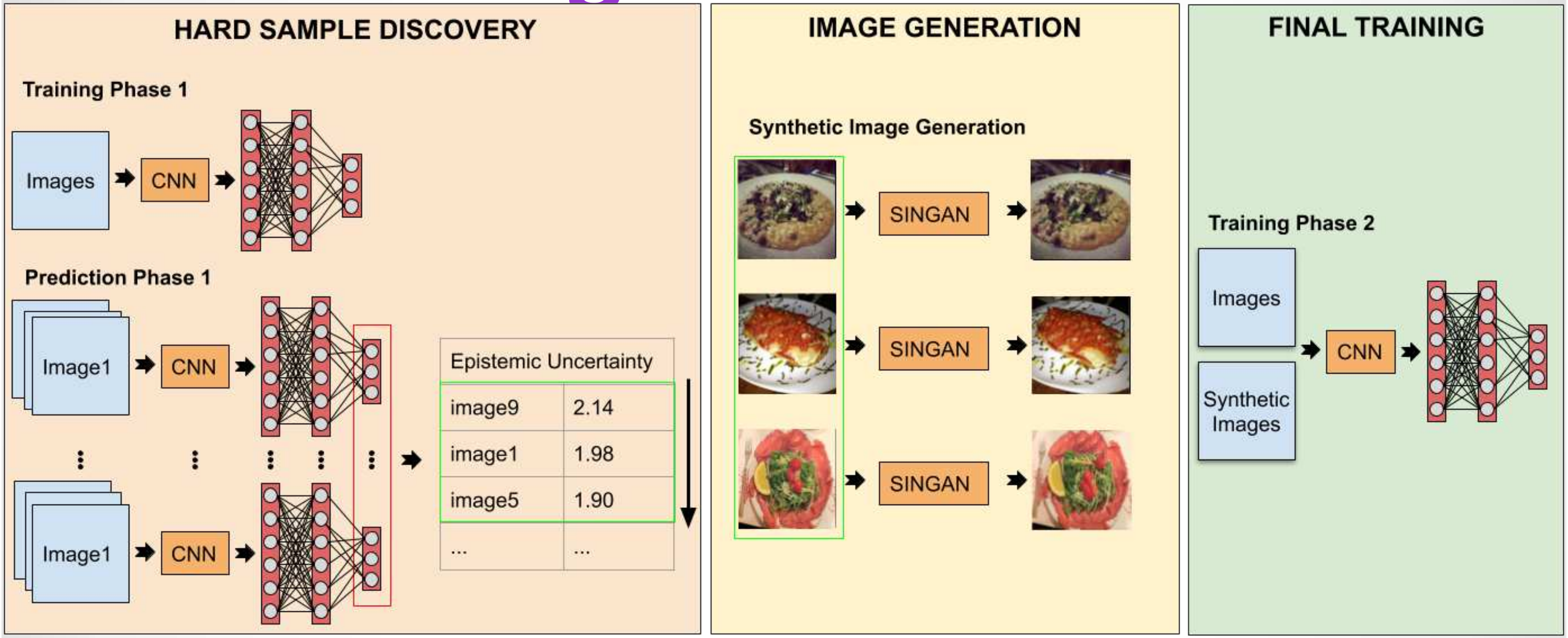
$$L_D = \underbrace{\mathbb{E}_x\, ln\, D(x)}_{\substack{\text{Real} \\ D(x) \to 1 \\ \Rightarrow ln\, D(x) \to 0}} + \underbrace{\mathbb{E}_z\, ln\, (1 - D(G(z)))}_{\substack{\text{Fake} \\ D(x) \to 0 \\ \Rightarrow ln\, (1 - D(x)) \to 0}}$$

- Loss function for G
  - Maximize the error of D

- Minimax procedure

$$\min_D \max_G \mathbb{E}_x\, ln\, D(x) + \mathbb{E}_z\, ln\, (1 - D(G(z)))$$

Use Uncertainty for Data Augmentation



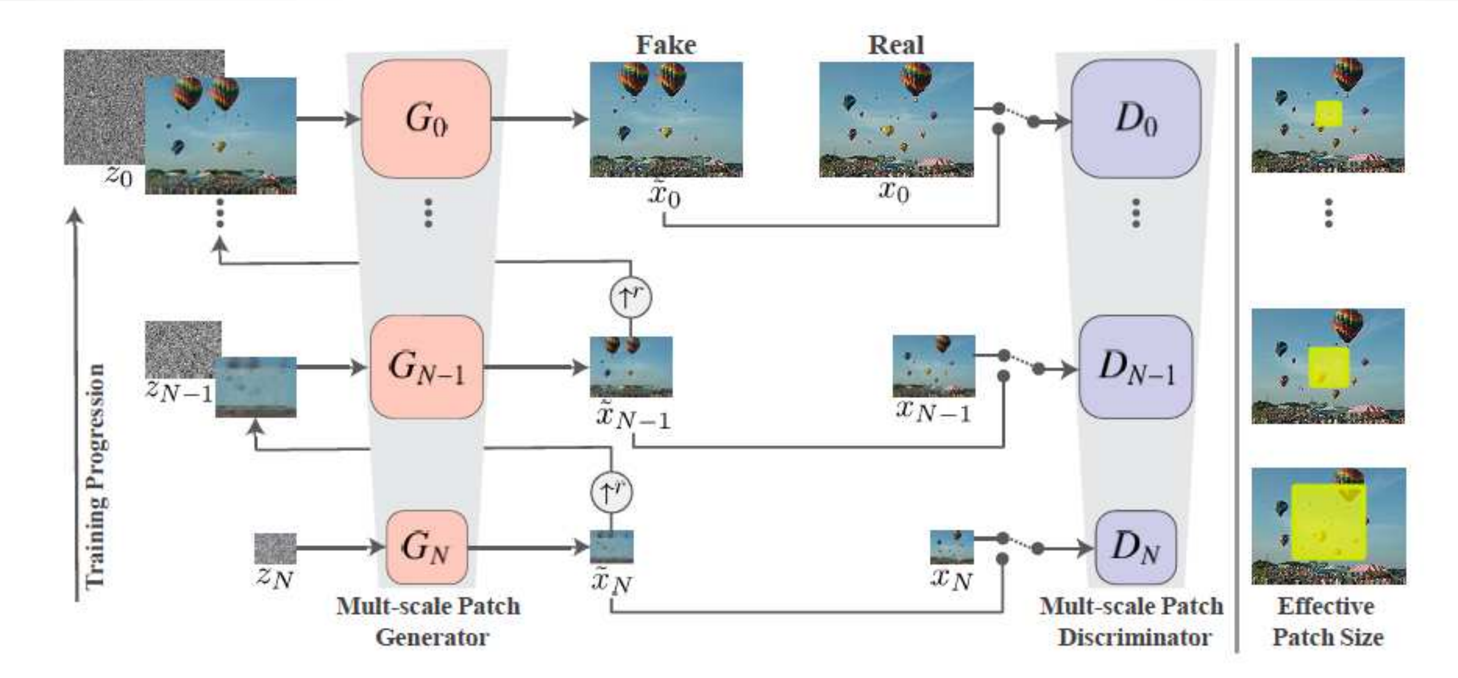
Use the data augmentation applied class-conditionally to improve the results in terms of accuracy and also to reduce the overall epistemic uncertainty.

