The 12th KIAS Workshop on Particle Physics and Cosmology and
2024 Korea-France STAR Workshop @ KIAS
November 22, 2024

Weak Supervision in New Physics Searches

Cheng-Wei Chiang
National Taiwan University
National Center for Theoretical Sciences

Ref: Hugues Beauchesne, Zong-En Chen, and CWC, JHEP 02 (2024) 138

Outline

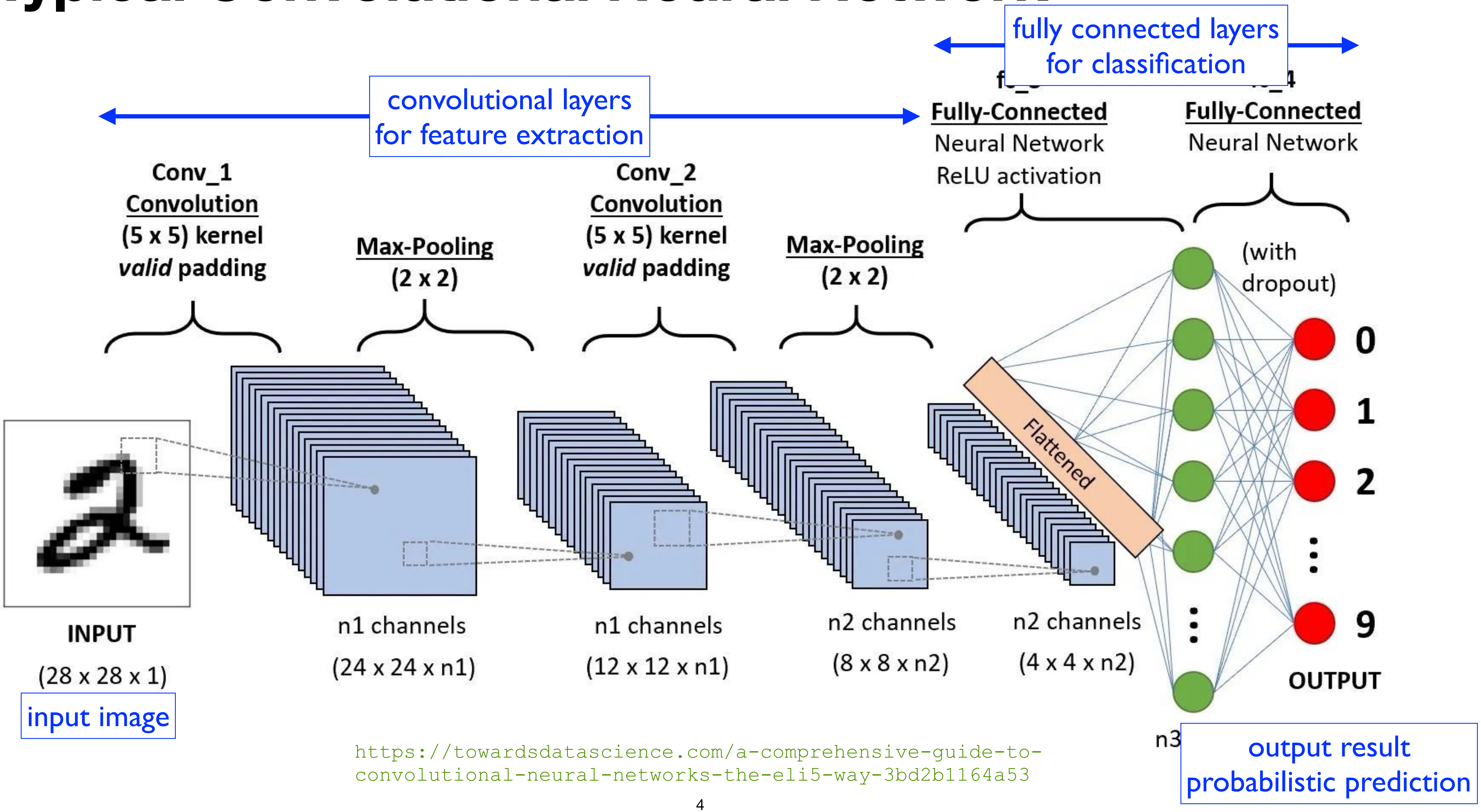
- Introduction
- CNN with full supervision
- CNN with weak supervision — CWoLa
- Dark valley model as our protagonist
- Transfer learning
- Summary

Types of Machine Learning

- **Supervised learning** (full and weak supervision)
  - Training data with labels (e.g., recognizing photos of cats and dogs)
- **Unsupervised learning**
  - Training data without labels (e.g., analyze and cluster unlabeled datasets)
- **Reinforced learning**
  - Data from interactions with the environment (e.g., chess and Go games)

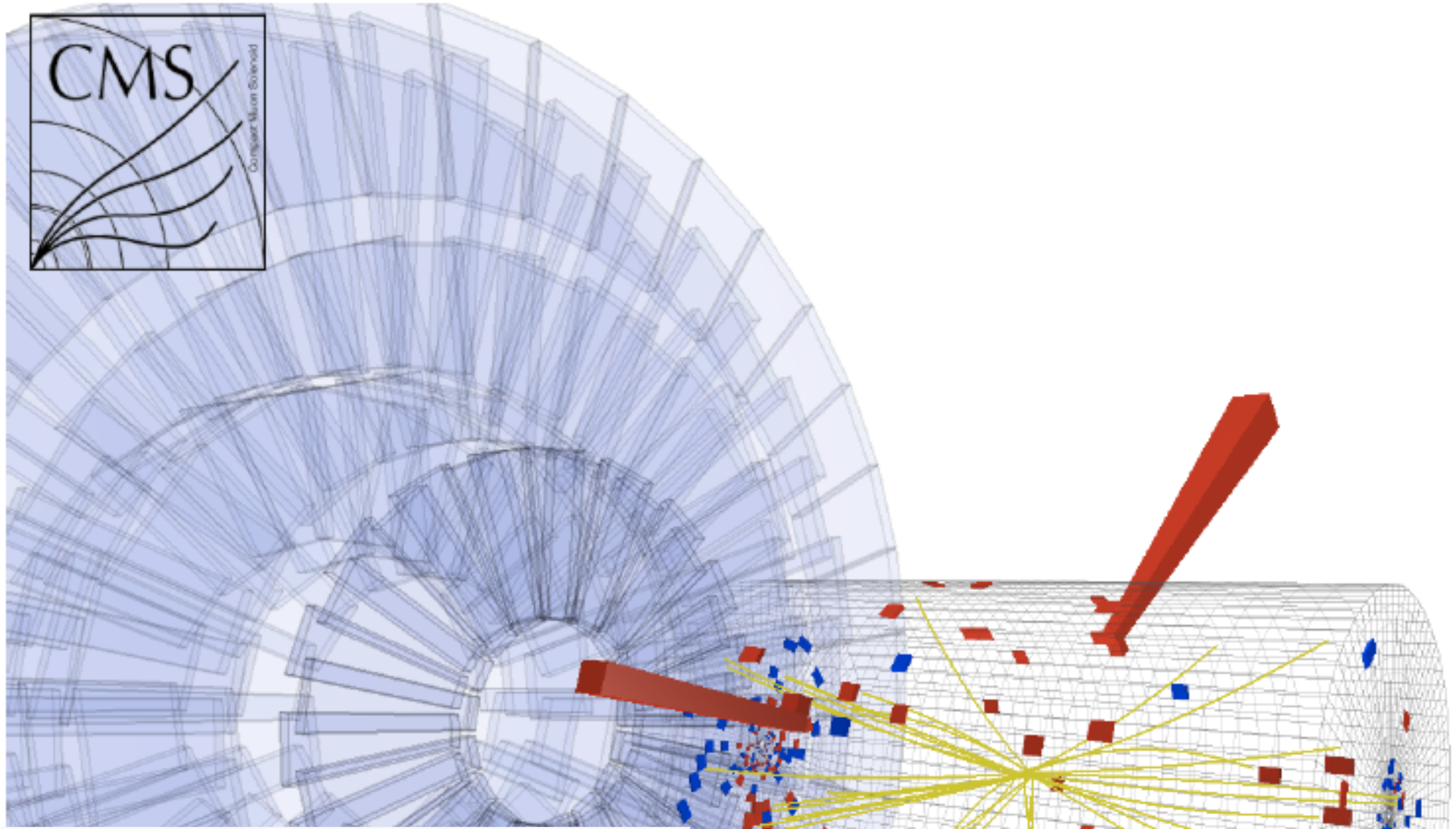
https://www.youtube.com/watch?v=Atg-S132vOo

A Typical Convolutional Neural Network



https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53

A Higgs to Diphoton Event



Event parameters:

