Published for SISSA by Springer

Received: *April 7, 2020*
Revised: *May 21, 2020*
Accepted: *June 19, 2020*
Published: *July 17, 2020*

# Neural network-based top tagger with two-point energy correlations and geometry of soft emissions

JHEP07(2020)111

**Amit Chakraborty,**[a] **Sung Hak Lim,**[b] **Mihoko M. Nojiri**[b,c,d] **and Michihisa Takeuchi**[e]

[a] *Centre for High Energy Physics, Indian Institute of Science,*
*Bengaluru, Karnataka 560012, India*
[b] *Theory Center, IPNS, KEK,*
*1-1 Oho, Tsukuba, Ibaraki 305-0801, Japan*
[c] *The Graduate University of Advanced Studies (Sokendai),*
*1-1 Oho, Tsukuba, Ibaraki 305-0801, Japan*
[d] *Kavli IPMU (WPI), University of Tokyo,*
*5-1-5 Kashiwanoha, Kashiwa, Chiba 277-8583, Japan*
[e] *Kobayashi-Maskawa Institute for the Origin of Particles and the Universe, Nagoya University,*
*Furo-cho, Chikusa-ku, Nagoya, Aichi 464-8602, Japan*

*E-mail:* achakraborty@iisc.ac.in, sunghak.lim@kek.jp, nojiri@post.kek.jp, takeuchi@kmi.nagoya-u.ac.jp

Abstract: Deep neural networks trained on jet images have been successful in classifying different kinds of jets. In this paper, we identify the crucial physics features that could reproduce the classification performance of the convolutional neural network in the top jet vs. QCD jet classification. We design a neural network that considers two types of sub-structural features: two-point energy correlations, and the IRC unsafe counting variables of a morphological analysis of jet images. The new set of IRC unsafe variables can be described by Minkowski functionals from integral geometry. To integrate these features into a single framework, we reintroduce two-point energy correlations in terms of a graph neural network and provide the other features to the network afterward. The network shows a comparable classification performance to the convolutional neural network. Since both networks are using IRC unsafe features at some level, the results based on simulations are often dependent on the event generator choice. We compare the classification results of `Pythia 8` and `Herwig 7`, and a simple reweighting on the distribution of IRC unsafe features reduces the difference between the results from the two simulations.

Keywords: Jets, QCD Phenomenology

ArXiv ePrint: 2003.11787

Article funded by SCOAP$^3$.

https://doi.org/10.1007/JHEP07(2020)111

## Contents

## 1 Introduction

Interest in deep learning in collider physics [1–5] has been growing in recent years. Many applications of deep learning have appeared in jet classification [6–26], anomaly detection [27–37], particle identification [38–40], pileup mitigation [41–43], event generation [44–58], unfolding [59, 60], and parton distribution functions [61–78]. Deep learning will be used more in the analysis of LHC run III data. Among those, jet classification using neural networks is one of the well-established areas. Several approaches have been proposed, and the performance of different models has been compared [79]. For the classification between top jets and QCD jets, neural networks trained on low-level inputs showed a significant improvement in the classification performance compared to the previous methods [80].

Before the deep learning in jet classification, the classification using the jet substructure information achieved remarkable success. The particles coming from the decay of a boosted heavy particle give clear substructures inside the reconstructed jet. The substructure maybe characterized by various manners; for example, by going through the jet clustering sequence [81–89], reclustering jet constituents into the jets with smaller radius to identify subjets [81, 90–93], or the energy correlations [94–104]. Note that such substructures are often defined by infrared and collinear (IRC) safe algorithms or observables which are theoretically more predictable. The IRC unsafe quantities are also used in the jet classification. For example, the number of charged tracks [99] is very useful quantity for the quark jet vs. gluon jet classification. In some cases, the IRC unsafe counting variable has an IRC safe counterpart such as soft drop multiplicity [105].

The pattern of soft radiation is also important for the classification. For example, a color singlet boosted heavy particle has emission isolated in terms of soft activity unlike quark and gluon jets. The related substructure quantity has been incorporated in Higgs taggers [106, 107] and top taggers [108]. Such soft particle distribution may also contribute to the jet classification using neural networks in order to improve the performance.

While the improvement using deep learning is impressive, the physics behind it has not been addressed. So far, the classifier based on a convolutional neural network (CNN) trained on the jet image performs well for selecting the top jets. It is numerically shown that the CNN uses IRC safe features mostly [109], but it is not easy to make an estimate of systematic uncertainties from various sources without knowing what kind of features of the jet is used in the model. Bayesian networks are capable of tracking those uncertainties [110, 111], but it is also useful to identify the features in order to interpret the network outputs and uncertainties. The aim of this paper is to provide a convenient parametrization of the jet feature contributing to the classification using jet images.

In this paper, we address the question in the following steps. In section 2, we first introduce a graph neural network [112–116] with constraints, and the network is more restrictive than CNN. Graph networks are flexible enough for analyzing multiple objects appears at the LHC, and have been studied in various contexts [16, 20, 25, 41, 117–124]. The graph network in this paper has access to only IRC safe two-point energy correlations [19, 21, 96–98, 125–128]. It was shown that the network has comparable performance to the CNN in the Higgs jet vs. QCD jet classification [21]. We use this network for top jet vs. QCD jet classification, and it is a good starting point toward the network whose top tagging performance is comparable to the CNN.

To integrate the IRC unsafe quantities to this framework, we formulate a sequence of novel morphological measures based on Minkowski functionals, in section 3. The sequence includes the number of pixels with finite energy deposit (active pixels), $N^{(0)}$, the number of pixels that touch the active pixels, $N^{(1)}$. These numbers can be considered as a discretized version of Minkowski functionals. They are formulated in a mathematical theory called integral geometry and describes geometric measures to the point distributions. The application of the Minkowski functionals has already been considered in the astrophysical analysis [129–142], and statistical mechanics [143–145]. We perform a morphological analysis to the distribution of soft activity in the jet.

When the first few elements of the Minkowski sequence are included in the graph network inputs, the new classifier has the same performance as the jet image CNN classifier, as shown in section 4. This means that the improvement of the CNN classifier comes from the geometric quantities of the pixels, and also it is summarized by just a few numbers of additional variables. Our result suggests that the CNN output is correlated to a few numbers of geometric quantities derived from the jet image.

In the collider study, event simulators are used extensively to estimate the signal and background distributions. The sequence of Minkowski functionals calculated from a jet image is IRC unsafe quantities, and the simulated data need to be calibrated by the experimental data. We propose an event reweighting method based on the IRC unsafe quantities for the calibration in section 5. We conclude in section 6.

## 2 IRC safe two-point energy correlations and relation network

The jet classifier using a deep learning model trained on the jet image has achieved better performance compared with the other statistical methods. Still, it is not straightforward to identify the key physical features that contributed to the improvement, other than looking for the hidden data representations of the CNN [7, 11, 28, 146], or checking the response of the network after perturbing the inputs [109]. Note that organized networks whose hidden representations have physical interpretations [9, 16, 18, 21, 24] allow us to interpret the results in terms of physics. For this purpose, we consider flexible and interpretable quantities derived from the jet image and use them as inputs to a jet classifier modeled by a multilayer perceptron (MLP). Additional inputs are considered until the performance of the classifier is equivalent to that of the best classifiers using the jet image.

We first introduce two-point energy correlation spectra $S_2$ [19, 21] as a function of the distance between the jet constituents $R$,

$$S_{2,\mathbf{J}_a\mathbf{J}_b}(R) = S_{2,ab}(R) = \int d\vec{R}_1 d\vec{R}_2\, P_{T,\mathbf{J}_a}(\vec{R}_1) P_{T,\mathbf{J}_b}(\vec{R}_2)\, \delta(R - R_{12}), \tag{2.1}$$

$$S_2(R) = S_{2,\mathbf{JJ}}(R), \tag{2.2}$$

where $S_{2,ab}$ is a shorthand notation of $S_{2,\mathbf{J}_a\mathbf{J}_b}$;

$$P_{T,\mathbf{J}_a}(\vec{R}) = \sum_{i\in\mathbf{J}_a} p_{T,i}\, \delta(\vec{R} - \vec{R}_i), \tag{2.3}$$

is an energy flow of a subjet $\mathbf{J}_a$ of a jet $\mathbf{J}$; $a$ and $b$ are indices of the subjet. The $R_{ij}$ is the relative angular distance between two constituents, $\sqrt{(\eta_i - \eta_j)^2 + (\phi_i - \phi_j)^2}$. The $S_{2,ab}$ is an IRC safe quantity. For the Higgs jet vs. QCD jet classification, an MLP trained on the transverse momenta, masses, and $S_2$'s of the jet and trimmed jet performs nearly as good as a CNN trained on jet images. In [21], we relate $S_{2,ab}$ to the generic jet classifiers through its formal expansion with respect to the energy flow. This shows that $S_{2,ab}$ is flexible enough to describe many quantities for the classification of jets.

In this section, we first derive $S_{2,ab}$ in terms of a vertex-labeled fully-connected graph to integrate them into a framework of the graph network and extend it for further ML

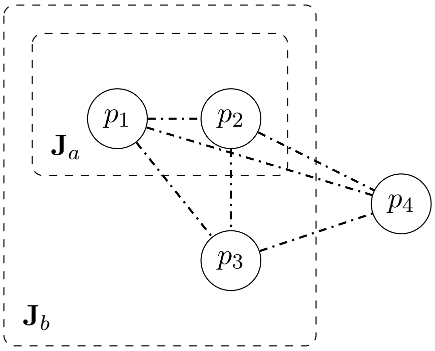


**Figure 1.** A schematic diagram of the graph representation of a jet used in this paper. Each vertex corresponds to a jet constituent, and a line between two circles represents the variable calculated from the two vertices. Each dashed rectangle represents a subjet that contains the enclosed jet constituents.

analysis. A graph is a set of the points and the lines connecting them, which are called vertices and edges, respectively. In our setup, each vertex of the graph corresponds to a jet constituent, and the inputs to the $i$-th vertex are the jet constituent momentum $p_i$. The labels of a vertex denote the subjets to which the constituent $i$ belongs. The graph network also has the other inputs $\boldsymbol{u}$ calculated from the given jet, for example, (sub)jet transverse momentum and mass. A schematic diagram of the graph is in figure 1. Each circle represents the jet constituent assigned to the corresponding vertex. The dot-dashed lines are the edges.