During the prediction phase, the same image is fed to the CNN several times to calculate the epistemic uncertainty given by the model for that image

E. Aguilar, and P. Radeva. "Class-conditional Data Augmentation Applied to Image Classification." International Conference on Computer Analysis of Images and Patterns (CAIP),2019.

Validation

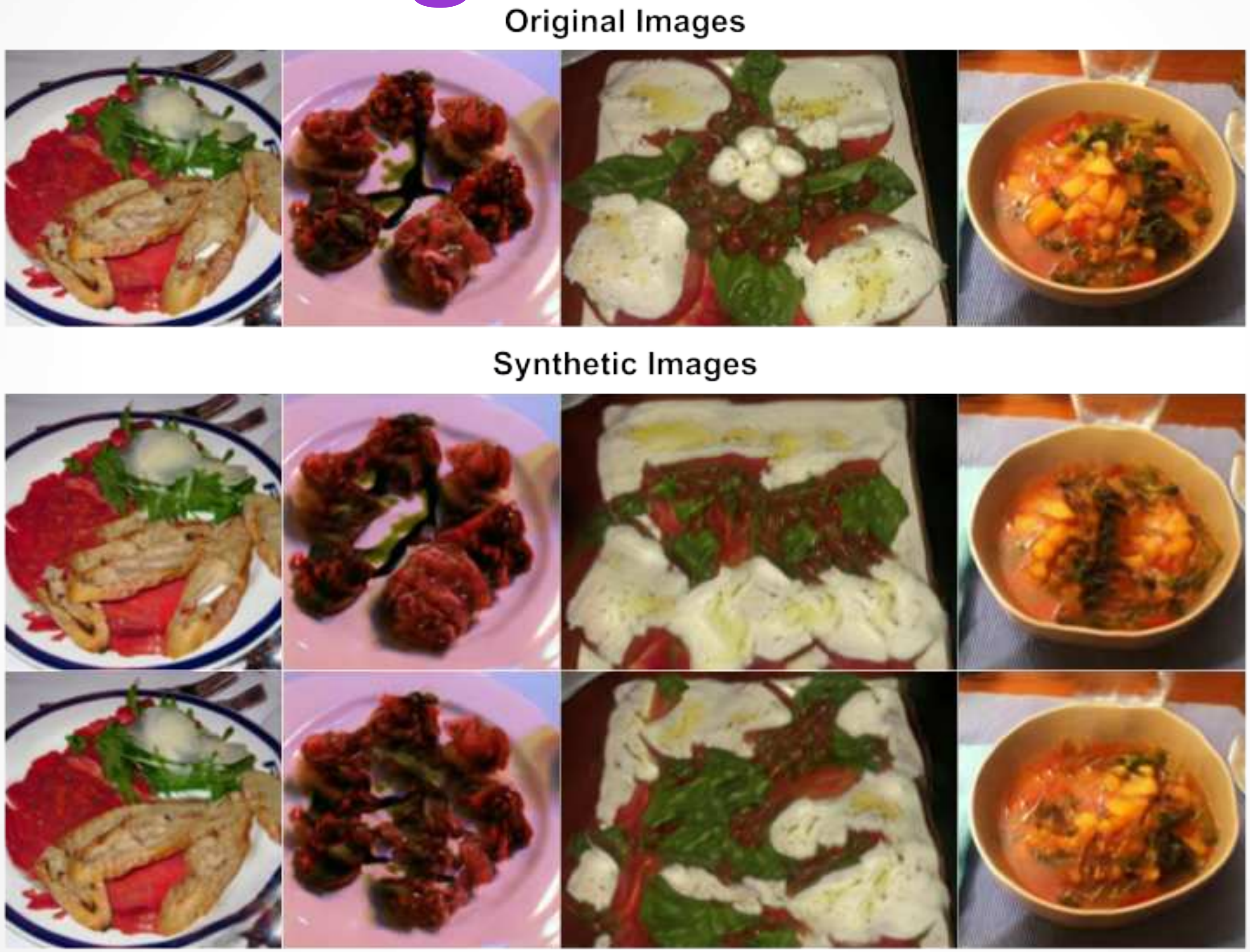
SINGAN



SinGAN's multi-scale pipeline: the model consists of a pyramid of GANs, where both training and inference are done in a coarse-to-fine fashion. At each scale, **G**n learns to generate image samples in which all the overlapping patches cannot be distinguished from the patches in the down-sampled training image, **x**n, by the discriminator **D**n; the effective patch size decreases as one goes up the pyramid (marked in yellow on the original image for illustration). The input to **G**n is a random noise image **z**n, and the generated image from the previous scale **~x**n, upsampled to the current resolution (except for the coarsest level which is purely generative).