$M_{\gamma\gamma}$ = 125.9 GeV

$p_T^{\gamma 1}$ = 89.8 GeV

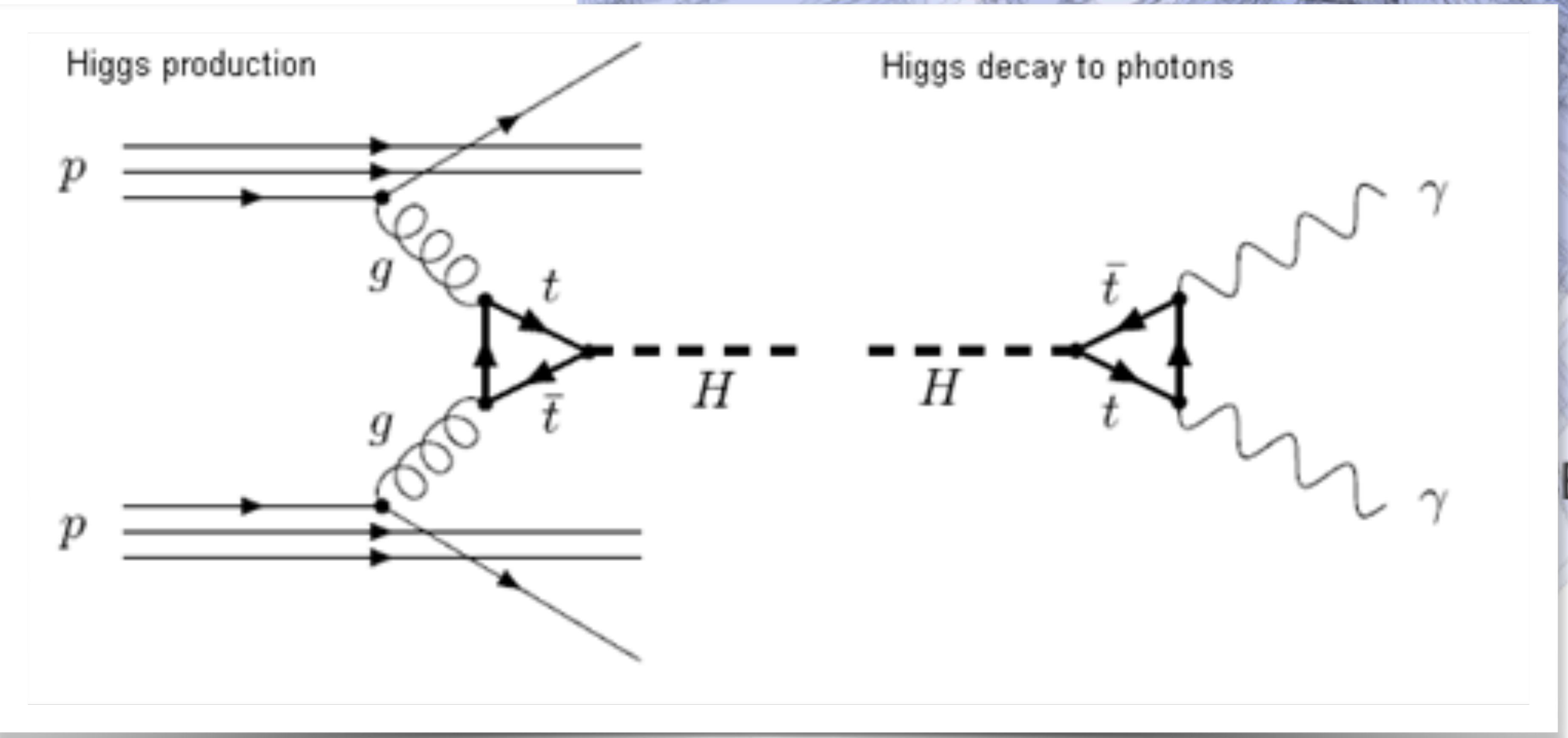
$p_T^{\gamma 2}$ = 46.5 GeV

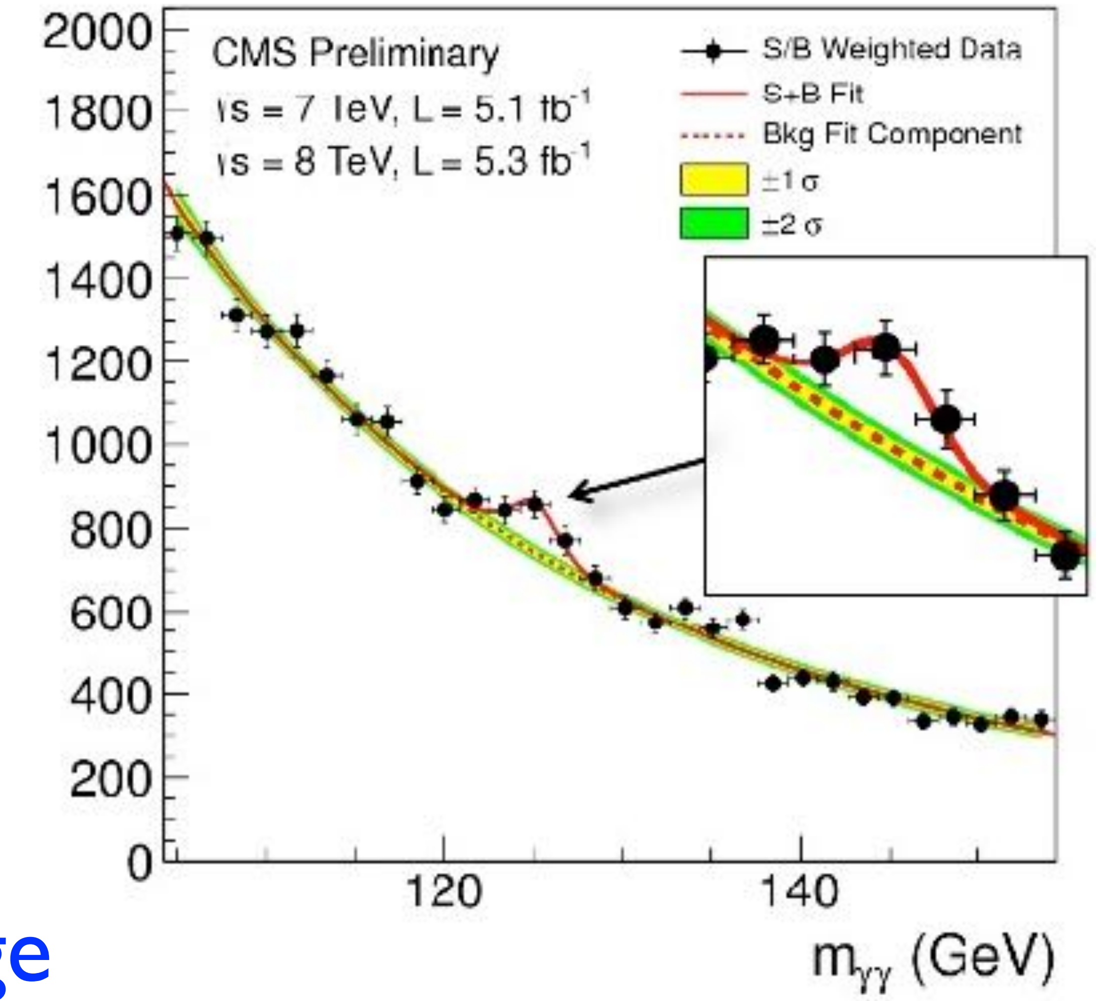
$\eta_{\gamma 1}$ = 0.06

$\eta_{\gamma 2}$ = -0.81

σ_M/M =0.89%

$p_T^{\gamma\gamma}$ =78.4 GeV





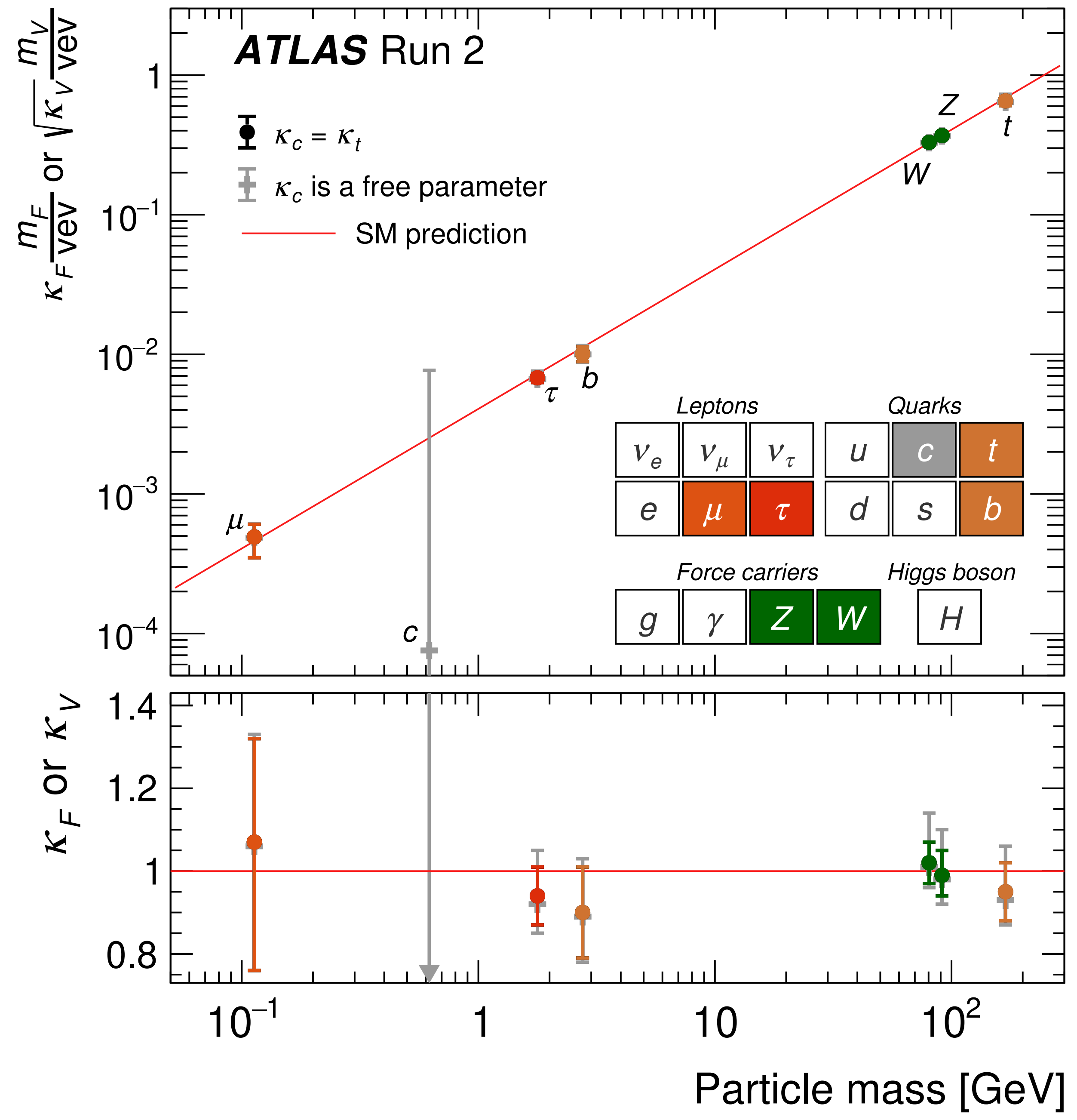
open up to render a 2D image

Full Supervision — One Application of CNN to Collider Physics

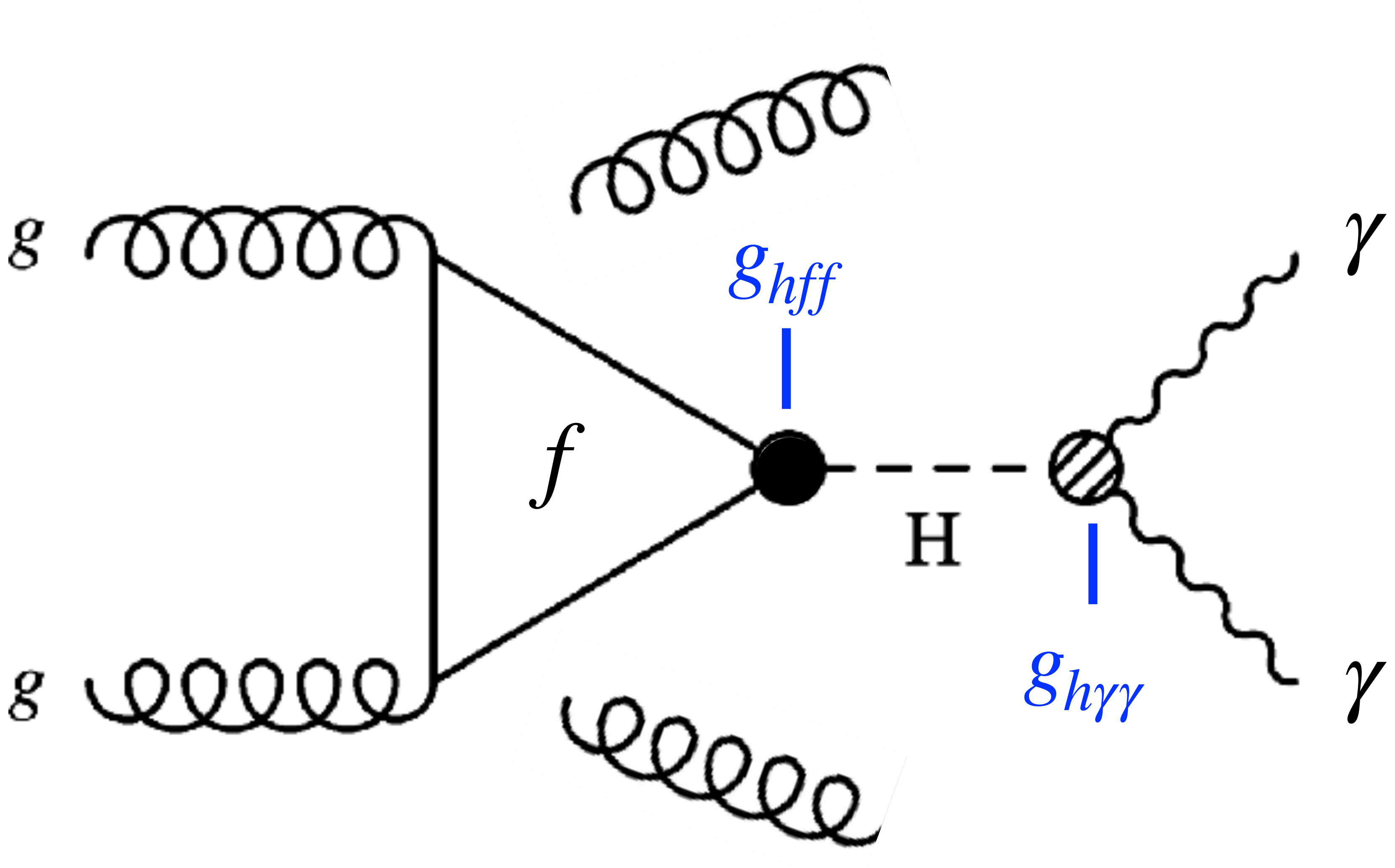
CWC, David Shih and Shang-Fu Wei, PRD **107**, 016014 (2023)
[talked at High1 2023]