We use a kind of graph network called a relation network (RN) [114, 115] that mainly utilizes correlations between two vertices. The reason for using this network is that the kernel of the parton shower model is 1 → 2 splitting of partons. The classifier can focus on the two-point correlations by using the relation network as a functional model. The classifier output $\boldsymbol{u}'$ is the value of a functional model $\phi^u$ applied to the edge outputs $\bar{\boldsymbol{e}}_{ab}$, the vertex outputs $\bar{p}_a$, and the predefined inputs $\boldsymbol{u}$.

$$\boldsymbol{u}' = \phi^u \left(\bar{\boldsymbol{e}}_{ab}, \bar{\boldsymbol{p}}_a, \boldsymbol{u}\right). \tag{2.4}$$

The edge output $\bar{\boldsymbol{e}}_{ab}$ is the aggregated two-point correlation between $\mathbf{J}_a$ and $\mathbf{J}_b$,

$$\bar{\boldsymbol{e}}_{ab} = \sum_{\substack{i\in\mathbf{J}_a\\ j\in\mathbf{J}_b}} \phi^e_{ab}(p_i, p_j, \boldsymbol{u}), \tag{2.5}$$

where $\phi^e_{ab}(p_i, p_j, \boldsymbol{u})$ is a functional model of a two-point correlation assigned on edge linking two jet constituents $i$ and $j$. The vertex output $\bar{\boldsymbol{p}}_a$ is the aggregated one-point correlation of $\mathbf{J}_a$,

$$\bar{\boldsymbol{p}}_a = \sum_{i\in\mathbf{J}_a} \phi^v_a(p_i, \boldsymbol{u}), \tag{2.6}$$

where $\phi^v_a(p_i, \boldsymbol{u})$ is a functional model of a one-point correlation assigned to a vertex that corresponds to a jet constituent $i$. The correlations $\bar{\boldsymbol{p}}_a$ and $\bar{\boldsymbol{e}}_{ab}$ are symmetric for the permutation of the jet constituents. We train $\boldsymbol{u}'$ to be the logits for the classification.[1]

[1] See [116] for other formalism of the graph neural network.

We use energy correlators [94, 103] for $\phi^e$ and $\phi^v$ to restrict $\boldsymbol{u}'$ to be IRC safe. Namely, we consider the following IRC safe $C$-correlators for $\bar{\boldsymbol{p}}_a$ and $\bar{\boldsymbol{e}}_{ab}$,

$$\bar{\boldsymbol{p}}_a = \sum_{i\in\mathbf{J}_a} p_{T,i} w_a(\vec{R}_i;\boldsymbol{u}) \to p_{T,\mathbf{J}_a} w_a(\boldsymbol{u}), \tag{2.7}$$

$$\bar{\boldsymbol{e}}_{ab} = \sum_{\substack{i\in\mathbf{J}_a\\ j\in\mathbf{J}_b}} p_{T,i}p_{T,j} w_{ab}(\vec{R}_i,\vec{R}_j;\boldsymbol{u}) \to \sum_{i\in\mathbf{J}_a}\sum_{j\in\mathbf{J}_b} p_{T,i}p_{T,j} w_{ab}(R_{ij};\boldsymbol{u}), \tag{2.8}$$

where $p_{T,i}$ is the transverse momentum of the $i$-th constituent, $\vec{R}_i = (\eta_i, \phi_i)$ is the pseudorapidity-azimuthal coordinate of the $i$-th constituent. The functions $w_a$ and $w_{ab}$ are the angular weighting functions of one-point and two-point energy correlators, respectively. The last step of the equation comes from the assumption that the classifier does not depend on the absolute angular coordinates of the (sub)jet constituents but uses the relative angular distances.

The last expression in eq. (2.8) can be written in terms of an integral [19],

$$\bar{\boldsymbol{e}}_{ab} = \int dR\, S_{2,ab}(R) w_{ab}(R,\boldsymbol{u}). \tag{2.9}$$

We may absorb the angular weighting functions $w_{ab}$ to $\phi^u$ so that the $S_{2,ab}$ and $p_{T,\mathbf{J}_a}$ can be considered as effective inputs to the network.

$$\boldsymbol{u}' = \phi^u \left(S_{2,ab}(R), p_{T,\mathbf{J}_a}, \boldsymbol{u}\right). \tag{2.10}$$

This setup is equivalent to the one using $S_{2,ab}$ as input, discussed in [19].

We now design a top tagger based on eq. (2.10). The structure of the graph is specified by the subjet label $a$ and $b$ of $S_{2,ab}$. We consider the following subjet labels for the top jet vs. QCD jet classification.

- the trimmed jet, $\mathbf{J}_{\text{trim}}$, denoted by $h$,
- the compliment set of $\mathbf{J}_{\text{trim}}$, $\mathbf{J}\setminus\mathbf{J}_{\text{trim}}$, denoted by $s$,
- the leading $p_T$ subjet, $\mathbf{J}_1$, denoted by 1,
- the compliment set of $\mathbf{J}_1$, $\mathbf{J}\setminus\mathbf{J}_1$, denoted by $c$.

Examples of the vertex-labeled graphs are in figure 2. Note that the following relations hold for $S_2$ and $S_{2,ab}$,

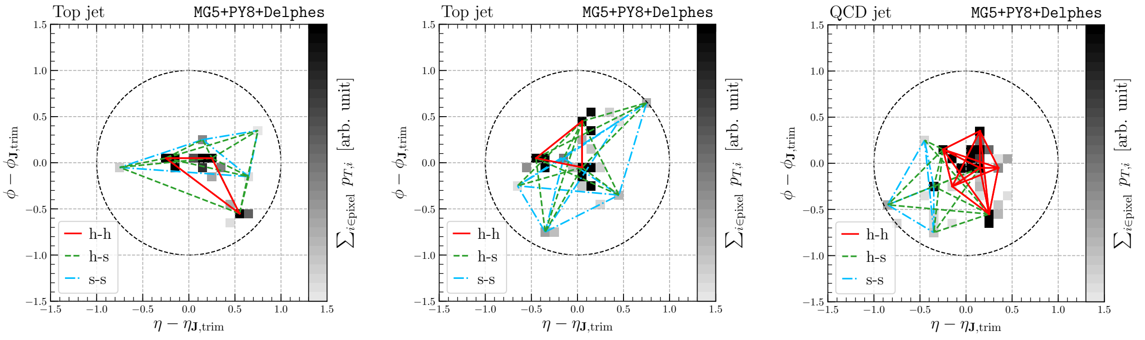
$$S_2(R) = S_{2,hh}(R) + 2S_{2,hs}(R) + S_{2,ss}(R), \tag{2.11}$$

$$= S_{2,11}(R) + 2S_{2,1c}(R) + S_{2,cc}(R). \tag{2.12}$$

Because $S_{2,ss}$ contains only the correlations between soft constituents, which is theoretically unpredictable and less reliable experimentally, we define the following combinations as in [19].

$$S_{2,\text{trim}}(R) = S_{2,hh}(R), \tag{2.13}$$

$$S_{2,\text{soft}}(R) = 2S_{2,hs}(R) + S_{2,ss}(R). \tag{2.14}$$



**(a)** **(b)** **(c)**

**Figure 2.** Schematic diagrams of the graph representations of jets. (a) and (b) are top jet images, and (c) is a QCD jet image. Lines represent the graphs on the jet images. The red solid lines are edges between the constituents of the trimmed jet $\mathbf{J}_{\mathrm{trim}}$. The green dashed lines are the edges between the constituents of $\mathbf{J}_{\mathrm{trim}}$ and the constituents of $\mathbf{J} \setminus \mathbf{J}_{\mathrm{trim}}$. The blue dot-dashed lines are edges between the constituents of $\mathbf{J} \setminus \mathbf{J}_{\mathrm{trim}}$. Note that we omitted some edges for readability.

The $S_{2,\mathrm{trim}}$ and $S_{2,\mathrm{soft}}$ distributions of the top jets and QCD jets in figure 2 are shown in figure 3.

In parton level, $S_{2,\mathrm{trim}}$ and $S_{2,\mathrm{soft}}$ of a top quark have up to four peaks of delta functions and written as follows if all partons are sufficiently high $p_T$.

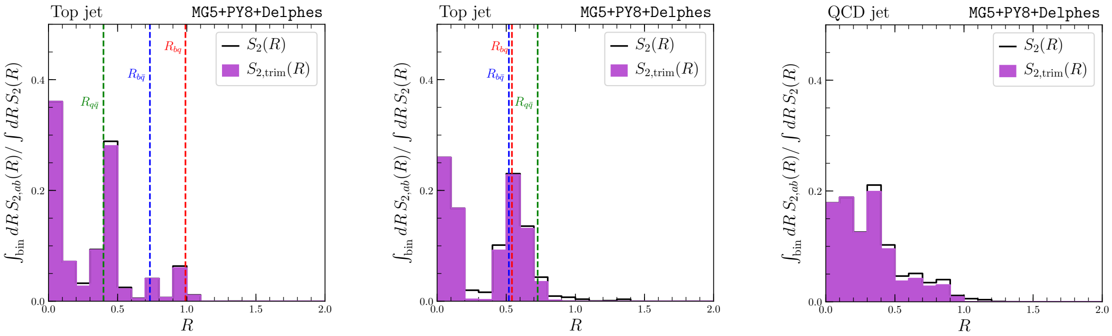
$$
\begin{aligned}
S_{2,\mathrm{trim}}(R) &= (p_{T,b}^2 + p_{T,q}^2 + p_{T,\bar{q}}^2)\,\delta(R) \\
&\quad + 2p_{T,b}p_{T,q}\delta(R - R_{bq}) + 2p_{T,b}p_{T,\bar{q}}\delta(R - R_{b\bar{q}}) + 2p_{T,q}p_{T,\bar{q}}\delta(R - R_{q\bar{q}}), \qquad (2.15) \\
S_{2,\mathrm{soft}}(R) &= 0. \qquad (2.16)
\end{aligned}
$$

Here, $b$ is a bottom quark from a top quark decay, and $q$ and $\bar{q}$ are quarks from the subsequent $W$ boson decay. Figure 3(a) is the $S_{2,\mathrm{trim}}$ of the top jet that has those four peaks clearly. This pattern is relatively rare for QCD jets. Figure 3(c) is the $S_{2,\mathrm{trim}}$ of a typical QCD jet.