Shaham, Tamar Rott, Tali Dekel, and Tomer Michaeli. "Singan: Learning a generative model from a single natural image." *Proceedings of the IEEE International Conference on Computer Vision*. 2019.

Use Uncertainty for Data Augmentation

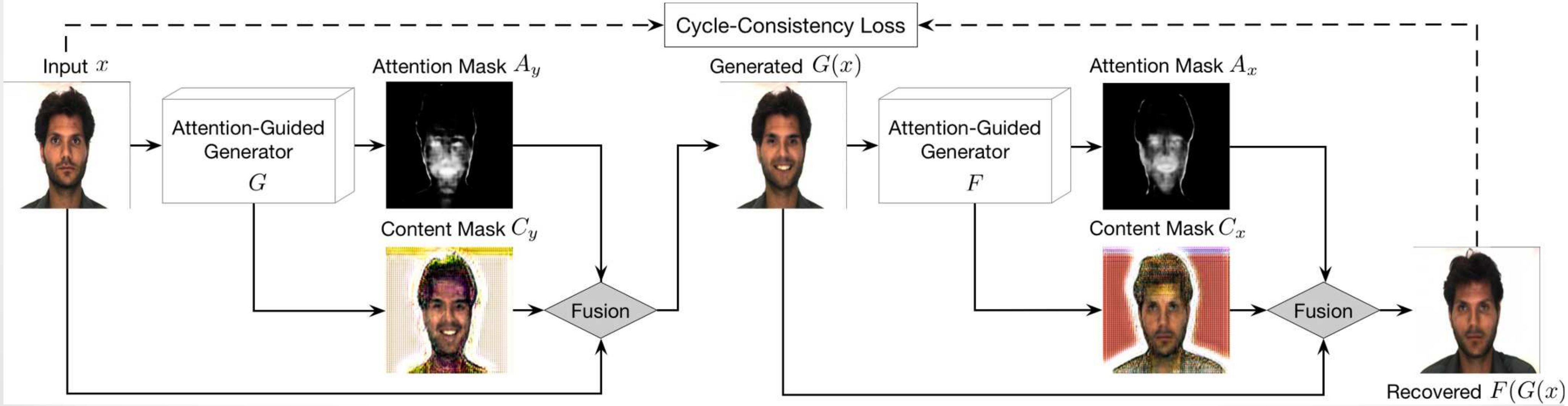


Synthetic image generated on the selected images from the training set

AttentionGAN

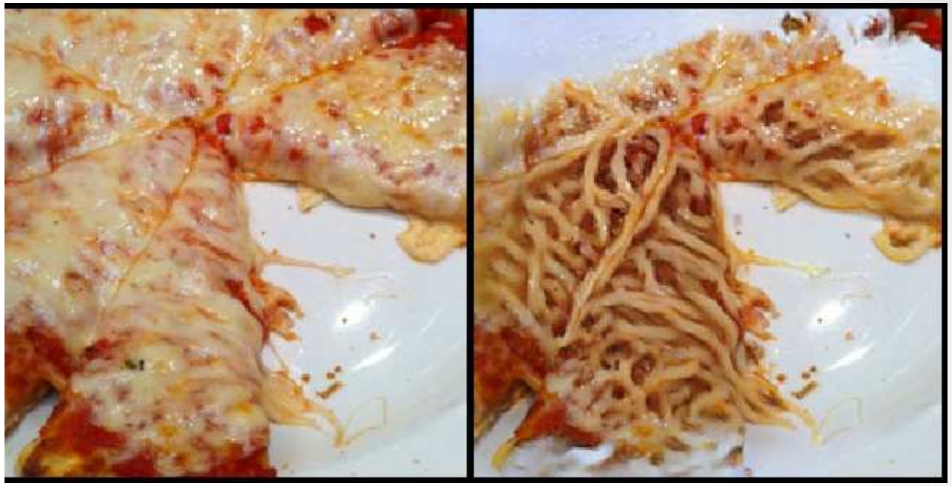
AttentionGAN: Unpaired Image-to-Image Translation using Attention-Guided Generative Adversarial Networks

Hao Tang, Hong Liu, Dan Xu, Philip H.S. Torr and Nicu Sebe



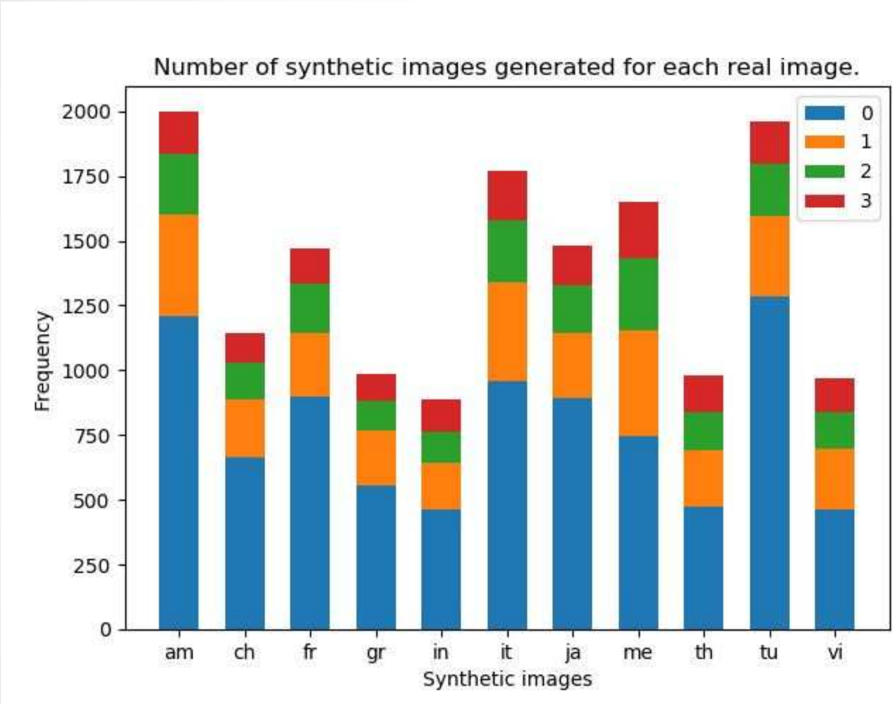
Framework of the proposed attention-guided generation scheme I, which contains two attention-guided generators G and F. One mapping is shown: x->G(x)->F(G(x))->x. The other mapping is: y->F(y)->G(F(y))->y. The attention-guided generators have a built-in attention module, which can perceive the most discriminative content between the source and target domains. The input image, the content mask and the attention mask are fused to synthesize the final result.