VBF vs GGF

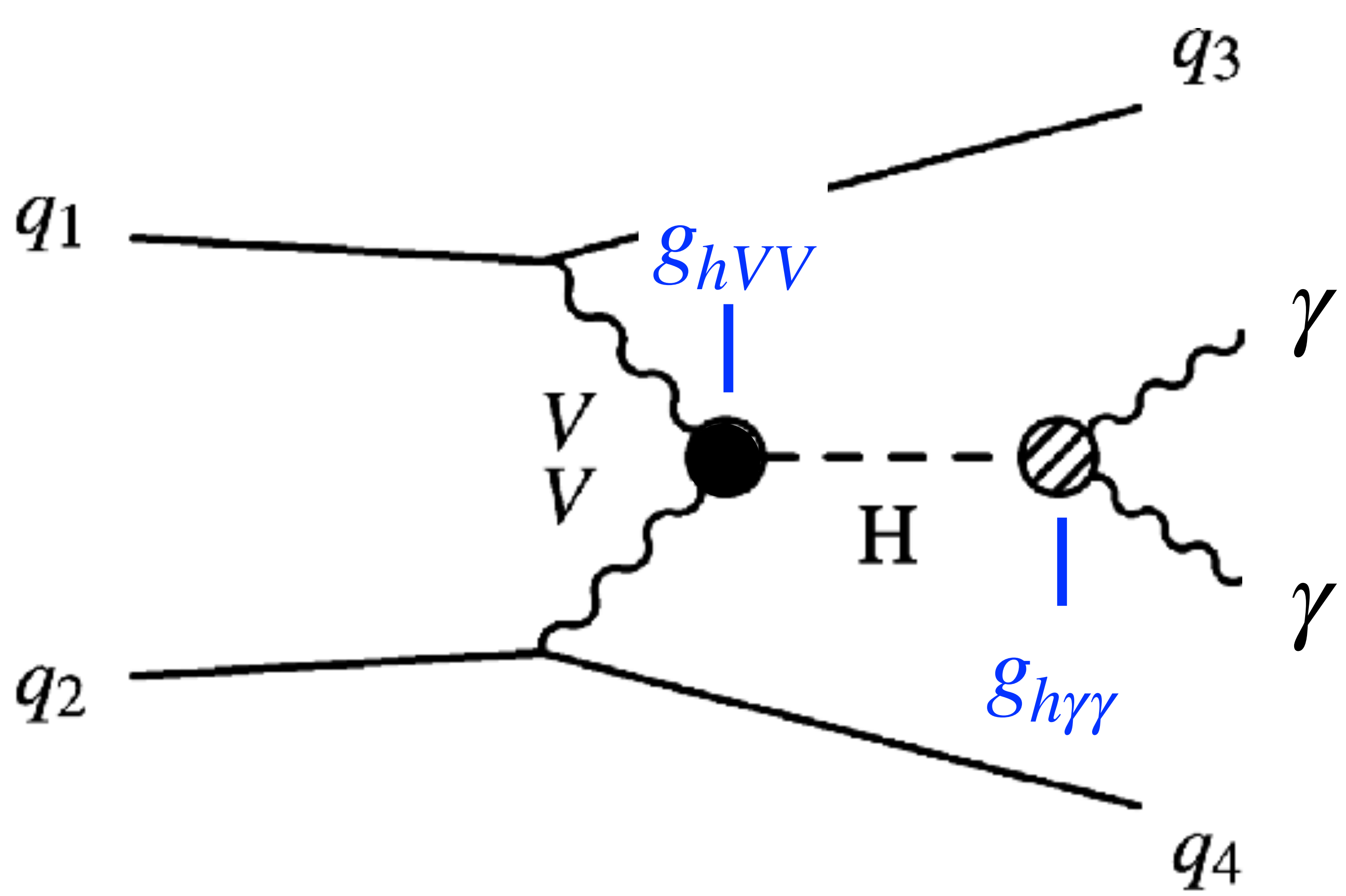
- **VBF** processes or the g_{hVV} coupling is essential for studying the role of the Higgs boson in the EWSB.
- Questions:
  - For any *detected* Higgs event, how can we *efficiently* and *correctly* determine/label its production mechanism?
  - Can it be *independent* of how the Higgs boson decays?



ATLAS 2019



(a) ggF production



(b) VBF production

BDT Input Features

- **Human-engineered high-level features** (kinematic and jet shape variables) used in BDTs:

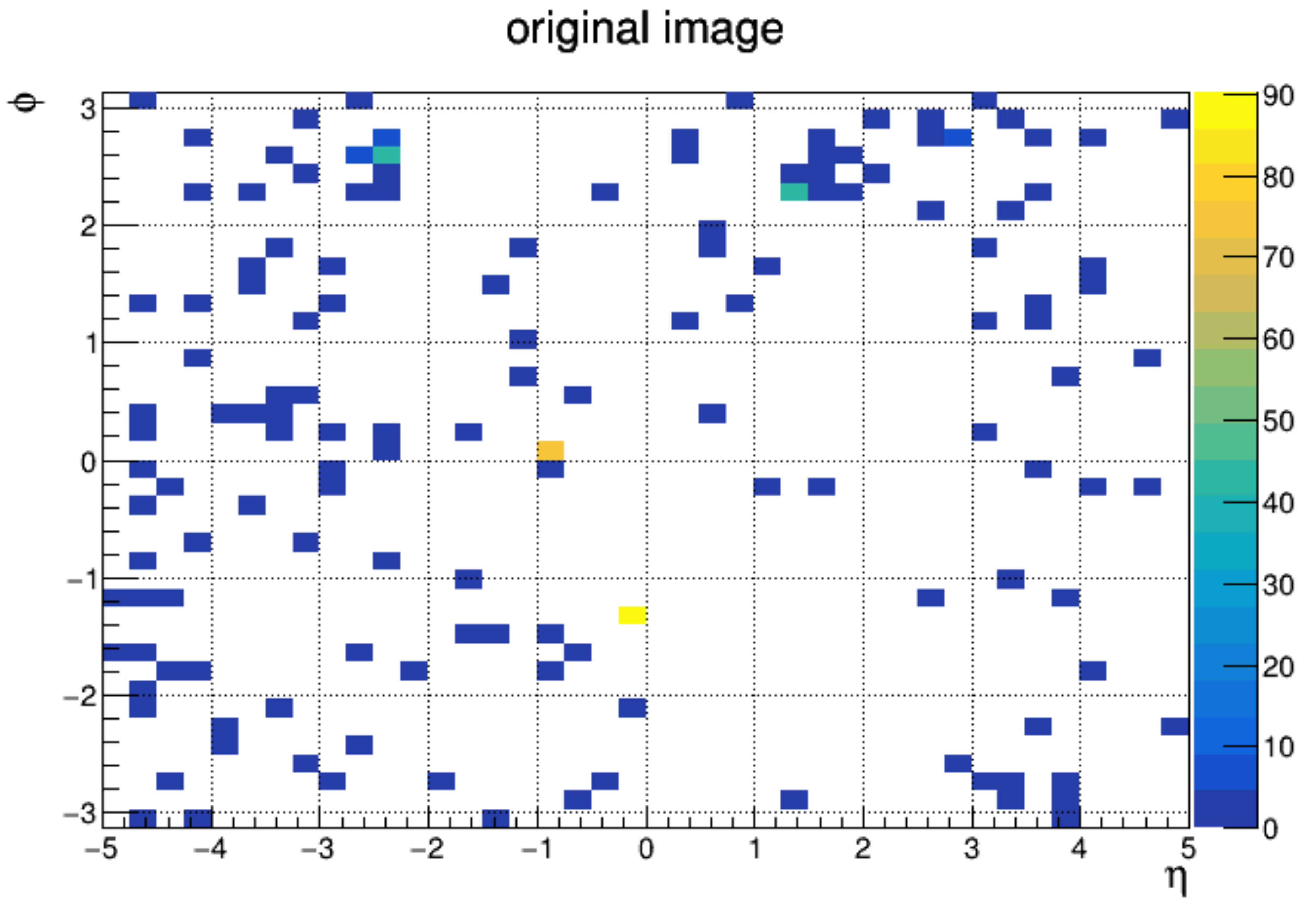
Higgs decay product-related	
baseline ATLAS 2018	1. m_{jj}, the invariant mass of j_1 and j_2 2. $\Delta\eta_{jj}$, the absolute difference of the pseudo-rapidities of j_1 and j_2 3. ϕ^*, defined by the ϕ-difference between the leading di-photon and di-jet 4. $p_{Tt}^{\gamma\gamma}$, defined by $\left\|(\mathbf{p}_T^{\gamma_1} + \mathbf{p}_T^{\gamma_2}) \times \hat{t}\right\|$, where $\hat{t} = (\mathbf{p}_T^{\gamma_1} - \mathbf{p}_T^{\gamma_2}) / \|\mathbf{p}_T^{\gamma_1} - \mathbf{p}_T^{\gamma_2}\|$ 5. $\Delta R_{\gamma j}^{\min}$, defined by the minimum η-ϕ separation between γ_1/γ_2 and j_1/j_2 6. η^*, defined by $\|\eta_{\gamma_1\gamma_2} - (\eta_{j_1} + \eta_{j_2})/2\|$, where $\eta_{\gamma_1\gamma_2}$ is the pseudo-rapidity of the leading di-photon
shape Shelton 2013	7. the girth summed over the two leading jets $\sum_{j=1}^{2} g_j = \sum_{j=1}^{2} \sum_{i\in J^j}^{N} p_{T,i}^j r_i^j / p_T^j$ 8. the central integrated jet shape $\Psi_c = \sum_{j=1}^{2} \sum_{i\in J^j}^{N} p_{T,i}^j (0 < r_i^j < 0.1)/(2p_T^j)$ 9. the sided integrated jet shape $\Psi_s = \sum_{j=1}^{2} \sum_{i\in J^j}^{N} p_{T,i}^j (0.1 < r_i^j < 0.2)/(2p_T^j)$