In the case where the characteristic angular scales of the top quark, $R_{bq}$, $R_{b\bar{q}}$, and $R_{q\bar{q}}$, are close to each other, it is not possible to see all peak structures in the $S_{2,\mathrm{trim}}(R)$ distributions. Such an example is shown in figure 3(b), although the relative strength of the peaks in the $S_{2,\mathrm{trim}}$ distribution contains partial information of the three-prong structures.[2]

The information of the three-prong substructure is more clearly encoded in $S_{2,11}$, $S_{2,1c}$, and $S_{2,cc}$. The two-point correlations of the top jets corresponding to figure 2(a) and figure 2(b) are shown in figure 4 and figure 5, respectively. This decomposition of a given jet into $\mathbf{J}_1$ and $\mathbf{J} \setminus \mathbf{J}_1$ factorizes the identification of a three-prong structure into that of two-prong substructures and its relative position from the $\mathbf{J}_1$. Those $S_{2,ab}$ in parton level

[2] For example, if all the partons from three-prong decay carry an equal fraction of momenta and their angular distances are the same, the ratio between the intensity of the two peaks is 1:2 in the parton level, while it is 1:1 for a two-prong decay [19].



**(a)** **(b)** **(c)**

**Figure 3.** The $S_2$ and $S_{2,\mathrm{trim}}$ distributions of the top jets and the QCD jet in figure 2. The dashed lines are the characteristic angular scales of the top jets in the parton level.

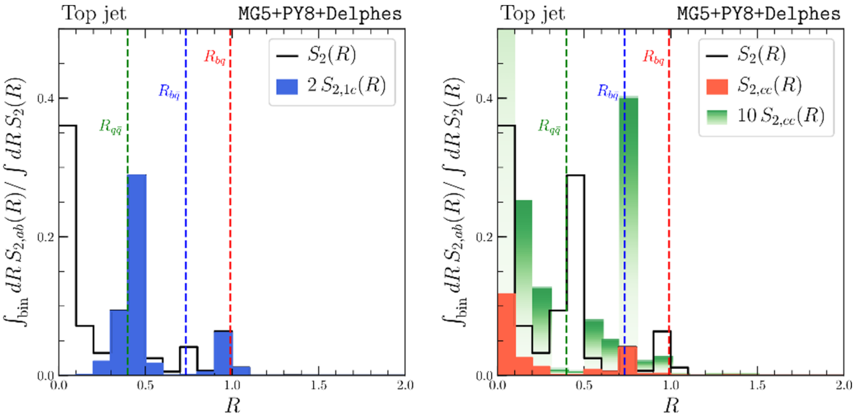


**Figure 4.** $S_{2,1c}$ and $S_{2,cc}$ distributions of the top jet in figure 2(a). The intensity of $S_{2,cc}$ is much smaller than $S_2$ because the subleading $p_T$ jets have small transverse momenta. The magnified distribution of $S_{2,cc}$ is shown in the green histogram. The dashed lines are the characteristic angular scales at the parton level.

are as follows,

$$S_{2,11}(R) = p_{T,i_1}^2 \delta(R), \tag{2.17}$$

$$2\, S_{2,1c}(R) = 2p_{T,i_1} p_{T,i_2} \delta(R - R_{i_1 i_2}) + 2p_{T,i_1} p_{T,i_3} \delta(R - R_{i_1 i_3}), \tag{2.18}$$

$$S_{2,cc}(R) = (p_{T,i_2}^2 + p_{T,i_3}^2)\delta(R) + 2p_{T,i_2} p_{T,i_3} \delta(R - R_{i_2 i_3}), \tag{2.19}$$

where $i_k$ is the $k$-th leading $p_T$ parton. Figure 4 shows that the two peaks are in $S_{2,1c}$ and the other two peaks are in $S_{2,cc}$. Figure 5 is the case where values of $R_{bq}$ and $R_{b\bar{q}}$ are similar. The $S_{2,cc}$ distribution has a peak at $R \approx 0.6$, and the peak intensity is comparable to that of the peak at $R = 0$ because the $\mathbf{J} \setminus \mathbf{J}_1$ has a two-prong substructure. In addition, the $S_{2,1c}$ distribution suggests that the high $p_T$ constituents of $\mathbf{J} \setminus \mathbf{J}_1$ are away from $\mathbf{J}_1$ by a distance of 0.5. Note that the analysis on $S_{2,1c}$ is essentially telescoping jets [147, 148] with respect to $\mathbf{J}_1$.

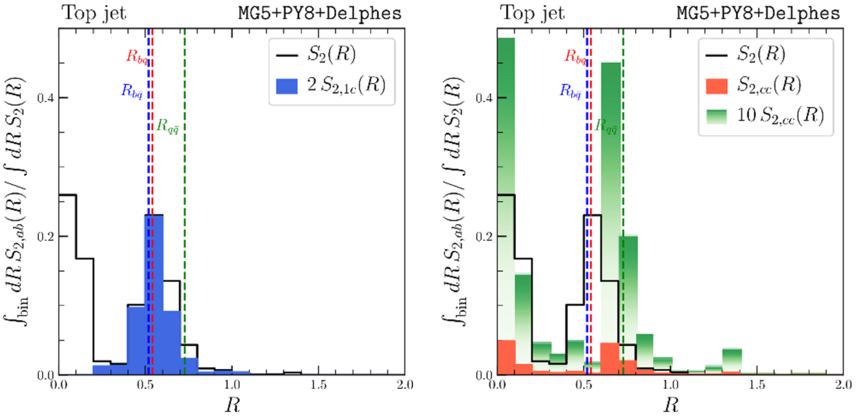


**Figure 5.** Same as figure 4 but the top jet in figure 2(b).

## 3 Morphological analysis of soft emissions

The number of particles of top jets and QCD jets is significantly different. For the boosted top quark decaying hadronically, i.e., $t \to bW \to bq\bar{q}'$, the significant fraction of energy goes to color singlet $W$ boson. The number of particles in a top jet is less than that of a gluon jet with the same jet mass and momentum, and the particles are concentrated near the quark directions. The number of active pixels in the jet image, $N_{\rm pixel}$, is correlated to the number of particles in the jet, and therefore, it should be a crucial quantity of the jet image in the classification. This quantity is IRC unsafe as $E \to 0$ and depends on the physics at a low energy scale, and its accuracy of the theoretical prediction is limited.[3] Indeed PY8 and HW7 predict significantly different pixel distributions for gluon jets, even though they are tuned to the experimental data.

To generalize the idea of $N_{\rm pixel}$, we introduce a morphological analysis of soft emissions on jet images. We consider two morphological operations: dilation, and filtering. Let $N^{(i)}$ be a number of pixels in a dilated image,

$$N^{(i)} = \#(\mathcal{V}^{(i)}), \quad i \in \{0, 1, \cdots\}, \tag{3.1}$$

where $\mathcal{V}^{(i)}$ is the Minkowski sum of the set of the $(\eta, \phi)$ coordinate vectors $\vec{R}_i$ of the active pixels, $\mathcal{V}^{(0)}$, and a set of discrete coordinate vectors on a square for dilation, i.e.,

$$\mathcal{V}^{(i)} = \mathcal{V}^{(0)} + \Delta R \times B^{(i)} = \{a + \Delta R\, b \mid a \in \mathcal{V}^{(0)}, b \in B^{(i)}\}, \tag{3.2}$$

$$B^{(i)} = \{(k, l) | k, l \in \{-i, -i + 1, \cdots, i - 1, i\}\}. \tag{3.3}$$

We denote a set of pixels whose centers belong to $\mathcal{V}^{(i)}$ as $\mathcal{P}^{(i)}$. Note that $\mathcal{P}^{(0)}$ is identical to the set of active pixels, and $N^{(0)}$ is $N_{\rm pixel}$. The set $\mathcal{P}^{(i)}$ is then a cover of the jet image, i.e., it is a union of the squares that attached to each active pixel. The covers obey a recurrence relation that $\mathcal{P}^{(i)}$ includes pixels in $\mathcal{P}^{(i-1)}$ and those touching one of the edges or corners of $\mathcal{P}^{(i-1)}$. This morphological mapping is illustrated in figure 6. Note that $N^{(i)}$

[3] Note that $N_{\rm pixel}$ in this paper is not calculated in the exact limit, $E \to 0$. The electronic calorimeter and hadronic calorimeter simulations have energy thresholds of 0.5 GeV and 1 GeV, respectively.

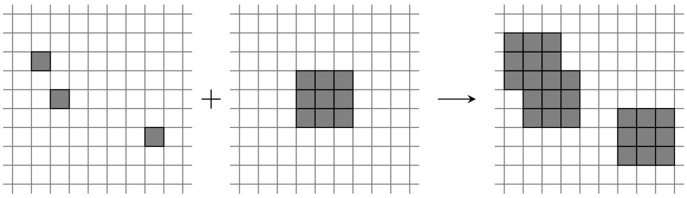


**Figure 6.** An illustration of the Minkowski sum in eq. (3.2). The most left figure shows the active pixels $\mathcal{P}^{(0)}$, the figure at the center shows the pixels whose centers are $B^{(1)}$, and $\mathcal{P}^{(1)}$ is shown in the right figure.

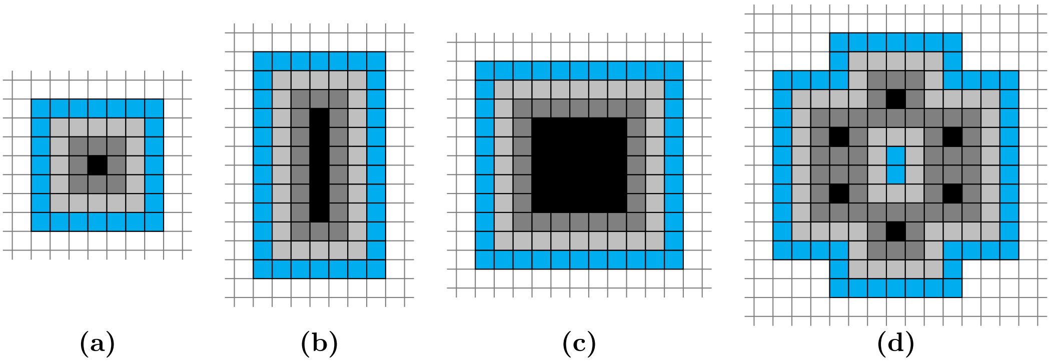


**Figure 7.** Illustrations of $\mathcal{P}^{(i)}$ for (a) an isolated pixel, (b) a line of pixels, (c) a $5 \times 5$ square of pixels, (d) a ring of six pixels. For each plot, black pixels belongs to $P^{(0)}$; dark gray, light gray, blue pixels are the difference $P^{(i)} \setminus P^{(i-1)}$ for $i = 1, 2, 3$, respectively.

is proportional to the area $A^{(i)}$ of pixels in the cover $\mathcal{P}^{(i)}$ because each pixel has the same area $(\Delta R)^2$, i.e.,

$$A^{(i)} = (\Delta R)^2 \times N^{(i)}. \tag{3.4}$$

For the analysis of soft activity, we consider a filtered image whose active pixels have $p_T$ larger than $E$. Let $N^{(i)}(E)$ be the number of active pixels in the filtered image. If we choose sufficiently large threshold $E$, the number $N^{(i)}(E)$ is relatively stable against the choice of the event simulators. The difference between the values of $N^{(i)}$ and $N^{(i)}(E)$ will provide us geometric information about the soft activity.