Using AttentionGan

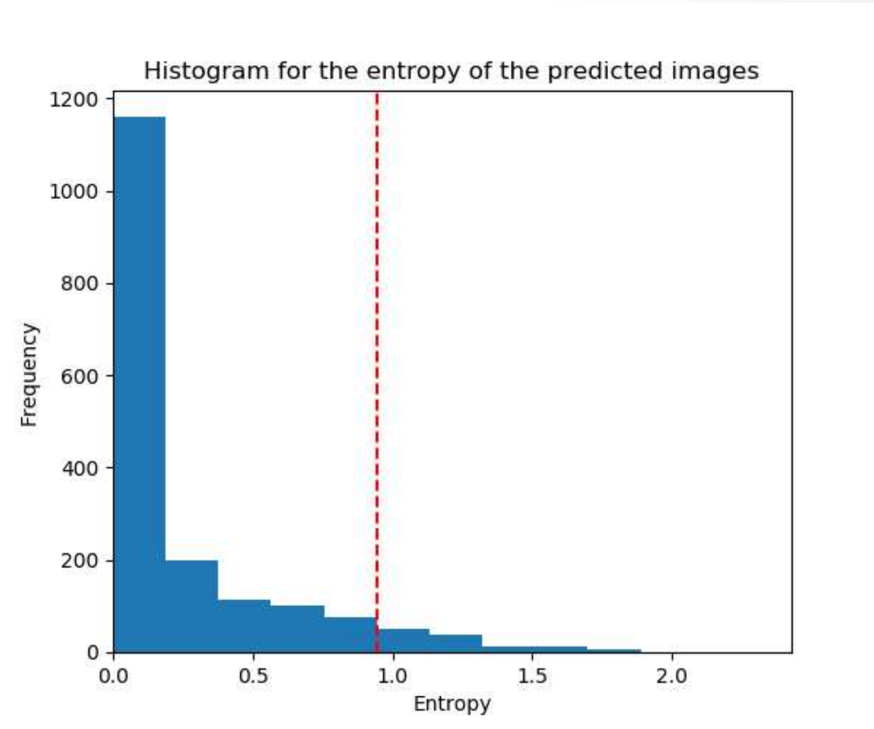


Chen, Xinyuan, et al. "Attention-GAN for object transfiguration in wild images." Proceedings of the European Conference on Computer Vision (ECCV). 2018.