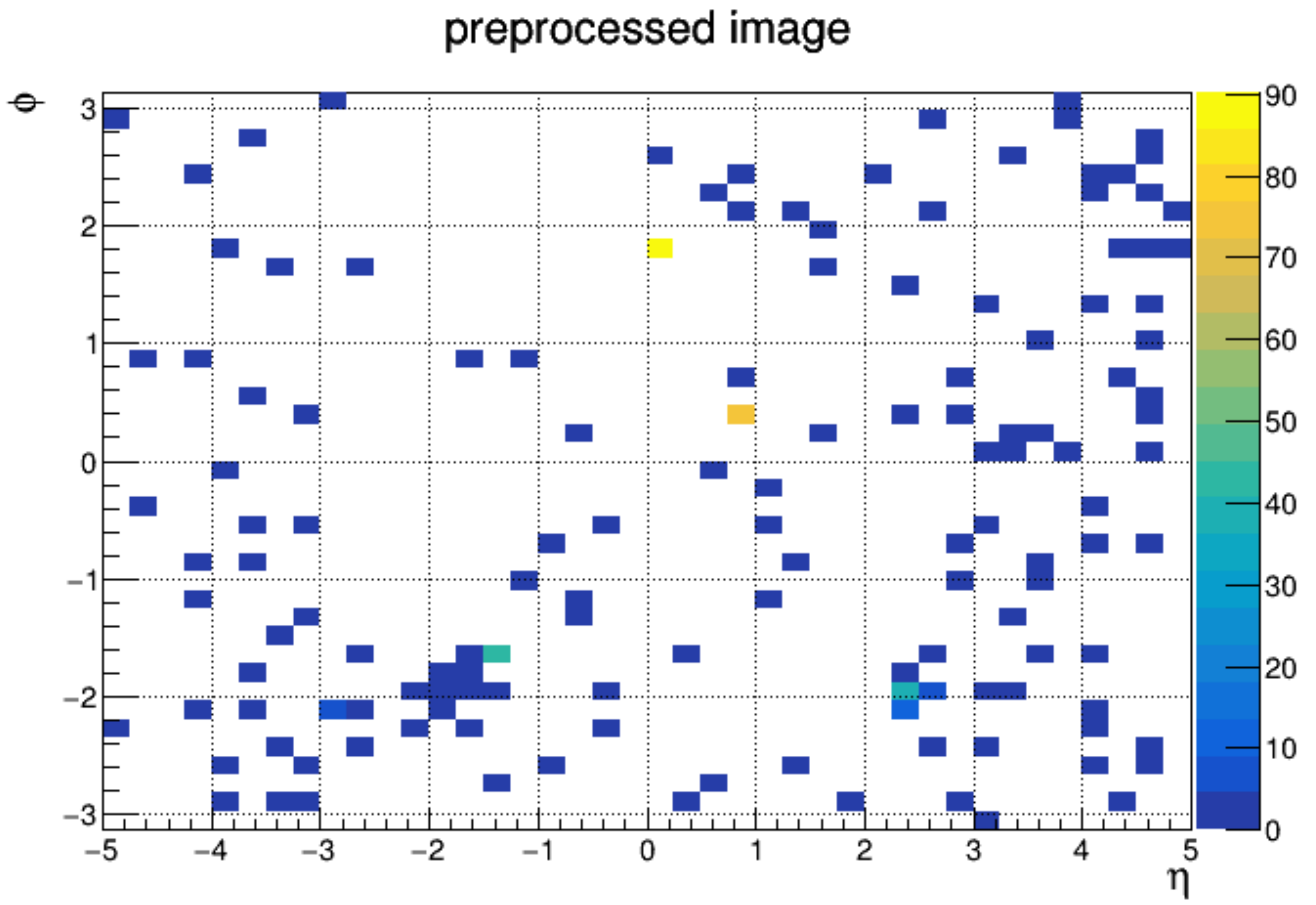
constituent label

distance between the constituent and the jet axis

Event Image Preparation for Event-CNN

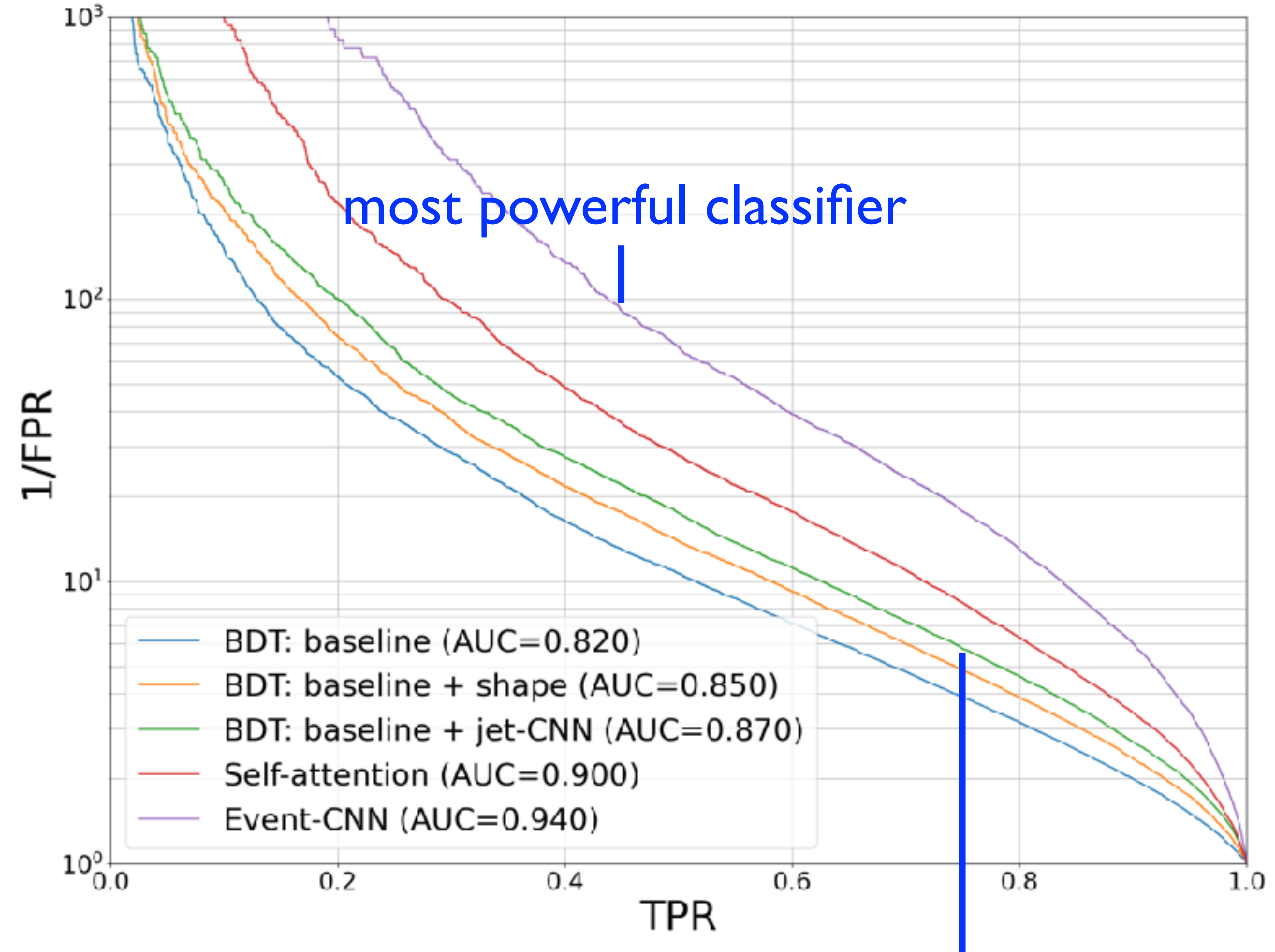
- **Pre-processing**: move the p_T-weighted center to the origin along the ϕ direction, and flip the image vertically or horizontally to make the upper-right quadrant more energetic than all the others ➟ standardize the images
- **Pixelation**: from detector responses into 40×40 pixels
- **6 channels**: Tower E_T, Tower hits, Track E_T, Track hits, Photon E_T, and Photon hits





Comparison of Classifiers

ROC curves (Receiver Operating Characteristic curves)



area under the ROC curve

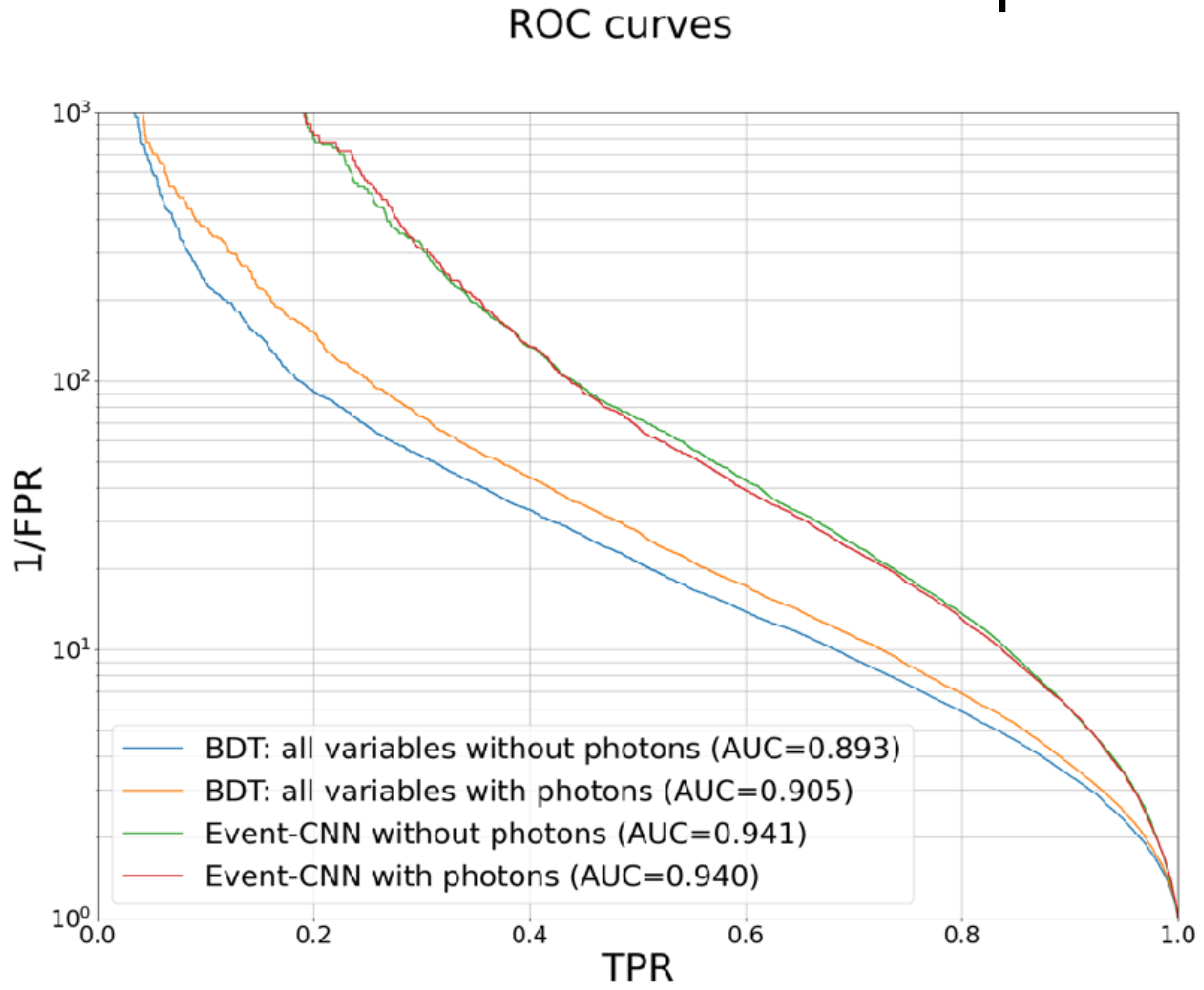
	FPR	ACC	AUC
BDT: baseline	0.035	0.593	0.820
BDT: baseline + shape	0.597	0.027	0.850
BDT: baseline + jet-CNN	0.022	0.599	0.870
Self-attention	0.010	0.604	0.900
Event-CNN	0.003	0.607	0.940

Performance comparison at TPR = 0.3

- Our jet-CNN score is more useful than jet shape variables.
- Tried the combination of jet shapes and jet-CNN scores, but did not make any further improvement.
  ➠ jet-CNN has learned the information contained in the human-engineered jet shape variables

Removal of Photon Information

- Using the **diphoton** mode as an explicit example, we show that the information of the two photons does not affect the performance of the classifier.
- A comparison of performance for **BDT: all variables** and **event-CNN** with and without the information of the photon pair is given as follows.