The sequence of $N^{(i)}$ gives a quantitative description of the spatial distribution of pixels in the jet. Before going into some mathematical background, let us capture the idea using simple examples. Consider the relations between $N^{(0)}$ and $N^{(1)}$ of figure 7(a), figure 7(b), and figure 7(c):

1. Active pixels are separated by two or more pixels.

$$N^{(1)} = 9N^{(0)} \tag{3.5}$$

This corresponds to the limit of sparse and scattered pixels.

2. Active pixels are aligned on a line.

$$N^{(1)} = 3N^{(0)} + 6 \tag{3.6}$$

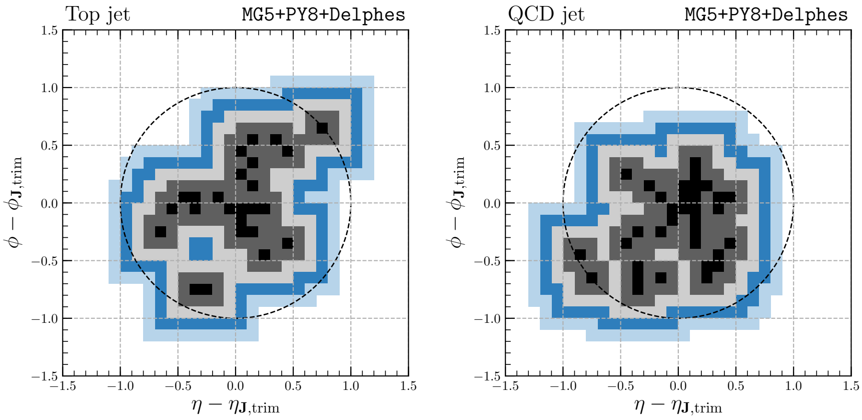


**Figure 8.** The Minkowski sum $\mathcal{P}^{(i)}$ of the top jet image and QCD jet image. For each plot, black pixels belongs to $P^{(0)}$ dark gray, light gray, blue pixels are the difference between $P^{(i)} - P^{(i-1)}$ for $i = 1, 2, 3$, respectively.

This case is the limit when soft activities come from a very narrow color string between two quarks at each end.

3. Active pixels are clustered in a square.

$$N^{(1)} \sim (\sqrt{N^{(0)}} + 2)^2 \tag{3.7}$$

This is the limit of a one-prong jet such as quark jet.

The ratio $N^{(1)}/N^{(0)}$ in large $N^{(0)}$ limit is approximately 9, 3, 1, respectively. If pixel clusters appear at small angular scale, $N^{(1)}/N^{(0)}$ is reduced. Therefore, $N^{(1)}/N^{(0)}$ quantifies the level of isolation of the pixels.

Figure 8 shows $\mathcal{P}^{(i)}$ of top and QCD jet images in figure 2(b) and figure 2(c), respectively. One quick observation is that $\mathcal{P}^{(i)}$ has some non-trivial structures for small $i$ ($i = 0, 1$), but the pixels quickly merge into a single cluster as the index $i$ increases. In the large angular scale ($i \geq 2$), the only relevant physics for the top jet vs. QCD jet classification is the color charge of the parent parton, and the $N^{(i)}$ does not carry significant additional information. In the next section, we show that $N^{(0)}$ and $N^{(1)}$ are sufficient to describe the soft structure contributing to the top jet and QCD jet classifier modeled by CNN.

The analysis based on the pixels can be generalized to the particle level analysis with a continuous parameter $R$ as follows. Let $\mathcal{P}(R)$ be a cover of particles on $(\eta, \phi)$ plane.

$$\mathcal{P}(R) = \bigcup_{i \in \mathbf{J}} B_i(R), \tag{3.8}$$

where $B_i(R)$ is a disk with radius $R$ and whose center is the direction vector $\vec{R}_i$ of a particle $i$. The area $A(R)$ of the cover $\mathcal{P}(R)$ is a quantity related to $N^{(i)}$, i.e., $A^{(i)}$ can be considered as a discrete analog of $A(R)$.

The change of $A(R)$ with respect to $R$ also quantifies the spacial distribution of particles. As far as all the disks are isolated, $A(R)/(\pi R^2)$ is the number of particles. The ratio

$A(R)/R^2$ decreases at the scale where the disks start overlapping. Therefore, the profile of $A(R)/R^2$ along $R$ encodes all the distances between particles.

A more general description of these morphological measures can be obtained from the integral geometry. According to Hadwiger's theorem [149], any geometric measure that has a notion of the size of a polyconvex set in Euclidean space $\mathbb{R}^d$ can be described by a set of functions called "Minkowski functionals". The polyconvex set is a finite union of closed and bounded convex bodies. More precisely, the geometric measure $v$ should satisfy the following properties,

- **Valuation:** $v$ has a notion of size of a set. The value of $v$ of the empty set $\phi$ is zero, i.e, $v(\phi) = 0$, and $v$ satisfies the following inclusion-exclusion property,

  $$v(B_1 \cup B_2) = v(B_1) + v(B_2) - v(B_1 \cap B_2) \tag{3.9}$$

  where $B_1$ and $B_2$ are polyconvex sets.

- **Invariance:** $v$ is invariant under any rotation and translation.

- **Continuity:** for any sequence of polyconvex sets $B_n$ that converges[4] to $B$, its valuation $v(B_n)$ also converges to $v(B)$.

Such geometric measures can be represented as a linear combination of $d + 1$ Minkowski functionals $M_i$,

$$v(B) = \sum_{i=0}^{d} c_i M_i(B). \tag{3.10}$$

In $d = 2$, we have three Minkowski functionals: area, perimeter, and Euler characteristic. Since $\mathcal{P}(R)$ on $(\eta, \phi)$ plane is a finite union of closed and bounded convex bodies $B_i(R)$, its geometry can be described by the Minkowski functionals. We already discussed the area $A(R)$ of $\mathcal{P}(R)$, and its perimeter $L(R)$ and Euler characteristic $\chi(R)$ are also useful quantities. The discrete analog of $L(R)$ and $\chi(R)$ can be used for analyzing jet images.

The Minkowski functionals show that $N^{(1)}$ carries independent information to $N^{(0)}$. We denote the perimeter and Euler characteristic of $\mathcal{P}^{(i)}$ as $L^{(i)}$ and $\chi^{(i)}$. If all the active pixels of a jet image are isolated enough, we may represent $A^{(1)}$ as $A^{(0)} + L^{(0)} \cdot (\Delta R) + 4\chi^{(0)} \cdot (\Delta R)^2$ since $A^{(1)}$ is a valuation of $\mathcal{P}^{(0)}$.[5] The relation between $N^{(0)}$ and $N^{(1)}$ is then as follows,

$$N^{(1)} = N^{(0)} + L^{(0)}/(\Delta R) + 4\chi^{(0)}. \tag{3.11}$$

Note that this relation only holds when the squares attached to active pixels do not overlap each other. Once some squares start to overlap, the relation begins to deviate, and the persistence of this relation can be used as a morphological indicator for the topological change of $\mathcal{P}^{(i)}$. Therefore, $N^{(0)}$ and $N^{(1)}$ are effective variables for analyzing the geometry of soft particles of the jet image.

[4] The convergence is defined in terms of the Hausdorff metric.

[5] For the rectangle shape pixels, the term proportional to $4\chi^{(0)}$ corresponds to the number of pixels that touch only the corner of the pixels in $\mathcal{P}^{(0)}$.

Figure 7(d) is an example that the sequence of $\mathcal{P}^{(i)}$ shows a non-trivial topological change. The sequence starts with six isolated pixels, $\mathcal{P}^{(1)}$ and $\mathcal{P}^{(2)}$ are a ring, and $\mathcal{P}^{(3)}$ is a single large cluster. The Euler characteristic $\chi^{(i)}$ and the perimeter $L^{(i)}$ of $\mathcal{P}^{(i)}$ are as follows.

$$\chi^{(i)} = (6,\ 0,\ 0,\ 1, \cdots) \tag{3.12}$$

$$\frac{L^{(i)}}{\Delta R} = (24,\ 52,\ 52,\ 54, \cdots). \tag{3.13}$$

The non-monotonic behavior of the sequence of Minkowski functionals for analyzing the topology of point distributions is often discussed in other literature [129, 130]. Utilizing this topological information for jet classification problems or global event topology analysis might be interesting, but the full analysis of the sequence of the Minkowski functionals is outside the scope of this paper.

Morphological analysis have been applied in physics to quantify the distribution of the objects. In [129, 130], Minkowski functionals are used to identify the structure of the astrophysical objects. In more recent papers, persistence topology turns out to be useful tool for charcterizing seemingly random distribution of the points and applied in analysis of cosmic microwave background [150] and string landscape [151]. It is tempting to consider other roles of morphological analysis with Minkowski functionals in jet classifications.

## 4 Top tagger based on relation network and jet morphology

In this section, we describe our setup of classifiers trained on the inputs discussed in the previous sections, $S_{2,ab}$ and $N^{(i)}$. These inputs are derivable from jet images, so the CNN performs better than those RNs in principle. We show that the deep learning on the small number of derived inputs reproduces the performance of the CNN. Therefore, those inputs are associated with the relevant physics for solving the classification problem. Moreover, the network using the derived inputs typically has less overfitting than that using the raw inputs.

### 4.1 Training data and model implementation

We simulate top jet and QCD jet samples by `Madgraph5` [152], followed by `Pythia 8` (`PY8`) [153] or `Herwig 7` (`HW7`) [154, 155]. The detector response of generated events is simulated by `Delphes` [156]. Jets are reconstructed by the anti-$k_T$ algorithm with radius parameter $R_{\mathbf{J}} = 1.0$. The jet constituents are calorimeter towers with angular resolution approximately $\Delta R = 0.1$. The details of the sample preparation are explained in appendix A.

We categorize the inputs to the RNs into the four sets: $\boldsymbol{x}_{\text{trim}}$, $\boldsymbol{x}_{\mathbf{J}_1}$, $\boldsymbol{x}_{\text{kin}}$, and $\boldsymbol{x}_{\text{geometry}}$.