Use Uncertainty for Data Augmentation

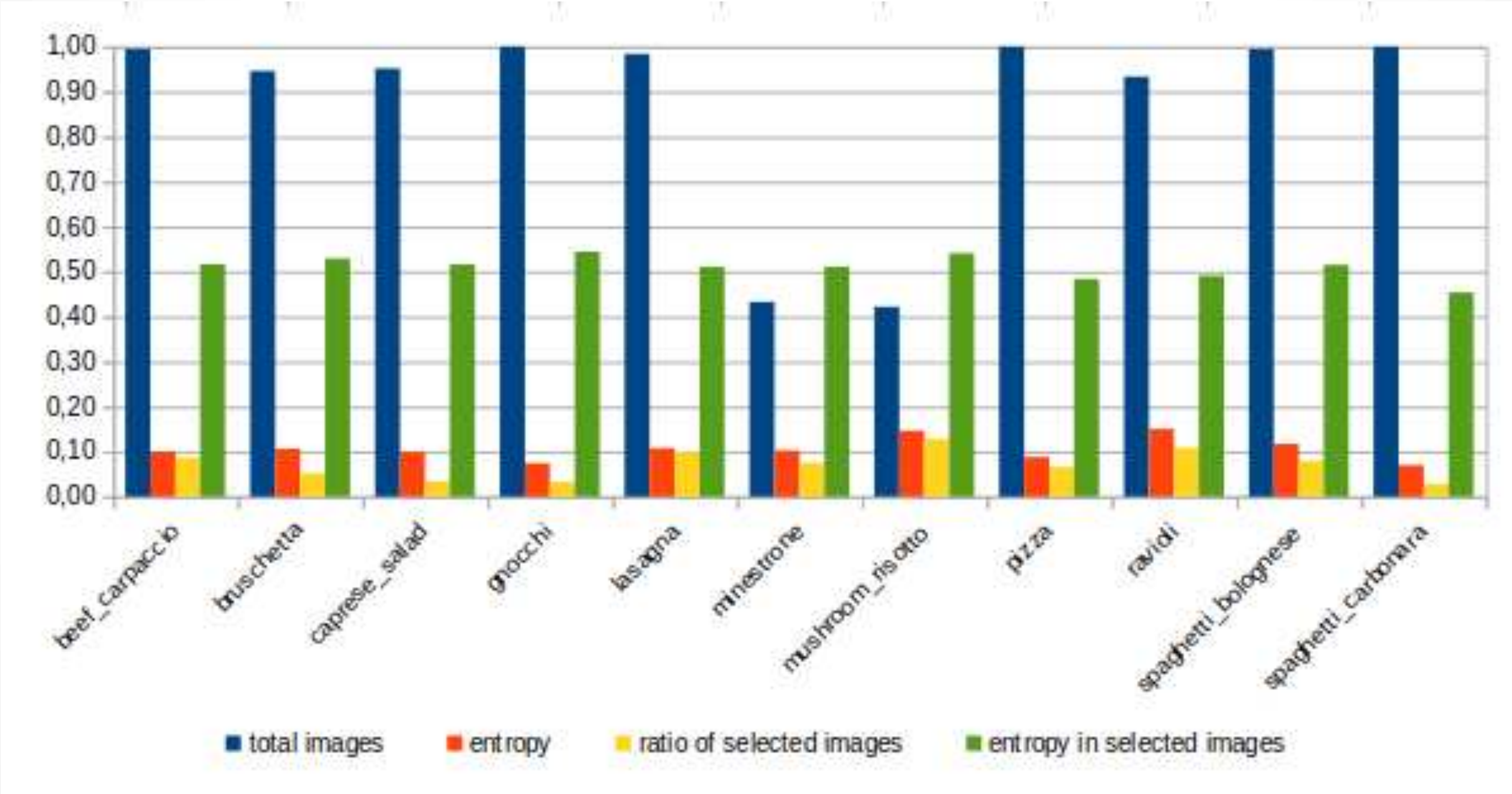


Number of synthetic images generated after the third training cycle.



Histogram for the entropy of the predicted images

Use Uncertainty for Data Augmentation



Training images vs epistemic uncertainty

E. Aguilar, and P. Radeva. "Uncertainty-aware Integration of Local and Flat Classifiers forFood Recognition." Pattern Recognition Letters, 2020.

Use Uncertainty for Data Augmentation

Model	*Acc*	*NEU*
ResNet50	61.00%	30.22%
ResNet50+DA	65.02%	33.55%
ResNet50+DA+A	64.65%	36.53%
Proposed method	65.54%	33,51%

Results on UECFOOD-256 in terms of Acc and NEU for the models trained with different data augmentation techniques.

Model	*Acc*	*NEU*
ResNet50	77.66%	19.85%
ResNet50+DA	82.65%	27.35%
ResNet50+DA+A	82.54%	29.45%
Proposed method	82.82%	26.25%

Results on Food-101 in terms of Acc and NEU for the models trained with different data augmentation techniques.

Use Uncertainty for Data Augmentation

Dataset	ResNet50 (S_1)	ResNet50 (S_2)	ResNet50 (S_3)	ResNet50 (S_4)
American	81,99%	83,69%	84,10%	**84,26%**
Chinese	87,93%	90,05%	90,60%	**91,17%**
French	89,01%	90,33%	**94,12%**	92,54%
Greek	89,12%	89,34%	89,90%	**92,11%**
Indian	87,67%	92,96%	**93,29%**	92,41%
Italian	80,72%	82,44%	**84,31%**	84,07%
Japanese	88,08%	90,85%	**91,20%**	90,93%
Mexican	79,12%	80,37%	81,64%	**81,96%**
Thai	70,98%	**79,91%**	79,85%	79,22%
Turkish	91,44%	91,65%	91,92%	**92,15%**
Vietnamese	84,67%	86,99%	88,14%	**89,85%**

Results obtained on the test sets in terms of Rmacro

$$R_{macro}(y, \hat{y}) = \frac{1}{|C|} \sum_{c \in C} R_{micro}(y_c, \hat{y}_c)$$

•

How many dishes there are all over the world?