- Could train a single VBF vs. GGF classifier that is agnostic to the Higgs decay mode.
- Could be applied to a variety of Higgs decay channels in a uniform way.
- Could have benefits for data-driven calibration and reducing systematic uncertainties.

Weak Supervision

Collider Simulations

- Particle experimentalists deal with real data collected by detectors around colliders.
  ➠ just like analyzing real images for CS people
  ➠ even current multivariate approaches for classification rely on simulations and must be corrected later on using data-driven techniques

https://www.catbreedslist.com/stories/what-breed-of-cat-is-garfield.html

- As particle theorists, we think we are simulating verisimilar data using various packages.
  ➠ in fact, we have been generating **fake data** all along
  ➠ problems: fixed-order in perturbation (e.g., CalcHEP, MadGraph), model-dependent showering/hadronization (e.g., Pythia, Herwig), crude detector simulations (e.g., Delphes, GEANT)

https://en.wikipedia.org/wiki/Garfield_(character)

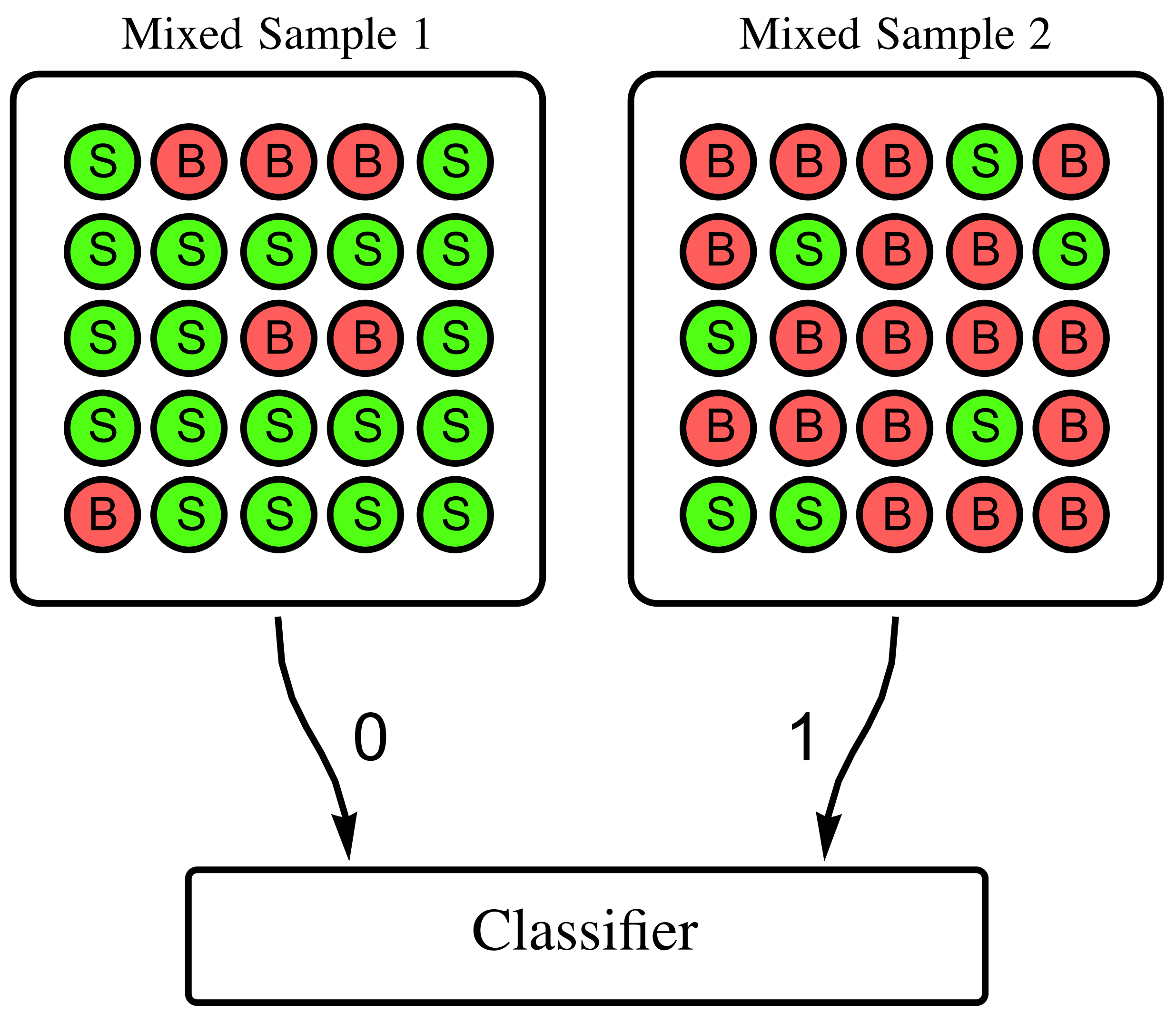
Can We Be More Realistic?

- Using **adversarial networks**? Louppe, Kagan, Cranmer 2016
  ➠ can alleviate model dependence during training, but at the cost of algorithmic performance and computational resources

- It would be nice to train directly using real data.
  ➠ but real data are **unlabeled**…

- Introduce **classification without labels** (**CWoLa**). Metodiev, Nachman, Thaler 2017
  ➠ belonging to a broad framework called **weak supervision**, whose goal is to learn from *partially* and/or *imperfectly labeled* data Herna´ndez-Gonz´alez, Inza, Lozano 2016
  ➠ first weak supervision application in particle physics for quark vs gluon tagging using only class proportions during training; shown to match the performance of fully supervised algorithms Dery, Nachman, Rubbo, Schwartzman 2017

A Theorem for CWoLa



Metodiev, Nachman, Thaler 2017

- Let $\vec{x}$ represent a list of observables or an image, used to distinguish signal S from background B, and define:
  - $p_S(\vec{x})$: probability distribution of $\vec{x}$ for the signal,
  - $p_B(\vec{x})$: probability distribution of $\vec{x}$ for the background.
- Given mixed samples M_1 and M_2 defined in terms of pure events of S and B (both being *identical* in the two mixed samples) using

$$p_{M_1}(\vec{x}) = f_1 p_S(\vec{x}) + (1 - f_1)\, p_B(\vec{x})$$
$$p_{M_2}(\vec{x}) = f_2 p_S(\vec{x}) + (1 - f_2)\, p_B(\vec{x})$$

with *different* signal fractions $f_1 > f_2$, an *optimal classifier* (most powerful test statistic) trained to distinguish M_1 from M_2 is also optimal for distinguishing S from B.

Remarks

- An important feature of CWoLa is that, unlike the learning from label proportions (LLP) weak supervision, the label proportions f_1 and f_2 are **not required** for training as long as they are *different*.
- This proof only guarantees that the optimal classifier from CWoLa, if reached, is the same as the optimal classifier from fully-supervised learning.
- Just like most cases, successful training for CWoLa also requires **a large amount of samples**.

- What happens if available data for the mixed samples are **insufficient or limited**, as is often the case of real data for BSM searches?

Dark Valley Model — Application of CWoLa

Dark Valley Model and Dark Jets

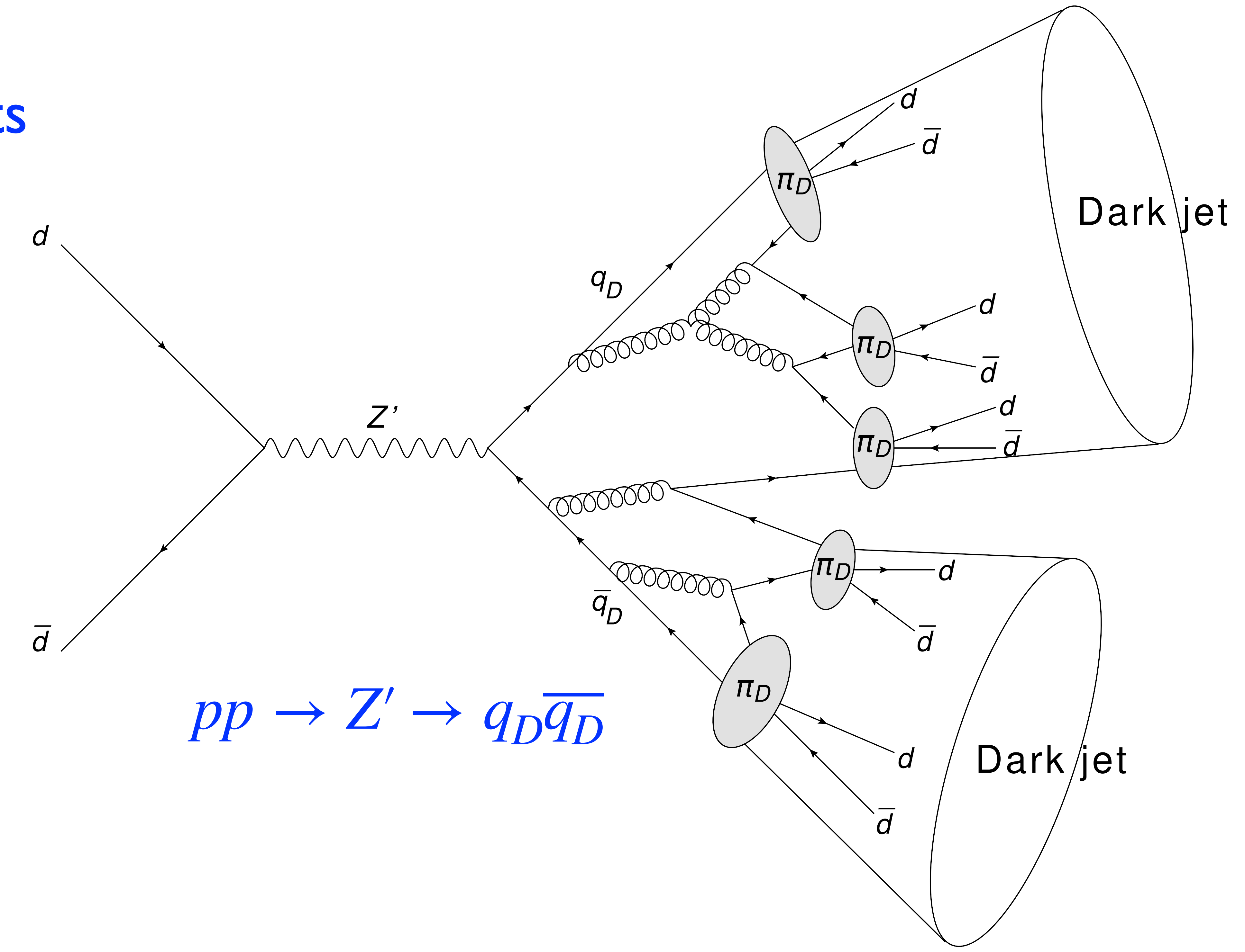
- Assume the existence of a **dark confining sector** that communicates with the visible sector via a **heavy Z' portal**:

dark quarks

$$\mathcal{L} \supset -Z'_\mu \left(g_q \overline{q_i} \gamma^\mu q_i + g_{q_D} \overline{q_{D\alpha}} \gamma^\mu q_{D\alpha} \right)$$

respective effective coupling constants

- For our purposes here, we
  - consider Z' couplings to the d-quarks only, though other SM particles are also possible;
  - give Z' a mass without specifying its source;
  - will not worry about such issues as anomaly cancellation and $Z - Z'$ mixing.