- $\boldsymbol{x}_{\text{trim}}$ is a set of discretized $S_{2,\text{trim}}$ and $S_{2,\text{soft}}$ up to angular scale $R = 1.5$,

$$\boldsymbol{x}_{\text{trim}} = (S^{(i)}_{2,\text{trim}} | i = 0, \cdots, 14) \oplus (S^{(i)}_{2,\text{soft}} | i = 0, \cdots, 14), \tag{4.1}$$

where $S^{(i)}_{2,ab}$ is the binned spectrum of $S_{2,ab}$, with bin size $\Delta R$ in order to keep the same angular resolution to the jet image,

$$S^{(i)}_{2,ab} = \frac{1}{\Delta R}\int_{i\Delta R}^{(i+1)\Delta R} dR\, S_{2,ab}(R) = \frac{1}{\Delta R}\sum_{\substack{j\in\mathbf{J}_a,k\in\mathbf{J}_b\\ R_{jk}\in[i\Delta R,(i+1)\Delta R)}} p_{T,j}p_{T,k}. \tag{4.2}$$

We may consider the angular scale up to the diameter $2R_{\mathbf{J}} = 2.0$, but $S^{(i)}_{2,\mathrm{trim}}$ and $S^{(i)}_{2,\mathrm{soft}}$ at such large $R$ are less useful [21].

- $\boldsymbol{x}_{\mathbf{J}_1}$ is a set of discretized $S_{2,11}$, $S_{2,1c}$ and $S_{2,cc}$ as follows,

  $$\boldsymbol{x}_{\mathbf{J}_1} = (S^{(i)}_{2,11}|i=0,\cdots,3)\oplus(S^{(i)}_{2,1c}|i=0,\cdots,9)\oplus(S^{i}_{2,cc}|i=0,\cdots,14). \tag{4.3}$$

  Again, we consider spectra only up to the relevant angular scales. For $S^{(i)}_{2,11}$ and $S^{(i)}_{2,cc}$, the scale is the diameter of the corresponding subjet but too large angular scale is ignored. For $S^{(i)}_{2,1c}$, the scale is the jet radius because it is the correlation between the core part $\mathbf{J}_1$ and its surroundings.

- $\boldsymbol{x}_{\mathrm{kin}}$ is a set of global inputs,

  $$\boldsymbol{x}_{\mathrm{kin}} = (p_{T,\mathbf{J}}, m_{\mathbf{J}}, p_{T,\mathbf{J}_{\mathrm{trim}}}, m_{\mathbf{J}_{\mathrm{trim}}}, p_{T,\mathbf{J}\backslash\mathbf{J}_1}, m_{\mathbf{J}\backslash\mathbf{J}_1}). \tag{4.4}$$

  In addition to the transverse momenta, we include the masses as the inputs because $2m_{\mathbf{J}_a}/p_{T,\mathbf{J}_a}$ is a characteristic angular scale of $\mathbf{J}_a$.

- $\boldsymbol{x}_{\mathrm{geometry}}$ is a set of the numbers of pixels of the jet images $\mathcal{P}^{(0)}$ and $\mathcal{P}^{(1)}$,

  $$\boldsymbol{x}_{\mathrm{geometry}} = (N^{(0)}, N^{(1)}, N^{(0)}(4\,\mathrm{GeV}), N^{(1)}(4\,\mathrm{GeV})). \tag{4.5}$$

We modularize the implementations of the model outputs $\boldsymbol{u}' = \phi^u(\boldsymbol{x})$ to avoid the curse of dimensionality. When inputs are too many, there is a potential danger of overfitting due to sparsely distributed samples. In our previous work, we use $\sim 40$ inputs for the classification of Higgs jets and QCD jets [19, 21]. The inputs for the classification of top jets and QCD jets are increased to $\sim 70$, and training of a simple MLP classifier on these inputs may have difficulties. Therefore, we compress $\boldsymbol{x}_{\mathrm{trim}}$ and $\boldsymbol{x}_{\mathbf{J}_1}$ to a smaller number of hidden variables $\boldsymbol{h}_{\mathrm{trim}}$ and $\boldsymbol{h}_{\mathbf{J}_1}$ by a neural network and get $\boldsymbol{u}'$ from them. The following is the closed-form expression of $\mathrm{RN}_{S_2}$ that uses only the IRC safe inputs: $\boldsymbol{x}_{\mathrm{trim}}$, $\boldsymbol{x}_{\mathbf{J}_1}$, and $\boldsymbol{x}_{\mathrm{kin}}$.

$$\boldsymbol{h}_{\mathrm{trim}} = \mathrm{MLP}_{\mathrm{trim}}(\boldsymbol{x}_{\mathrm{trim}}, \boldsymbol{x}_{\mathrm{kin}}; \boldsymbol{\theta}_{\mathrm{trim}}), \tag{4.6}$$

$$\boldsymbol{h}_{\mathbf{J}_1} = \mathrm{MLP}_{\mathbf{J}_1}(\boldsymbol{x}_{\mathbf{J}_1}, \boldsymbol{x}_{\mathrm{kin}}; \boldsymbol{\theta}_{\mathbf{J}_1}), \tag{4.7}$$

$$\boldsymbol{u}' = \mathrm{MLP}_{\mathrm{logit}}(\boldsymbol{h}_{\mathrm{trim}}, \boldsymbol{h}_{\mathbf{J}_1}, \boldsymbol{x}_{\mathrm{kin}}; \boldsymbol{\theta}_{\mathrm{logit}}), \tag{4.8}$$

where $\mathrm{MLP}_a$ is a multilayer perceptron (MLP) and $\boldsymbol{\theta}_a$ are its trainable parameters. We provide $\boldsymbol{x}_{\mathrm{kin}}$ to each network to tell the characteristic angular scales directly. We use the exponential linear unit (ELU) [157] as the activation function of each MLP. The dimensions

| model | $\boldsymbol{x}_{\rm kin}$ | $\boldsymbol{x}_{\rm trim}$ | $\boldsymbol{x}_{\mathbf{J}_1}$ | $\boldsymbol{x}_{\rm geometry}$ | | | |
|---|---|---|---|---|---|---|---|
| | | | | $N^{(0)}$ | $N^{(0)}(4\,\text{GeV})$ | $N^{(1)}$ | $N^{(1)}(4\,\text{GeV})$ |
| $\text{RN}_{S_2}$ | ○ | ○ | ○ | | | | |
| $\text{RN}_{S_2,N^{(0)}}$ | ○ | ○ | ○ | ○ | | | |
| $\text{RN}_{S_2,N^{(0)},N^{(0)}(4\,\text{GeV})}$ | ○ | ○ | ○ | ○ | ○ | | |
| $\text{RN}_{S_2,N^{(0)},N^{(1)}}$ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |

**Table 1.** The list of inputs used in each RN. The circle represents that the given input is used. The CNN trained on jet images can utilize all this information.

of $\boldsymbol{h}_{\rm trim}$ and $\boldsymbol{h}_{\mathbf{J}_1}$ are 5. The output $\boldsymbol{u}'$ is a dimension two vector and will be transformed into the softmax outputs for the binary classification purpose.

$$\hat{y}^i = \frac{\exp(u'_i)}{\exp(u'_0) + \exp(u'_1)}, \quad i = 0, 1 \tag{4.9}$$

When the geometric information $\boldsymbol{x}_{\rm geometry}$ is included in the inputs, we use them as arguments of $\text{MLP}_{\rm logit}$,

$$\boldsymbol{u}' = \text{MLP}_{\rm logit}(\boldsymbol{h}_{\rm trim}, \boldsymbol{h}_{\mathbf{J}_1}, \boldsymbol{x}_{\rm kin}, \boldsymbol{x}_{\rm geometry}; \boldsymbol{\theta}_{\rm logit}) \tag{4.10}$$

We consider three additional relation networks that uses the geometric information: $\text{RN}_{S_2,N^{(0)}}$, $\text{RN}_{S_2,N^{(0)},N^{(0)}(4\,\text{GeV})}$, and $\text{RN}_{S_2,N^{(0)},N^{(1)}}$. Their inputs are listed in table 1. The detailed implementations of these RNs are in appendix C.1.

The softmax output is trained by minimizing the cross-entropy loss function. In addition, we marginalize the $p_{T,\mathbf{J}}$ distribution in the classification because the top jet samples and QCD jet samples have different $p_{T,\mathbf{J}}$ distributions. To do this, we train networks in a way that interpolates binary classifiers for the jets at given $p_{T,\mathbf{J}}$. The corresponding cross-entropy loss function $\mathcal{L}_{\rm CE}$ is as follows.

$$\text{CE}(p_{T,\mathbf{J}}; \boldsymbol{\theta}) = -\frac{1}{2} \sum_{Y=\text{top,QCD}} \int d\tilde{\boldsymbol{x}} f_{\tilde{\boldsymbol{x}}|p_{T,\mathbf{J}}}(\tilde{\boldsymbol{x}}; Y) \sum_{i=0,1} y^i_Y \log \hat{y}^i(\boldsymbol{x}; \boldsymbol{\theta}) \tag{4.11}$$

$$\mathcal{L}_{\rm CE}(\boldsymbol{\theta}) = \frac{1}{p^{\max}_{T,\mathbf{J}} - p^{\min}_{T,\mathbf{J}}} \int^{p^{\max}_{T,\mathbf{J}}}_{p^{\min}_{T,\mathbf{J}}} dp_{T,\mathbf{J}}\, \text{CE}(p_{T,\mathbf{J}}; \boldsymbol{\theta}) \tag{4.12}$$

where $Y$ is a category label, $y_{\rm top} = (1, 0)$, and $y_{\rm QCD} = (0, 1)$. The function $f_{\tilde{\boldsymbol{x}}|p_{T,\mathbf{J}}}(\tilde{\boldsymbol{x}}; Y)$ is the conditional probability density of $\tilde{\boldsymbol{x}}$ given $p_{T,\mathbf{J}}$, and $\tilde{\boldsymbol{x}}$ is $\boldsymbol{x}$ without $p_{T,\mathbf{J}}$.