WIKIPEDIA
The Free Encyclopedia

More than
100.000 basic
foods

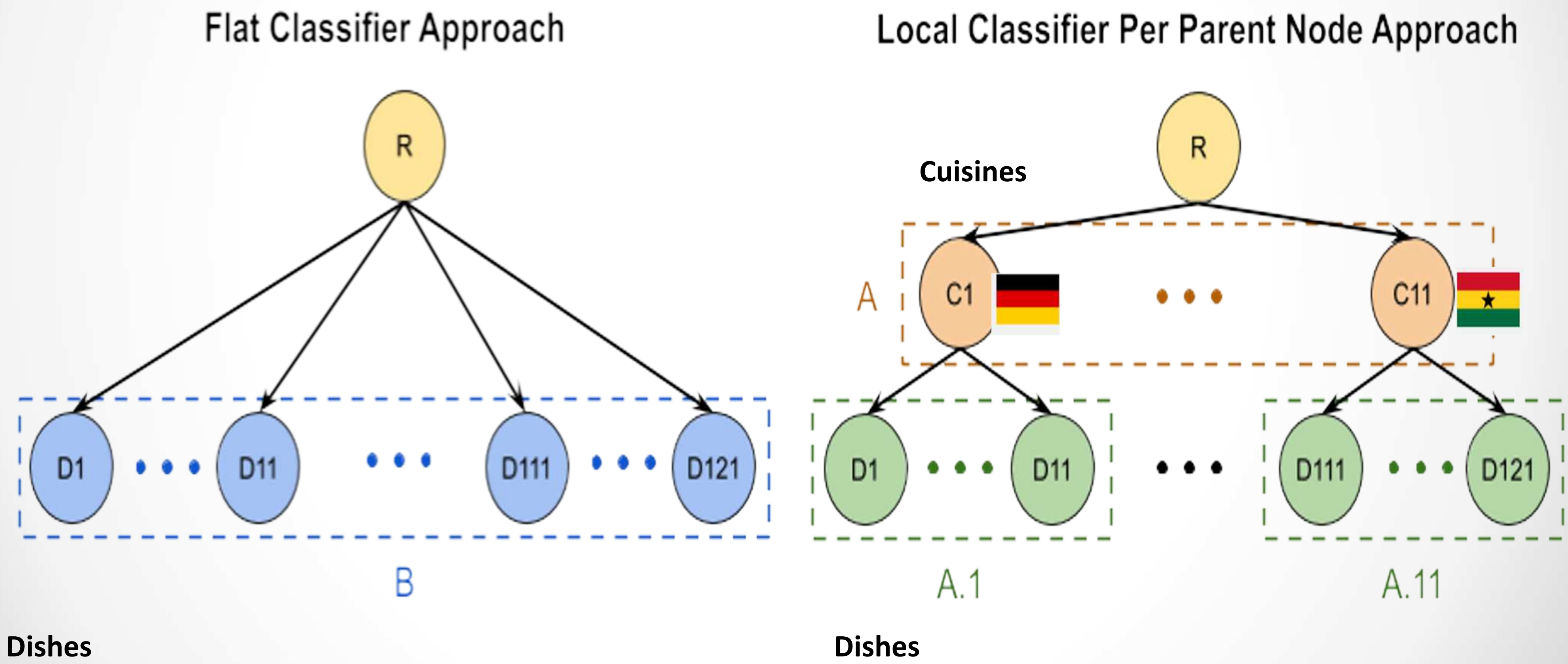
Imagine

- When you visit Mexico,

what is the probability to eat a food from Norway?

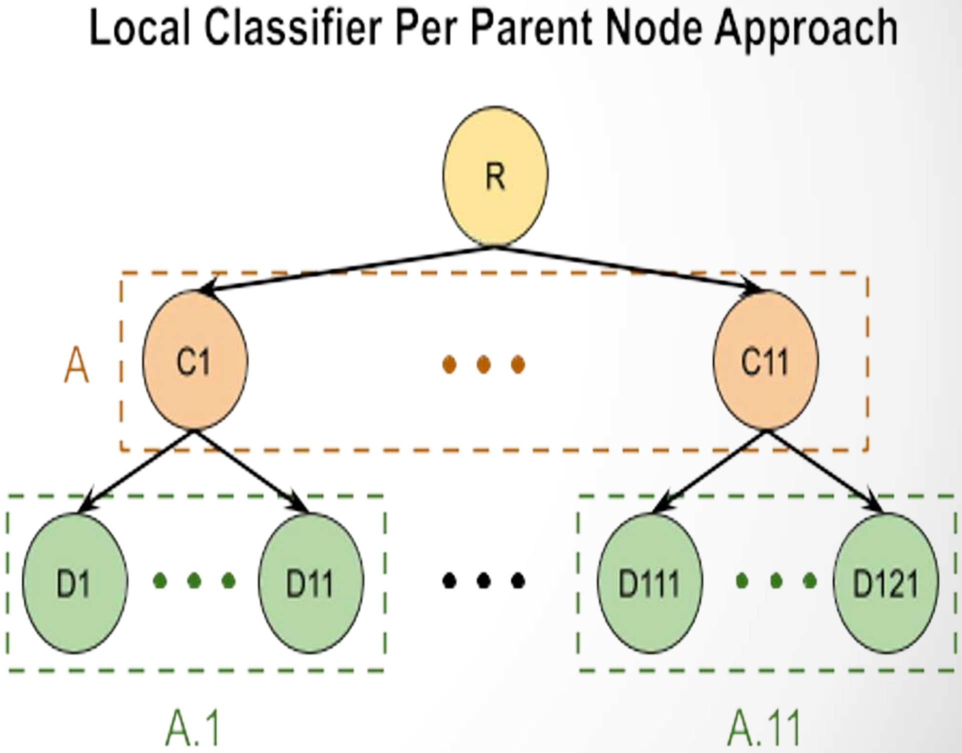
Let's organize classes in meta-classes

Let’s organize classes in meta-classes



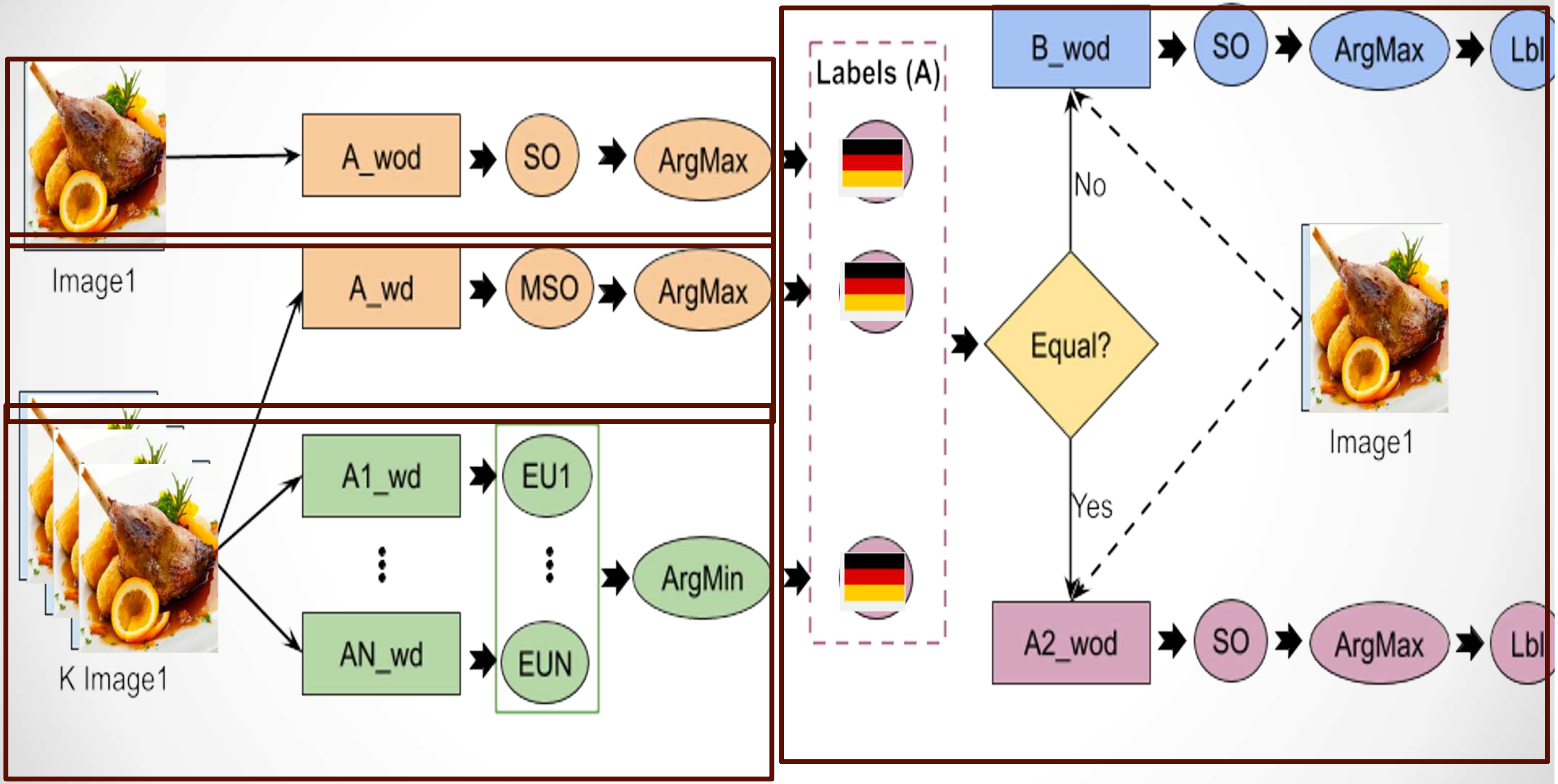
But …. Hierarchical classifiers have a big problem