Courtesy of Hugues Beauchesne

- The LHC signature is **a pair of dark jets** with invariant mass consistent with $m_{Z'}$.

Dark Sector Parameter Choices

- The Z' **mass** is fixed at 5.5 TeV, and its **width** is fixed at 10 GeV.
  ➠ invariant mass of the two leading jets being around 5.2 TeV (with some constituents falling outside the reconstructed jets)

- The **dark confining scale** $\Lambda_D \in \{1,\ 5,\ 10,\ 20,\ 30,\ 40,\ 50\}$ GeV.

- Dark vector ρ_D and pseudoscalar π_D masses and two (prompt) decay scenarios:

Albouy et al 2022

$$\frac{m_{\rho_D}}{\Lambda_D} = \sqrt{5.76 + 1.5\frac{m_{\pi_D}^2}{\Lambda_D^2}}$$

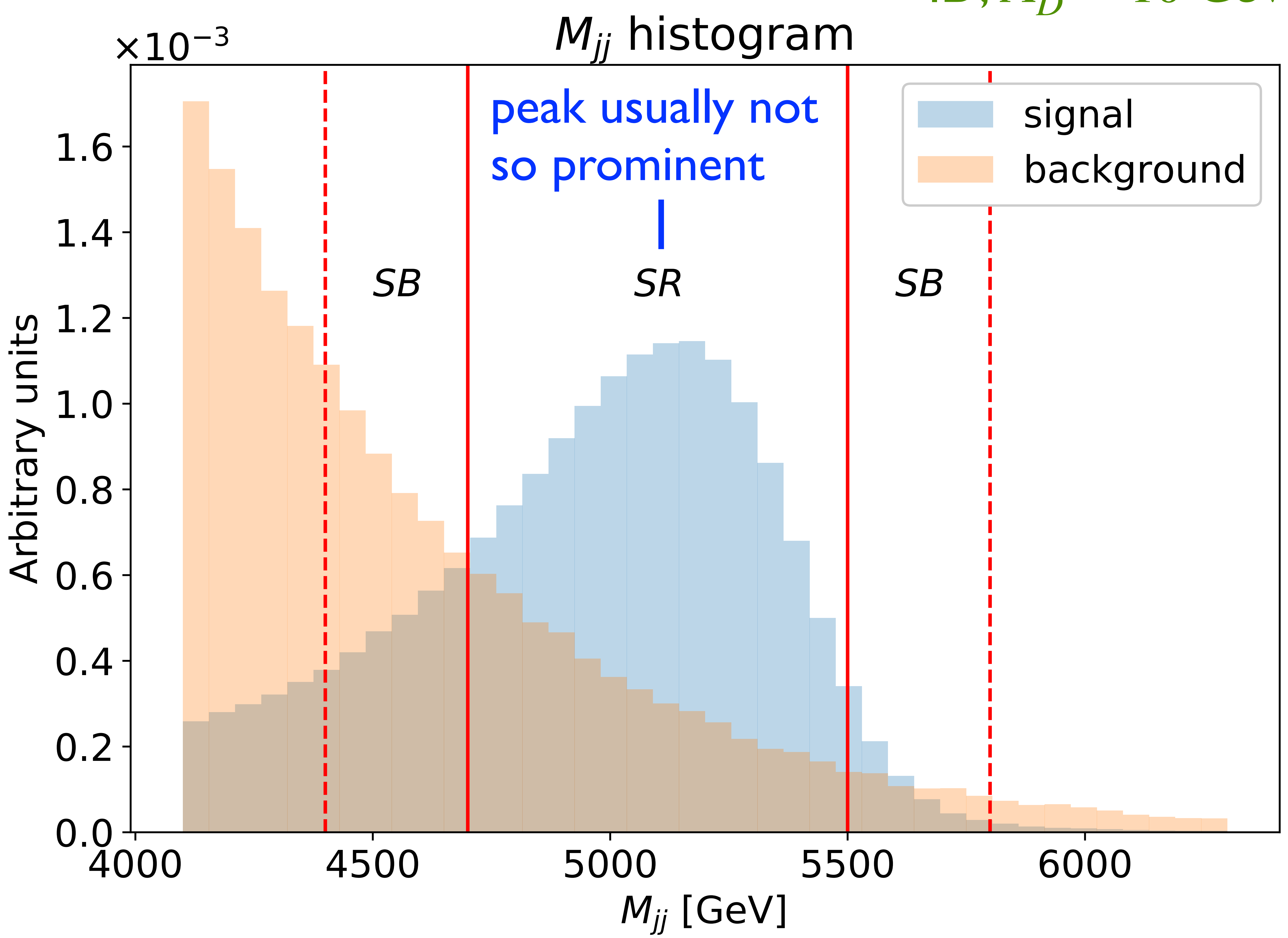
  - **Indirect Decay (ID)**: $\rho_D \to \pi_D \pi_D$ followed by $\pi_D \to d\bar{d}$ for $m_{\pi_D}/\Lambda_D = 1.0$
  - **Direct Decay (DD)**: $\rho_D,\ \pi_D \to d\bar{d}$ for $m_{\pi_D}/\Lambda_D = 1.8$

- Totally **14 "models"** from different combinations of the above parameters.

Dijet Invariant Mass Distributions

- Madgraph 2.7.3 with PDF = NN23LO1
- Pythia 8.307 with default settings
- Delphes 3.4.2 with default CMS card and jet radius $R = 0.8$

ID; $\Lambda_D = 10$ GeV



SR: signal region
SB: side-band region
➠ two mixed samples (M_1 and M_2) with different signal/background fractions

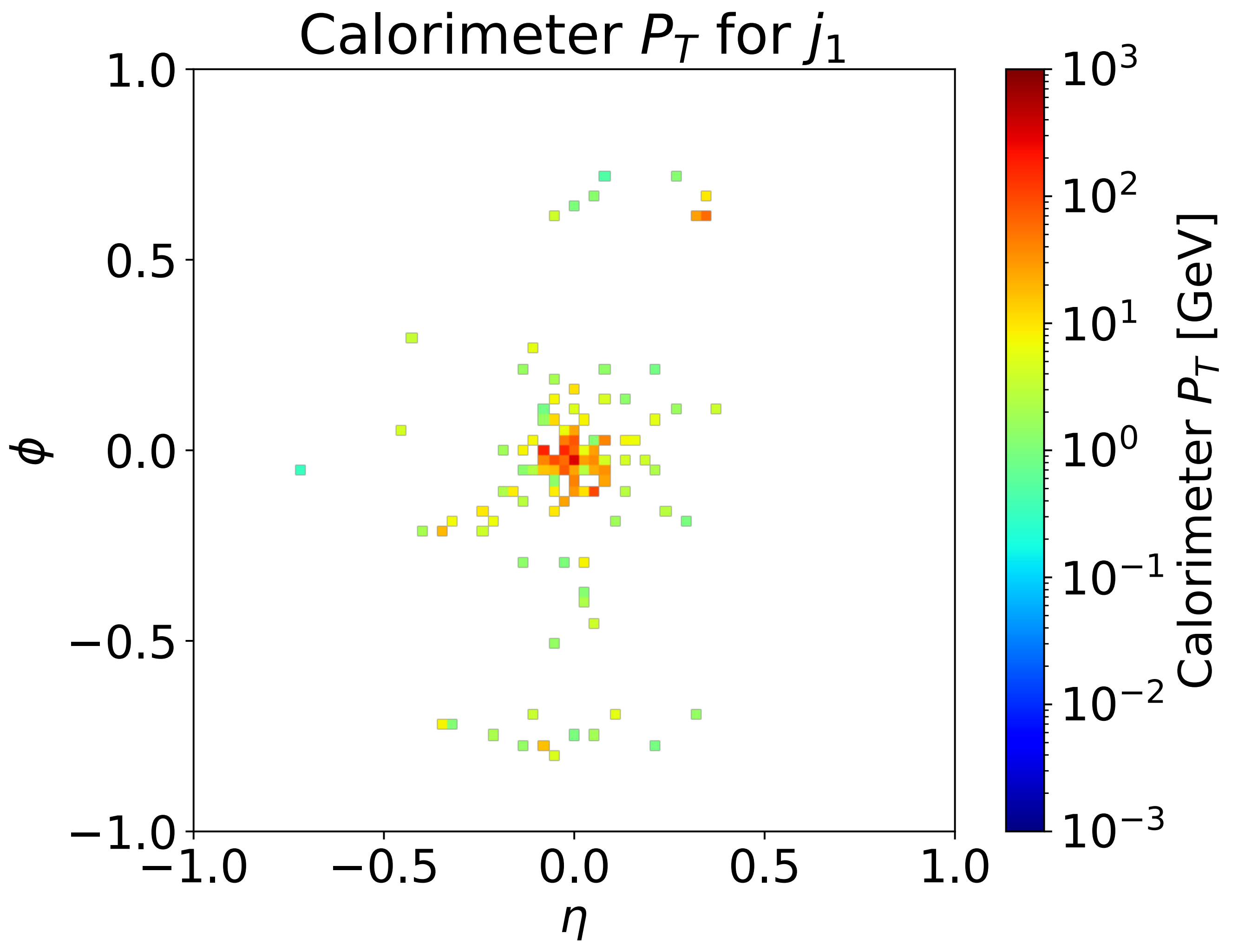
Signal and background events are assumed to be the same in both SR and SB, which should be valid to a good approximation.

Figure 1. Dijet invariant mass distributions for the indirect decaying scenario with $\Lambda_D = 10\,\text{GeV}$ and for the SM background. Distributions are normalized to unity. Both signal and background satisfy the selection criteria of table 1(b) except for the SR or SB conditions.