The integral can be approximated by a Monte-Carlo integration,

$$\mathcal{L}_{\rm CE}(\boldsymbol{\theta}) \approx -\frac{1}{2} \sum_{Y=\text{top,QCD}} \sum_{i_Y=1}^{N_Y} \frac{1}{f_{p_{T,\mathbf{J}}}(p^{[i_Y]}_{T,\mathbf{J}}; Y)} \sum_{i=0,1} y^i_Y \log \hat{y}^i(\boldsymbol{x}^{[i_Y]}; \boldsymbol{\theta}) \tag{4.13}$$

where $f_{p_{T,\mathbf{J}}}(p_{T,\mathbf{J}}; Y)$ is the probability density function of $p_{T,\mathbf{J}}$ given $Y$, and the variables with superscript $[i_Y]$ is the value at the $i_Y$-th sample in the training dataset of $Y$. The probability density function $f_{p_{T,\mathbf{J}}}(p_{T,\mathbf{J}};Y)$ is modeled by kernel density estimation (KDE) described in appendix B. The resulting loss function is essentially a cross-entropy with samples whose $p_{T,\mathbf{J}}$ distribution is reweighted to be uniform. In addition to this cross-entropy loss, $L_2$ regularizer $\mathcal{L}_{\text{reg}}$ [158–160] with the weight decay constant $\lambda = 0.001$ is added to regularize $\text{MLP}_a$.

$$\mathcal{L}_{\text{reg}} = \frac{\lambda}{2}\sum_a |\boldsymbol{W}_a|^2 \tag{4.14}$$

where $\boldsymbol{W}_a$ are the weights of hidden layers in $\text{MLP}_a$.

The training setup is as follows. We minimize the loss function $\mathcal{L}(\boldsymbol{\theta}) = \mathcal{L}_{\text{CE}}(\boldsymbol{\theta})+\mathcal{L}_{\text{reg}}(\boldsymbol{\theta})$ by ADAM optimizer [161] with learning rate 0.001, the first moment exponential moving average coefficient $\beta_1 = 0.9$, the second moment exponential moving average coefficient $\beta_2 = 0.999$, and stabilization constant $\epsilon = 10^{-7}$. Batched samples are used in order to reduce overfitting. The weights of the MLP are initialized by the He initializer [162], and the biases are initialized to be zero. We will use early stopping for the termination criterion, but there is a chance that the network is mildly overfitted to the validation dataset during learning the features of the rare events. If the gradients from the rare events distort the trained results for the dominant events, the network parameters have to be corrected again, and the training becomes noisy. The random overfitting to the validation sample occurs during this noisy learning on the rare events. To avoid this artifact, we use the exponential moving averages $\hat{\boldsymbol{\theta}}^{(t)}$ of the trainable parameters $\boldsymbol{\theta}^{(t)}$ at the epoch $t$ for the validation and testing. The details of the moving average can be found in appendix D. We monitor the loss $\mathcal{L}_{\text{tot}}(\hat{\boldsymbol{\theta}})$ of the validation samples during the training and terminate the training if the loss function does not improve during 50 latter epochs. The networks and training setup is implemented in Keras [163] with TensorFlow [164] backend. Optimization on the batch number is performed by the grid search. We iterate the training for batch numbers, 20, 50, and 100 and two different random number seeds.

The results of the RN-based classifier will be compared to that of a CNN-based classifier. The CNN model is similar to that of the previous paper [21] but with more nodes and layers. The closed-form expression of the CNN is as follows,

$$\boldsymbol{h}_{\text{image}} = \text{CNN}_{\text{image}}(\boldsymbol{x}_{\text{image}};\boldsymbol{\theta}_{\text{CNN}}) \tag{4.15}$$

$$\boldsymbol{u}' = \text{MLP}_{\text{logit}}(\boldsymbol{h}_{\text{image}},\boldsymbol{x}_{\text{kin}};\boldsymbol{\theta}_{\text{MLP}}), \tag{4.16}$$

where $\boldsymbol{x}_{\text{image}}$ is energy deposits of the preprocessed jet image described in appendix C.2. The module $\text{CNN}_{\text{image}}$ consists of 6 two-dimensional convolutional layers with $3 \times 3$ filters and ELU activations. We insert two $2 \times 2$ max-pooling layers after the third and sixth convolutional layers. The $\boldsymbol{h}_{\text{image}}$ are the flattened outputs of the $\text{CNN}_{\text{image}}$. The model outputs $\boldsymbol{u}'$ are from an MLP analyzing $\boldsymbol{h}_{\text{image}}$ together with the kinematic information $\boldsymbol{x}_{\text{kin}}$. The detailed implementation of this CNN is in appendix C.2. The training setup is the same as that of the RNs, but we check batch numbers 100, 200, and 500 instead because of the limitation of GPU memory.

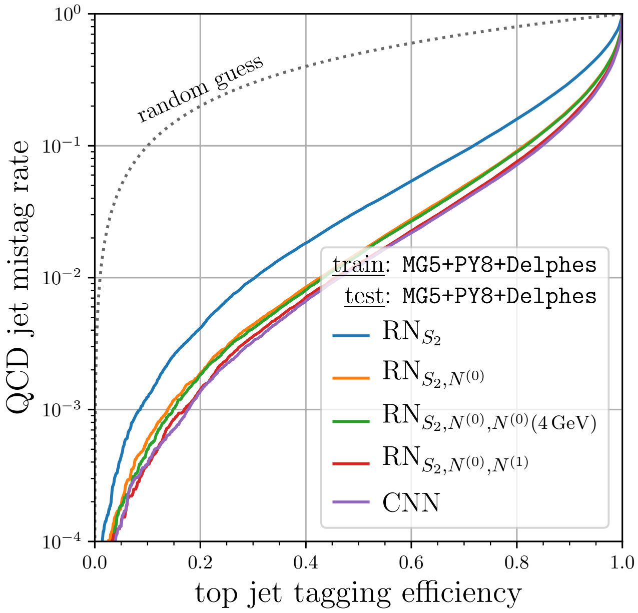


**Figure 9.** The ROC curves of the networks trained on PY8 samples.

### 4.2 Classification results

Figure 9 shows the ROC curves of the networks trained on PY8 samples. The AUC, which is the upper area of each curve, of $\mathrm{RN}_{S_2}$, $\mathrm{RN}_{S_2,N^{(0)}}$, $\mathrm{RN}_{S_2,N^{(0)},N^{(1)}}$, and CNN are 0.8990, 0.9352, 0.9442, and 0.9465, respectively. There is a large gap between the ROC curves of $\mathrm{RN}_{S_2}$ and CNN. This gap is partially filled by including an additional input $N^{(0)}$, as shown in the ROC curve of $\mathrm{RN}_{S_2,N^{(0)}}$. Surprisingly, when we consider all the geometric inputs $\boldsymbol{x}_{\mathrm{geometry}}$, the ROC curve of $\mathrm{RN}_{S_2,N^{(0)},N^{(1)}}$ is almost equal to that of the CNN. Therefore, the inputs $\boldsymbol{x}_{\mathrm{trim}}$, $\boldsymbol{x}_{\mathbf{J}_1}$, $\boldsymbol{x}_{\mathrm{kin}}$, and $\boldsymbol{x}_{\mathrm{geometry}}$ can be considered as useful middle-level variables for modeling the top jet classifier.

The reason for a big gap between the ROC curves of $\mathrm{RN}_{S_2}$ and CNN is the difference in $N^{(0)}$ distributions between top jet samples and QCD jet samples. The QCD jets in this paper are leading $p_T$ jets of $pp \to jj$ so that they are mostly gluon jets, which have a large $N^{(0)}$ than a jet from a color triplet parton. In addition, PY8 predicts significantly higher $N^{(0)}$ of gluon jets than HW7, as in figure 10. Similar situations have been pointed out for the counting variables such as the charged track multiplicity [99], and the soft drop multiplicity [105].

The situation may be compared with the classification of Higgs jets and QCD jets, studied in [19]. In this case, the difference between the ROC curves of $\mathrm{RN}_{S_2}$ and CNN is tiny. QCD jet samples in the study are leading $p_T$ jets of $pp \to Zj$ with invisibly decaying $Z$ boson, and most of the samples are the quark jets. The difference in the $N^{(0)}$ distribution of the Higgs jets and QCD jets is small, and therefore, $N^{(0)}$ does not play an important role there.

The remaining gap between the ROC curves of $\mathrm{RN}_{S_2,N^{(0)}}$ and CNN is almost filled by including $N^{(1)}$ in the analysis. As discussed in the previous section, the ratio $N^{(1)}/N^{(0)}$

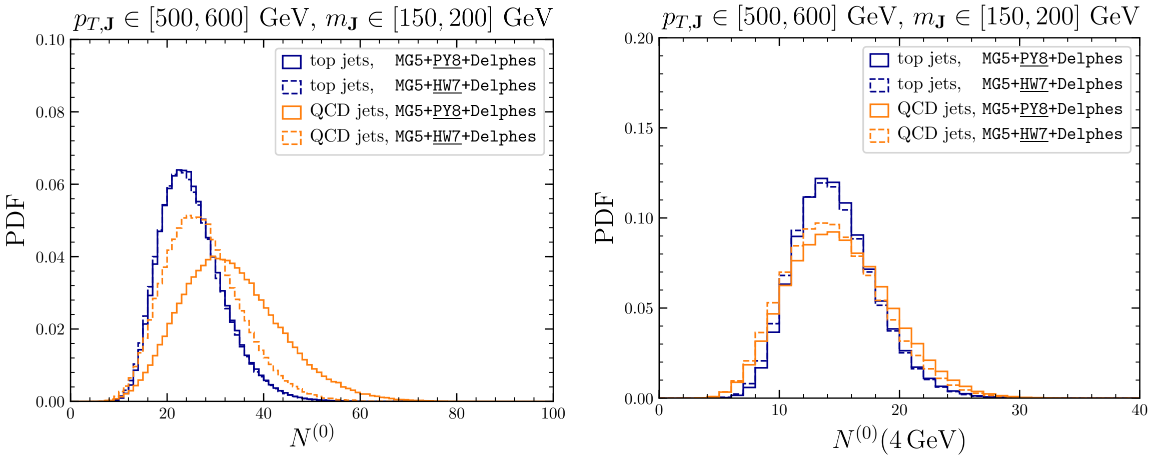


**Figure 10.** $N^{(0)}$ and $N^{(0)}(4\,\mathrm{GeV})$ distributions of PY8 and HW7 data sets. The blue histograms are for the top jets, and the orange histograms are for QCD jets. The solid lines are for PY8 generated samples, and the dashed lines are for HW7 generated samples.

is a morphological measure that quantifies the level of clustering of the pixels. Therefore, $N^{(1)}$ is useful for distinguishing compact top jets from QCD jets whose number of pixels is the same. The similarity of two ROC curves indicates that the information summarized in the Minkowski functionals is used in the jet image analysis.