Error propagation



Hypothesis: use uncertainty to decide if a LPN should be used

Proposed Method



Aguilar, Eduardo, and Petia Radeva. "Food Recognition by Integrating Local and Flat Classifiers." *Pattern Recognition Letters*, 2020 (in press).

Validation

MAFood Data - Ingredients101

Carrot Cake

Baby Back Ribs

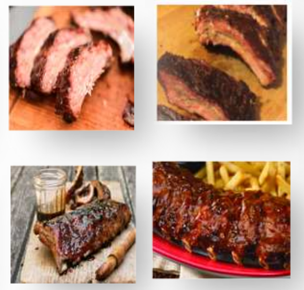
Dataset complementary to Food101:

- 101 classes / dishes
- 1000 images per class

A recipe for each downloaded resulting in a list of ingredients per class and a total of 446 unique ingredients.

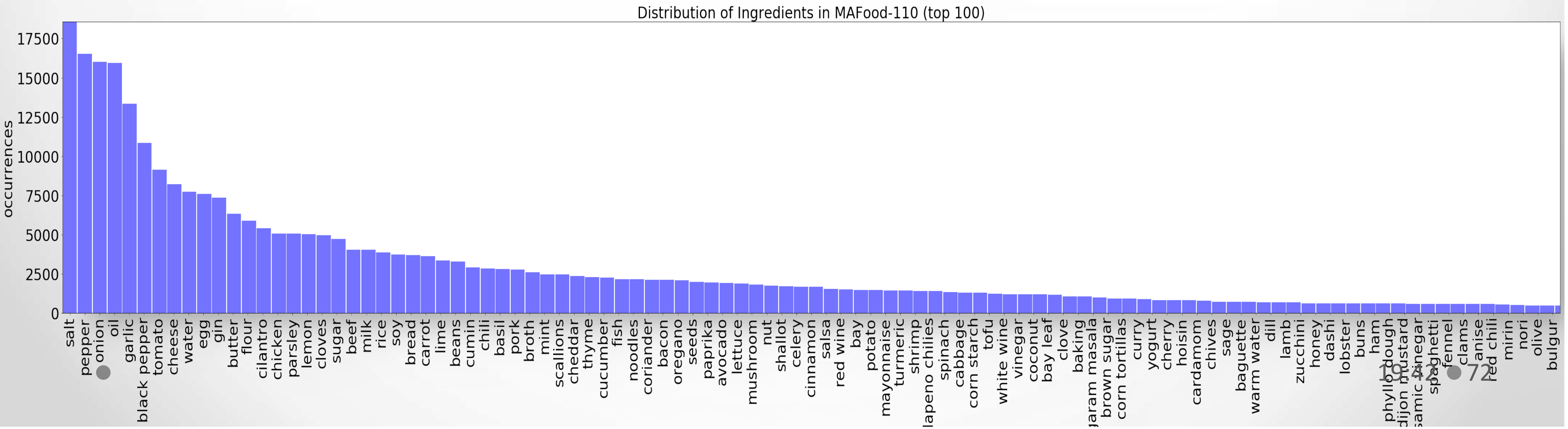
Yummly

Ingredients: 'salt', 'butter', 'all-purpose flour', 'large eggs', 'vanilla extract', 'baking powder', 'carrots', 'granulated sugar', 'powdered sugar', 'baking soda', 'brown sugar', 'ground cinnamon', 'canola oil', 'cream cheese', 'sour cream', 'ground nutmeg', 'chopped pecans', 'unsweetened applesauce',