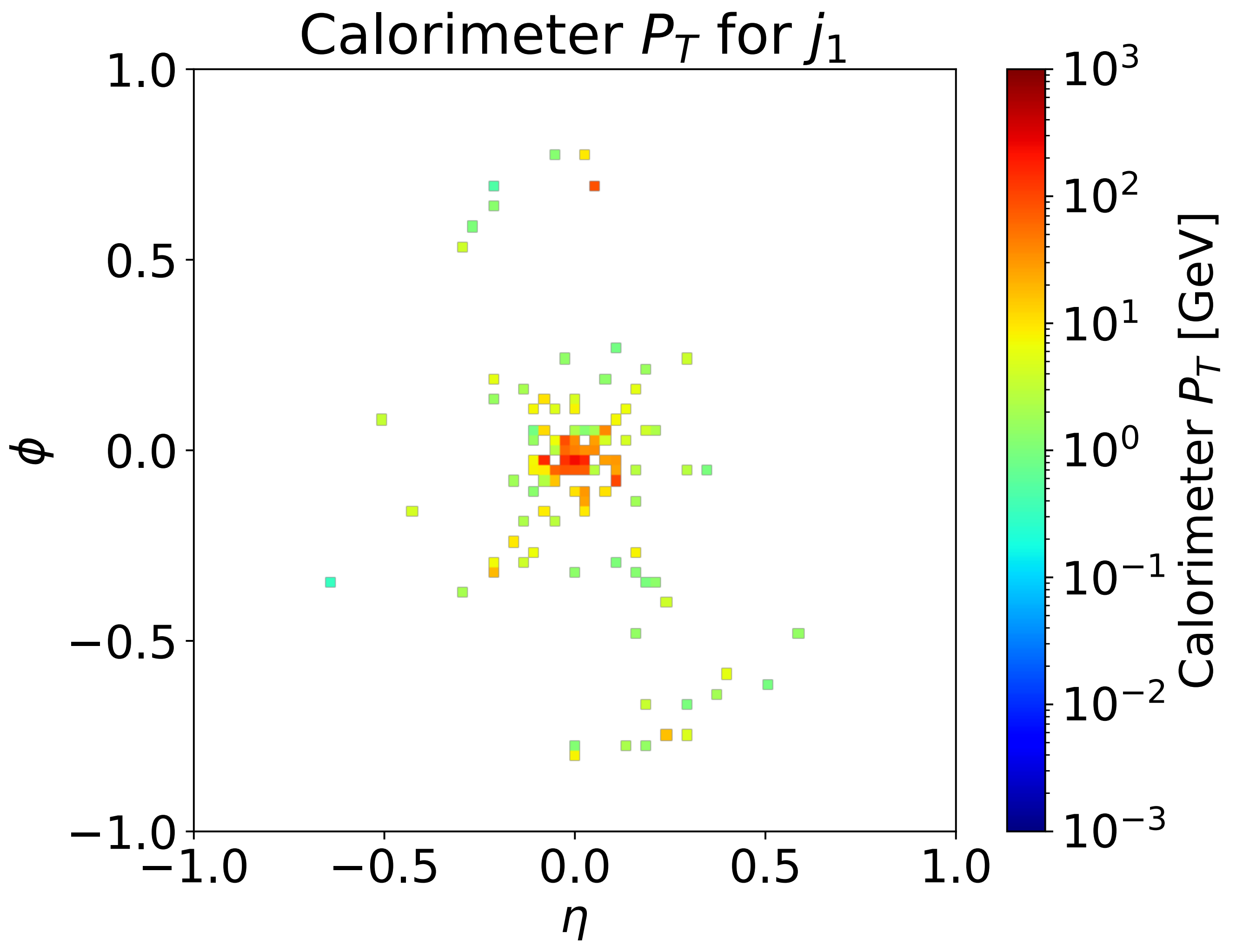
Jet Images Before/After Preprocessing

$\Lambda_D = 10$ GeV
Resolution = 75×75

Image of one signal jet in SR



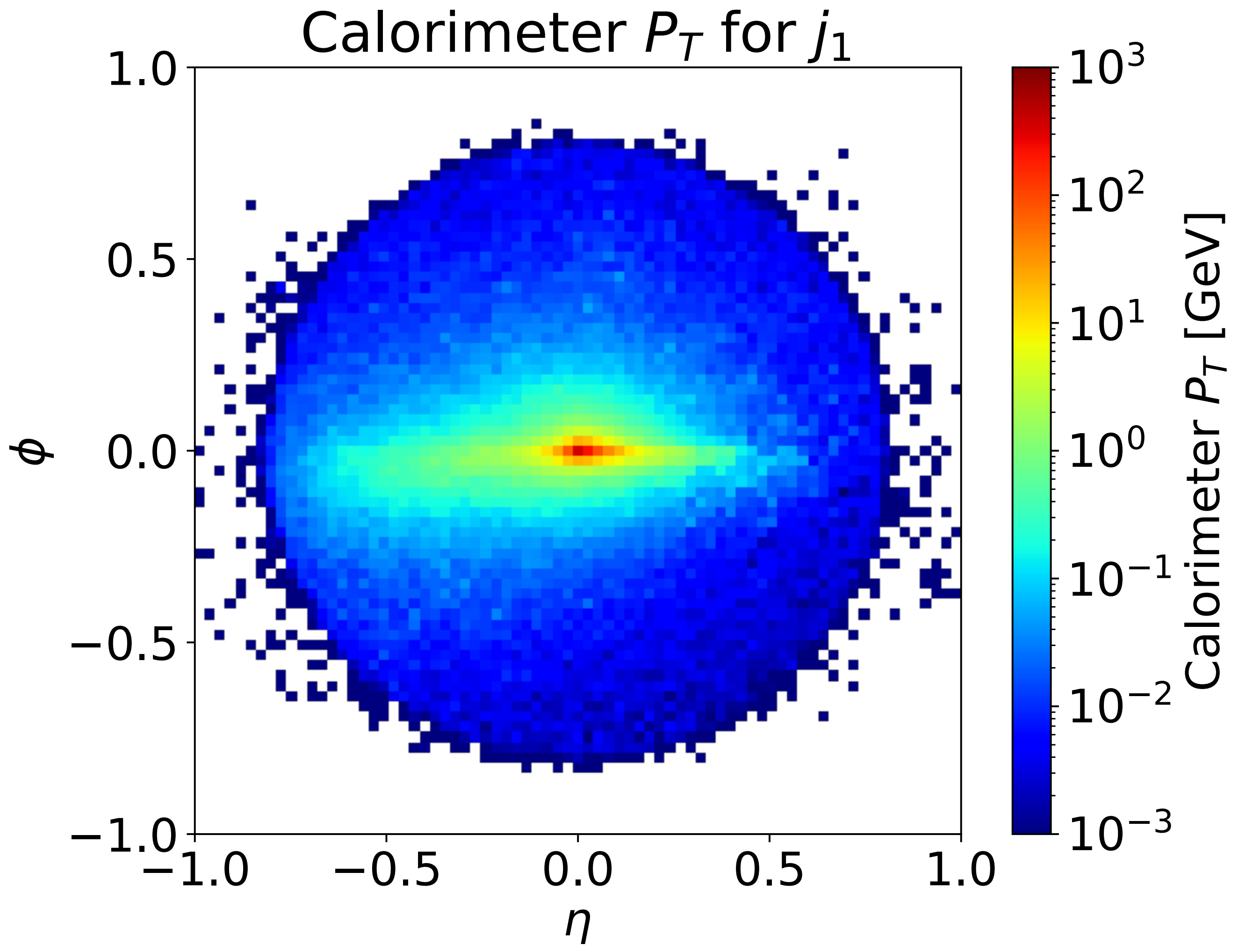
(a) Before preprocessing.



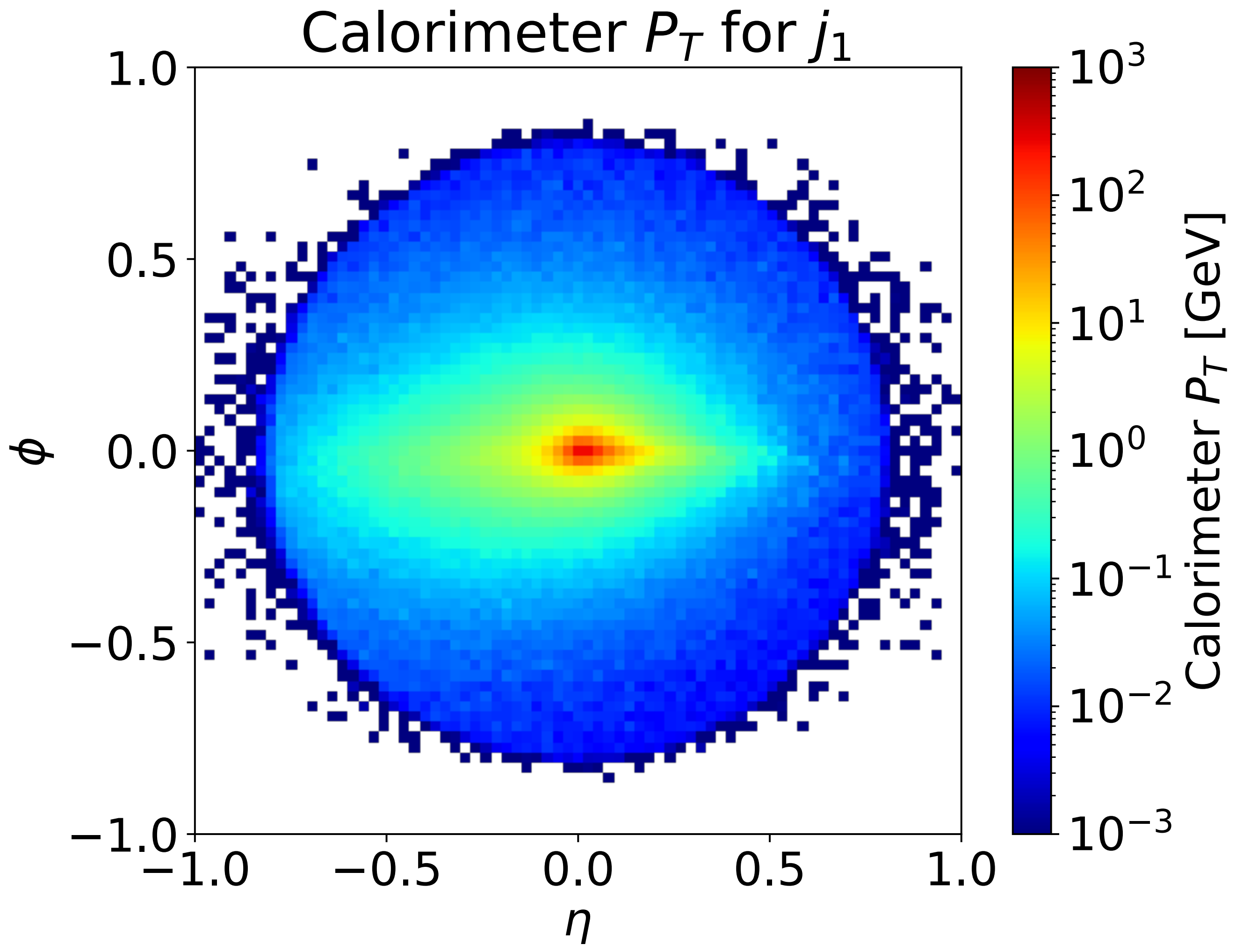
(b) After preprocessing.

Processing:
- shift jet axis to origin
- rotate the principal axis to the horizontal direction
- flip the strongest component to the first quadrant

Average jet image of 10k events



(c) Average histogram of background.



(d) Average histogram of signal.