Not only $\mathrm{RN}_{S_2,N^{(0)},N^{(1)}}$ gives a comparable result to CNN, but it is also significantly more stable. We compare the softmax output $\hat{y}^0$ of a network $\mathcal{N}$ and the output of the same network trained with a different random seed, and we call the alternative output $\hat{y}'^0$. The change of the seed affects the shuffling of the events between batches and alters the initialization of the network. Since the training of the neural network is not a convex optimization in general, the network output difference $\Delta\hat{y}^0[\mathcal{N}] = \hat{y}'^0[\mathcal{N}] - \hat{y}^0[\mathcal{N}] \neq 0$. In figure 11, we show the histogram of two outputs $(\hat{y}^0, \hat{y}'^0)$ for $\mathrm{RN}_{S_2,N^{(0)},N^{(1)}}$ and CNN. The distribution for $\mathrm{RN}_{S_2,N^{(0)},N^{(1)}}$ is narrower than that for CNN. This shows that training of $\mathrm{RN}_{S_2,N^{(0)},N^{(1)}}$ is more stable.

The better training stability of RN is due to the difference in the inputs of the functional model. The pair of preprocessed jet image and $\boldsymbol{x}_{\mathrm{kin}}$ contain more information than the two-point energy correlations and Minkowski functionals. Hence, CNN could approximate a wider variety of functions of jet constituents than RN. In other words, the training of CNN requires more effort in order to scan over larger space of functions. The training of a simpler model is much stable than that of a complex model because of less number of inputs and trainable parameters. A simpler model has a potential danger of underfitting, but it is less severe in RN because $S_{2,ab}$ and $N^{(i)}$ are reasonable set for describing functional space of energy correlation and geometry of the jet constituents, respectively.

We now compare the outputs $\hat{y}^0[\mathrm{CNN}]$ and $\hat{y}^0[\mathrm{RN}] = \hat{y}^0[\mathrm{RN}_{S_2,N^{(0)},N^{(1)}}]$.[6] Figure 12 shows the distributions of $\Delta\hat{y}^0[\mathrm{RN}]$, $\Delta\hat{y}^0[\mathrm{CNN}]$, and $\Delta\hat{y}^0[\mathrm{CNN},\mathrm{RN}] = \hat{y}^0[\mathrm{CNN}] - \hat{y}^0[\mathrm{RN}]$. The mean and standard deviation of these differences are summarized in table 2. All the

[6] From here, we denote $\mathrm{RN}_{S_2,N^{(0)},N^{(1)}}$ as RN.

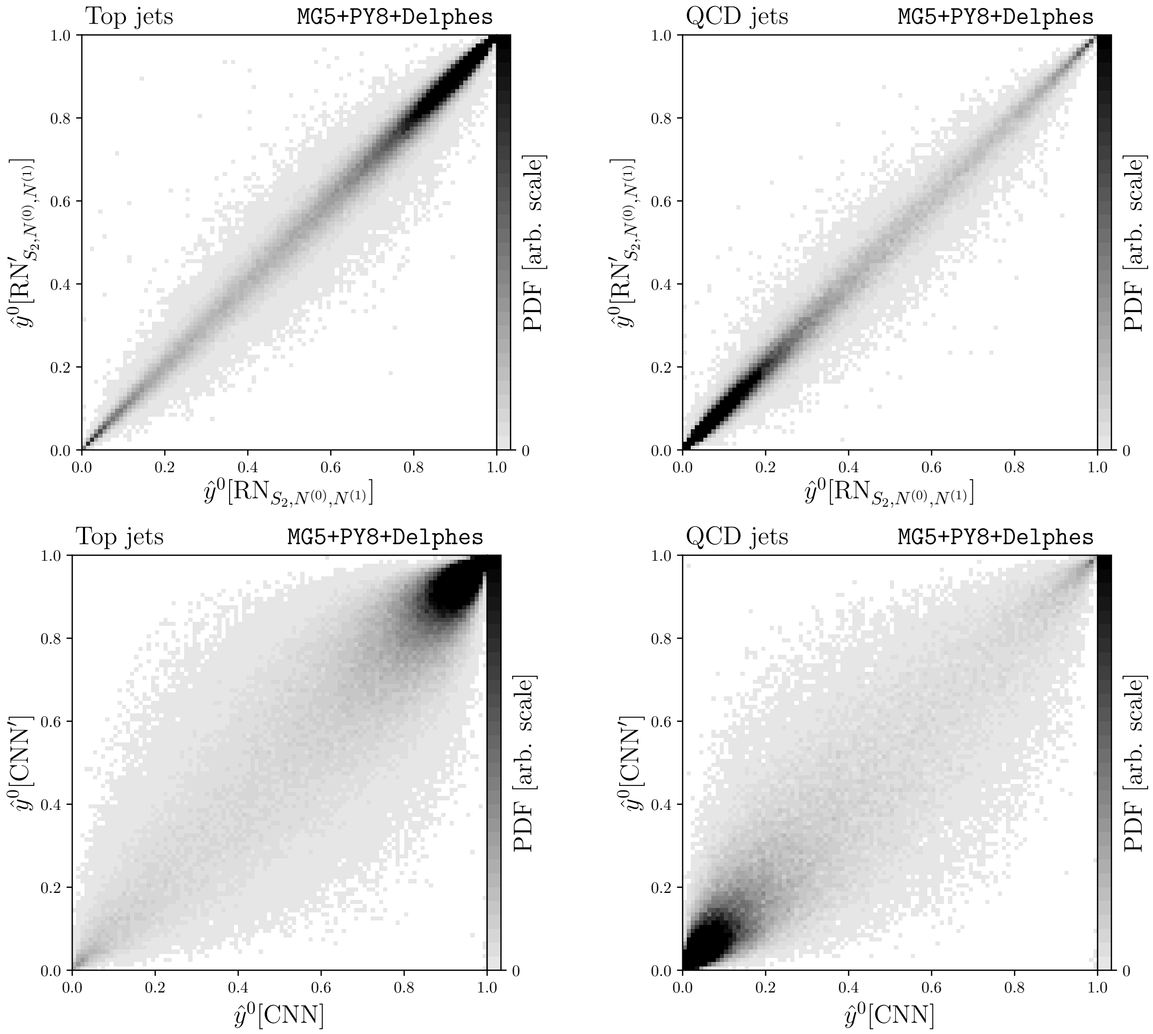


**Figure 11.** The distribution of softmax output of the classifiers for different random seed. upper two figures for $\hat{y}^0[\mathrm{RN}_{S_2,N^{(0)},N^{(1)}}]$, and the bottom two figures for $\hat{y}^0[\mathrm{RN}_{S_2,N^{(0)},N^{(1)}}]$. The left figures are top jets and the right figures are QCD jets.

$\Delta\hat{y}^0$ distributions are sharply peaked approximately at $\Delta\hat{y}^0 = 0$, which indicates that the classifiers make the same decision for the majority of the events. Since the training of the CNN is less stable than that of RN, the standard deviation $\sigma(\Delta\hat{y}^0[\mathrm{CNN}])$ of $\Delta\hat{y}^0[\mathrm{CNN}]$ is much larger than $\sigma(\Delta\hat{y}^0[\mathrm{RN}])$ of $\Delta\hat{y}^0[\mathrm{RN}]$. The standard deviation of $\Delta\hat{y}^0[\mathrm{CNN},\mathrm{RN}]$ is larger than the error $\sqrt{\sigma(\Delta\hat{y}^0(\mathrm{CNN}))^2 + \sigma(\Delta\hat{y}(\mathrm{RN}))^2}$, which is 0.091 for the top jet samples and 0.095 for the QCD jet samples. This indicates that the outputs of RN and CNN are highly correlated, but there are still some differences. We repeat the same analysis on non-typical events, which satisfies $0.15 < \hat{y}^0 < 0.85$ for one of RN or CNN. The results are similar, but the standard deviations are larger by a factor 1.5 because we removed samples easy to classify.

In order to understand the cases on which $\mathrm{RN}_{S_2,N^{(0)},N^{(1)}}$ and CNN gives us extremely different answers, we show two examples in figure 13. To choose jets with stable CNN

| output difference | | top jet samples | | QCD jet samples | |
|---|---|---|---|---|---|
| | | average | deviation | average | deviation |
| $\Delta\hat{y}^0[\mathrm{RN}]$ | $= \hat{y}'^0[\mathrm{RN}] - \hat{y}^0[\mathrm{RN}]$ | $-9.56 \times 10^{-4}$ | 0.0271 | $-1.65 \times 10^{-4}$ | 0.0279 |
| $\Delta\hat{y}^0[\mathrm{CNN}]$ | $= \hat{y}'^0[\mathrm{CNN}] - \hat{y}^0[\mathrm{CNN}]$ | $-1.46 \times 10^{-3}$ | 0.0867 | $-6.14 \times 10^{-3}$ | 0.0911 |
| $\Delta\hat{y}^0[\mathrm{CNN},\mathrm{RN}]$ | $= \hat{y}^0[\mathrm{CNN}] - \hat{y}^0[\mathrm{RN}]$ | $6.98 \times 10^{-3}$ | 0.141 | $3.10 \times 10^{-3}$ | 0.144 |
| | $\hat{y}'^0[\mathrm{CNN}] - \hat{y}^0[\mathrm{RN}]$ | $5.51 \times 10^{-3}$ | 0.137 | $9.26 \times 10^{-3}$ | 0.142 |
| after selection: $0.15 < \hat{y}^0[\mathrm{RN}_{S_2,N^{(0)},N^{(1)}}] < 0.85$ or $0.15 < \hat{y}^0[\mathrm{CNN}] < 0.85$ | | | | | |
| $\Delta\hat{y}^0[\mathrm{RN}]$ | $= \hat{y}'^0[\mathrm{RN}] - \hat{y}^0[\mathrm{RN}]$ | $-1.60 \times 10^{-3}$ | 0.0403 | $-1.06 \times 10^{-3}$ | 0.0409 |
| $\Delta\hat{y}^0[\mathrm{CNN}]$ | $= \hat{y}'^0[\mathrm{CNN}] - \hat{y}^0[\mathrm{CNN}]$ | $3.64 \times 10^{-3}$ | 0.129 | $3.51 \times 10^{-3}$ | 0.131 |
| $\Delta\hat{y}^0[\mathrm{CNN},\mathrm{RN}]$ | $= \hat{y}^0[\mathrm{CNN}] - \hat{y}^0[\mathrm{RN}]$ | $1.61 \times 10^{-2}$ | 0.215 | $3.50 \times 10^{-3}$ | 0.217 |
| | $\hat{y}'^0[\mathrm{CNN}] - \hat{y}^0[\mathrm{RN}]$ | $1.97 \times 10^{-2}$ | 0.206 | $9.39 \times 10^{-3}$ | 0.210 |