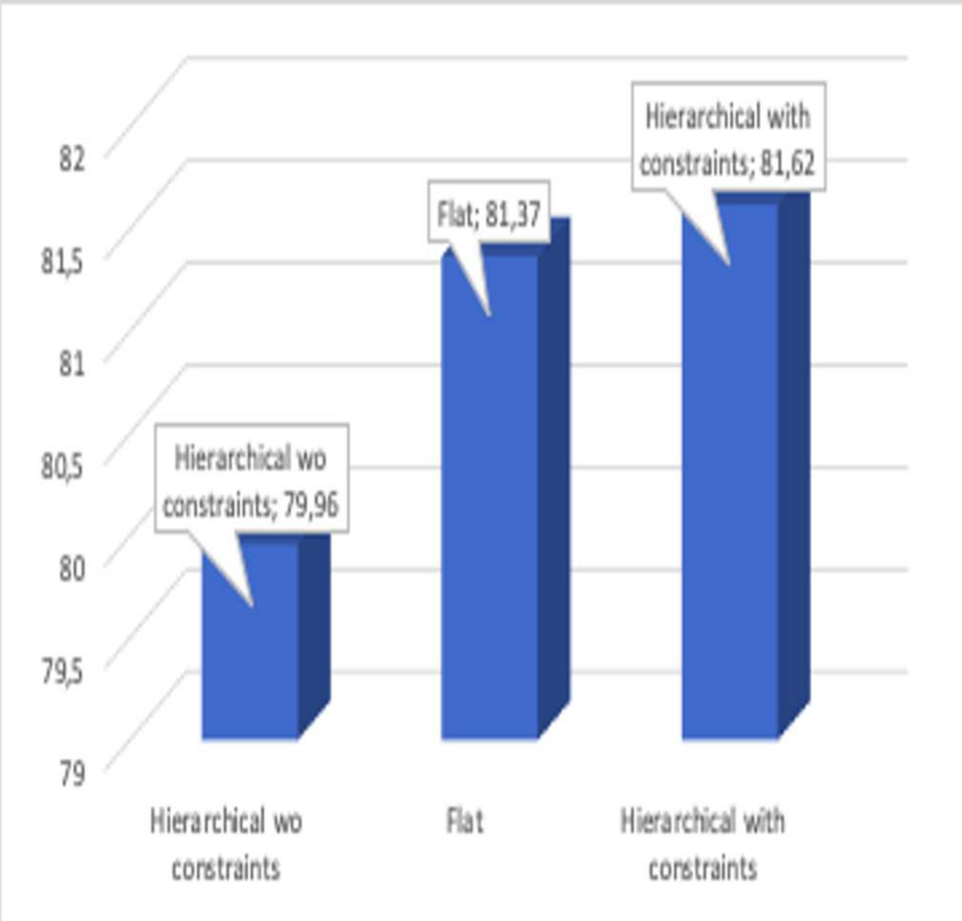


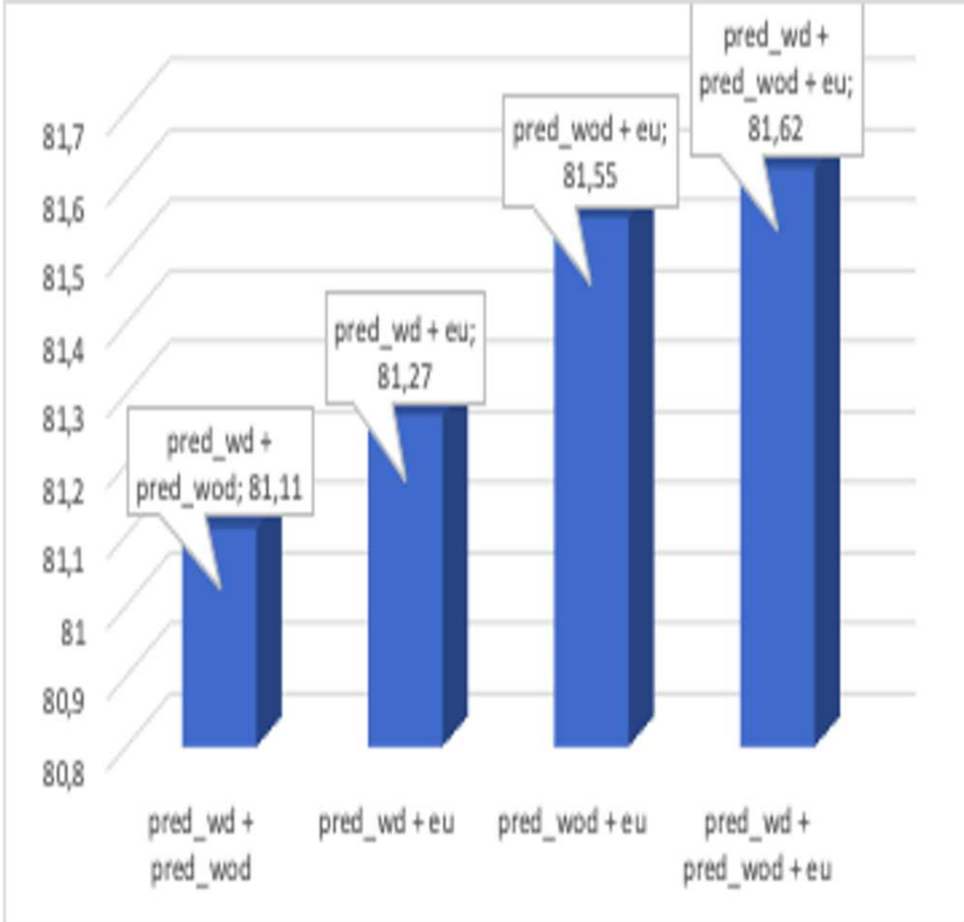
Ingredients: 'barbecue sauce', 'baby back ribs', 'chips', 'barbecue rub',

A total of 279 different ingredients were considered, visible or not, with an average of 19 per dish.



Ablation study





Results - Samples of the Smallest and Largest EU within the same class of Dish

Caesar Salad

Ravioli

Steak

Tacos

Conclusions

- Food image world brings us huge amount of data and Computer Vision questions

- Transfer learning and its subproblems (multi-task learning) open new opportunities

- Uncertainty modeling is a hot topic with many open questions and challenges!
  - Exclusivity relation between elements helps to the classification

  - Epistemic uncertainty

    New method for robust hierarchical classifiers..
    A good cue to improve recognition scalability.
    Epistemic uncertainty useful beyond the confidence of the model.
  - Aleatoric uncertainty

    Allows to weight different tasks according to uncertainty

  - For first time a food ontology is integrated into an end-to-end model

A huge impact of food analysis is expected from point of view of:

- Science, but also
- Real world applications, specially important for the society.


Thank you!