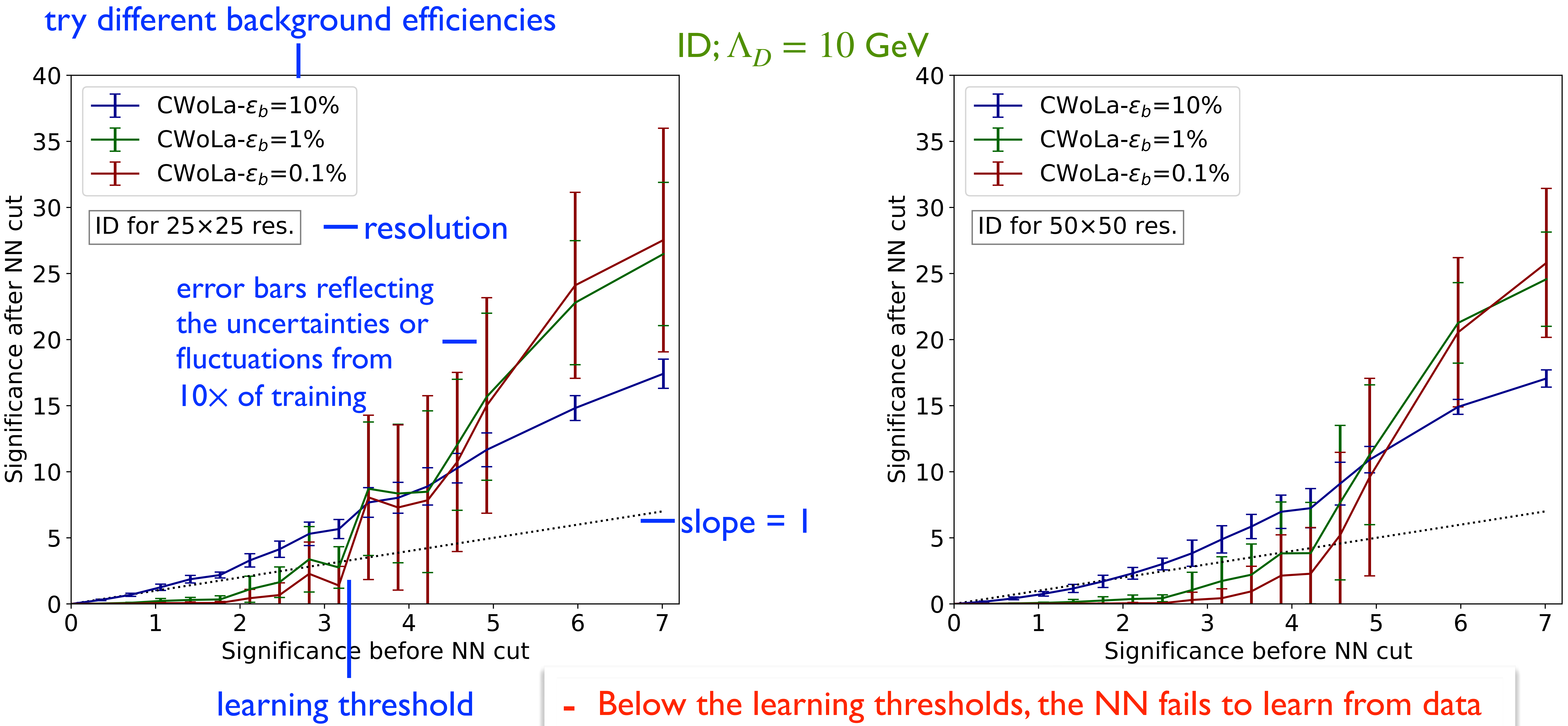
CNN + Dense Layers

- Prepare each jet image in **three resolutions**: 25×25, 50×50, 75×75.
- Use the **images of the two leading jets** as input data.
- Pass each image through a **common** CNN*, and each returns a score $\in [0,1]$.
- Take the **product** of these two scores as the output of the full NN.

- The convolutional part of the NN is referred to as the **feature extractor**, and its weights and biases are collectively labeled as Θ.
  ➠ to be transferred later
- The weights and biases of the dense layers are collectively labeled as θ.
  ➠ to be fine-tuned later

* All NNs are implemented using `Keras` with `TensorFlow` backend. Also, using two distinct networks for the two jets would give slightly inferior results, possibly caused by the lack of signal.

Results of Regular CWoLa



- Below the learning thresholds, the NN fails to learn from data because it cuts background and signal indiscriminately, resulting in a significance even worse than without employing the NN.
- Increasing resolution tends to shift the thresholds higher because more parameters are to be learned inside the NN.

Transfer Learning

Introduction to Transfer Learning

- The phrase “**transfer learning (TL)**” comes from psychology.
  ➠ a learner new to a fresh topic (e.g., playing violin or riding a motorcycle) typically has a higher learning threshold, while a learner experienced in related topics, even if different, (e.g., playing piano or riding a bicycle) usually has less difficulty in quickly picking it up
- As an ML technique, TL reuses a **pre-trained model** developed for one task as the starting point of a new model for a new task.
  ➠ transferring knowledge or experience extracted in the pre-trained model for a **source task/domain** to a new model for a **target task/domain**
  ➠ weights from the pre-trained model used to initialize those of the new model
- TL would only be successful when the features learned from the first model trained on its task can be *generalized* and *transferred* to the second task.
  ➠ dataset in the second training should be sufficiently similar to those in the first training

Pre-training and Fine-tuning

- **Pre-training**:
  - A neural network would first be trained on a larger dataset (source data) based upon *simulations*, which are only required to be sufficiently realistic but not necessarily faithful, to either learn certain concepts or become a more **efficient learner**.
- **Fine-tuning**:
  - The pre-trained model is subsequently trained on a new and possibly smaller dataset (target data), such as the actual data.

Transfer Learning by Pre-training and Fine-tuning

- **Step 1**: The NN is first trained to distinguish a sample of pure background from a pure combination of different signals, which includes all the models mentioned before (ID and DD, different values of Λ_D), except the benchmark on which the model will be tested.
  - ➠ **pre-training** on a large set of simulations as the **source data**
  - ➠ 200k S and 200k B events in the SR for training
    + 50k S and 50k B events for validation
  - ➠ training both Θ (from convolutional layers) and θ (from dense layers)

Layers of CNN subnetwork	$\begin{pmatrix}\text{convolutional 2D layer: 64 filters with } 5\times 5 \text{ kernel size} \\ \text{maxpooling layer: } 2\times 2 \text{ pool size}\end{pmatrix}\times 2$
	convolutional 2D layer: 128 filters with 3×3 kernel size
	maxpooling layer: 2×2 pool size
	convolutional 2D layer: 128 filters with 3×3 kernel size
	flatten layer
	(dense layer: 128 units) $\times 3$
	dense layer (output): 1 unit

Θ

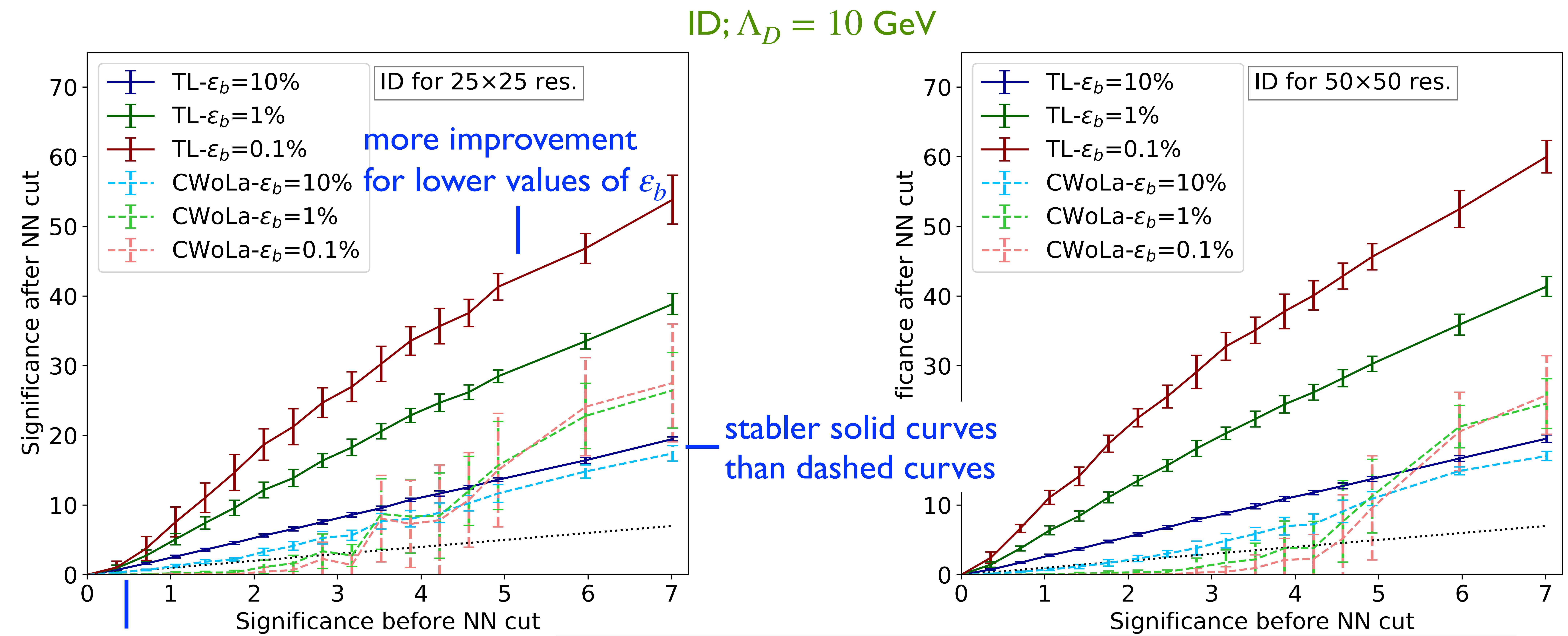
θ

Transfer Learning by Pre-training and Fine-tuning

- **Step 2**: The NN is then trained to distinguish the mixed samples (i.e., the SR and SB regions) using the actual data of the benchmark signal (of the true model) plus the SM background.
  ➠ **fine-tuning** on the actual data as **target data**
  ➠ freezing Θ in the convolutional layers and reinitializing and training θ in the dense layers
  ➠ fixing the feature extraction part while training the classification part

Layers of CNN subnetwork	$\begin{pmatrix} \text{convolutional 2D layer: 64 filters with } 5 \times 5 \text{ kernel size} \\ \text{maxpooling layer: } 2 \times 2 \text{ pool size} \end{pmatrix} \times 2$	
	convolutional 2D layer: 128 filters with 3×3 kernel size	
	maxpooling layer: 2×2 pool size	Θ
	convolutional 2D layer: 128 filters with 3×3 kernel size	
	flatten layer	
	(dense layer: 128 units) $\times 3$	θ
	dense layer (output): 1 unit	

Transfer Learning vs Regular CWoLa



lower learning thresholds for TL

- The amount of signal necessary to claim a 5σ discovery can be reduced by a factor of a few, which is due to the fact that the NN can better keep the signals.
- Fluctuations in the significance are reduced, due to a smaller amount of trainable parameters and more successful learning.

Summary

- **Weak supervision** techniques (CWoLa) have the advantages of being able to **train on real data** and of exploiting distinctive signal properties.
  ➠ ideal tools for **anomaly searches**
  ➠ fail when signals are **limited**
- We propose to use the **Transfer Learning** approach.
  - First, train an NN on simulations for **pre-training**.
  - Then, train the NN **on real data**, where signals may be scarce.
  - Use **scaling** and **shifting** parameters to obtain a better learner.
- **TL** can **drastically improve** the performance of CWoLa searches, particularly in the **low-significance region**, and the amount of signal required for discovery can be reduced by a factor of a few (because of better identification of signals).
- **Meta Transfer Learning** can only **slightly improve** the performance.

Thank You!