**Table 2.** Average and standard deviation of the output difference $\Delta y^0$.

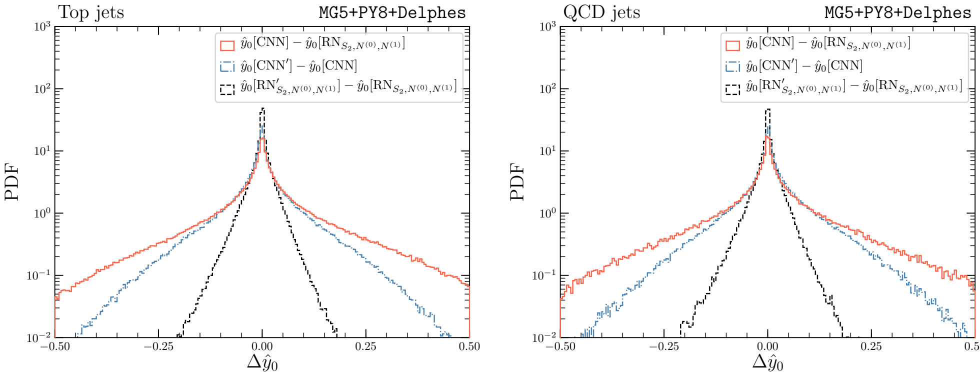


**Figure 12.** The difference of softmax output for various models, $\Delta\hat{y}^0[\mathrm{RN}]$, $\Delta\hat{y}^0[\mathrm{CNN}]$, and $\Delta\hat{y}^0[\mathrm{CNN},\mathrm{RN}]$ for top jets and QCD jets.

predictions, the selected jets have similar $\hat{y}^0[\mathrm{CNN}]$ and $\hat{y}'^0[\mathrm{CNN}]$. For the left jet image, CNN judges the jet as a top jet while RN does not. The $b$ quark and one of the light quarks accidentally overlap in this event. This type of event is certainly not typical. The probability that the angle $R_{bq}$ or $R_{b\bar{q}}$ is less than 0.2 is 5.6% without considering spin-correlation. For the right jet image, CNN judges that the jet is a QCD jet, but RN does not. It is a two-prong jet with many soft radiations and a small $p_T$ subjet from a quark due to longitudinal decay of W boson. In the longitudinal decay, one of the quark goes backward to the boost direction, but these jets suppressed in the phase space. Because both of the top jets are not typical, it is not surprising the two different models give very different results for those events.

We have checked if more aggressive training on these rare events improves the performance, for example, relaxing the regularizer setup. The AUCs of RN and CNN with the weight decay constant $\lambda = 10^{-4}$ are 0.9461 and 0.9465, respectively. There are tiny improvements in the classification performance, but it comes together with overfitting. The

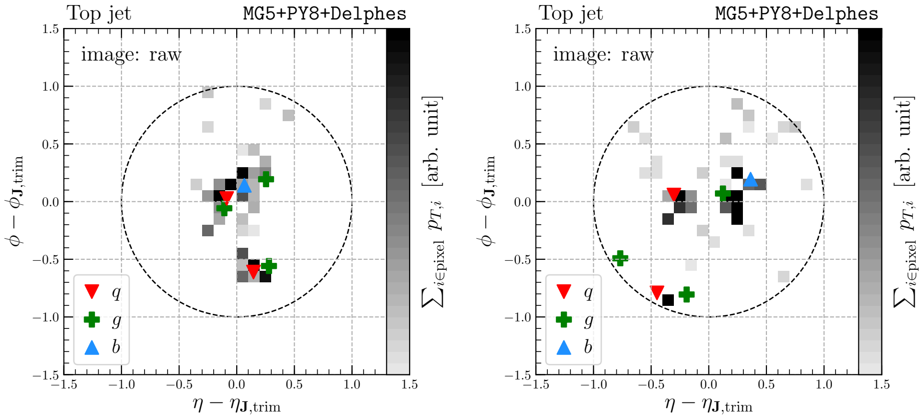


**Figure 13.** Jets images of top jets that $\mathrm{RN}_{S_2,N^{(0)},N^{(1)}}$ and CNN give different answers. The network outputs of each jet image are as follows: (left) $\hat{y}^0[\mathrm{RN}_{S_2,N^{(0)},N^{(1)}}] = 0.0795$, $\hat{y}^0[\mathrm{CNN}] = 0.908$, $\hat{y}'^0[\mathrm{CNN}] = 0.883$, (right) $\hat{y}^0[\mathrm{RN}_{S_2,N^{(0)},N^{(1)}}] = 0.836$, $\hat{y}^0[\mathrm{CNN}] = 0.0905$, $\hat{y}'^0[\mathrm{CNN}] = 0.101$,

validation loss $\mathcal{L}(\hat{\boldsymbol{\theta}})$ and training loss $\mathcal{L}(\boldsymbol{\theta})$ are 0.3044, and 0.2969 for RN; 0.3201, and 0.2971 for CNN, respectively. The $\mathcal{L}(\hat{\boldsymbol{\theta}})$ and $\mathcal{L}(\boldsymbol{\theta})$ in the original setup are 0.3076, and 0.3049 for $\mathrm{RN}_{S_2,N^{(0)},N^{(1)}}$; 0.3338, and 0.3371 for the CNN. The difference between the training and validation loss is much bigger in $\lambda = 10^{-4}$ setup, which is a sign of overfitting.

### 4.3 Alternative vertex label choice

Vertex label is a hyperparameter of the RNs, and we use labels based on the trimmed jet and leading $p_T$ subjet in order to explicitly identify hard substructures and subleading $p_T$ substructures. Other labels may be used depending on the purpose. For example, trimming may be replaced with recursive soft drop (RSD) [89, 165] for better analytic tractability of the two-point energy correlations. Let $\mathbf{J}_r$ be the groomed jet by the RSD,[7] and $r$ be its vertex label. We define $S_{2,\mathrm{RSD}}$ and $S_{2,\mathrm{RSD}^c}$,

$$S_{2,\mathrm{RSD}}(R) = S_{2,rr}(R), \tag{4.17}$$

$$S_{2,\mathrm{RSD}^c}(R) = S_2(R) - S_{2,rr}(R), \tag{4.18}$$

corresponding to $S_{2,\mathrm{trim}}$ and $S_{2,\mathrm{soft}}$, respectively. Figure 14 shows the $S_{2,\mathrm{RSD}}$ distribution of the top jets and QCD jet in figure 2, but the difference is small. The two-point energy correlation related to the soft activity that satisfies the soft drop condition may be included in $S_{2,\mathrm{RSD}}$.

Figure 15 shows the ROC curves of $\mathrm{RN}_{S_2}$ and $\mathrm{RN}_{S_2,N^{(0)},N^{(1)}}$ after replacing inputs $S_{2,\mathrm{trim}}$, $S_{2,\mathrm{soft}}$, $p_{T,\mathbf{J}_h}$, and $m_{\mathbf{J}_h}$ to $S_{2,\mathrm{RSD}}$, $S_{2,\mathrm{RSD}^c}$, $p_{T,\mathbf{J}_r}$, and $m_{\mathbf{J}_r}$, respectively. The performance does not change much because the change of inputs is simply a rearrangement of $S_2$ bins related to the soft activity that satisfies the soft drop condition. Therefore, the impact on the top jet classification performance due to the change of groomer is small.

[7] We use soft drop parameters $z_{\mathrm{cut}} = 0.5$ and $\beta = 1$, and fully inspect whole clustering history.

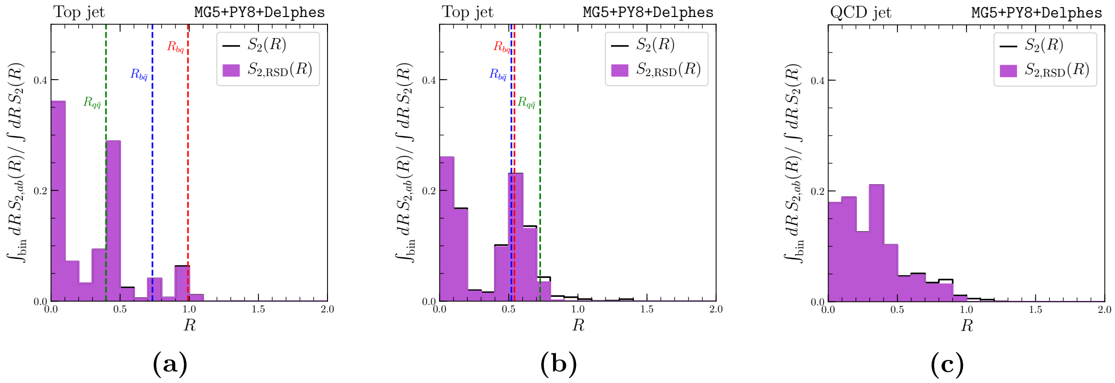


**Figure 14.** The $S_2$ and $S_{2,\mathrm{RSD}}$ distributions of the top jets and the QCD jet in figure 2. The dashed lines are the characteristic angular scales of the top jets in the parton level.

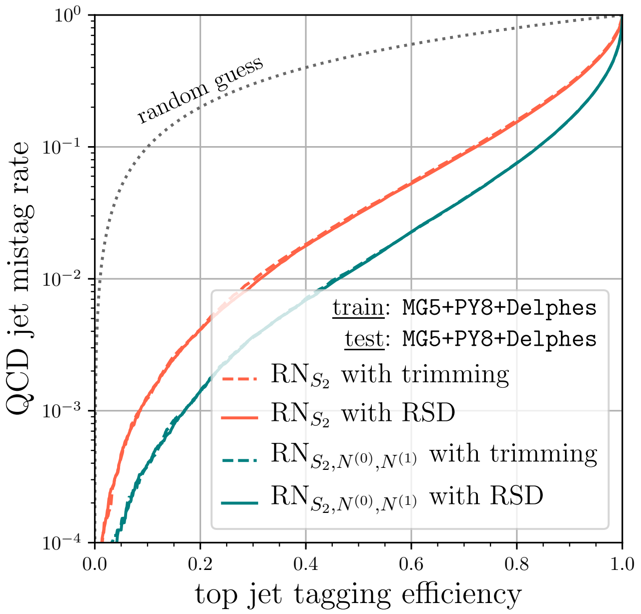


**Figure 15.** The ROC curves of $\mathrm{RN}_{S_2}$ and $\mathrm{RN}_{S_2,N^{(0)},N^{(1)}}$ with trimming or RSD.

### 4.4 Discussion on other top taggers

So far, we compare the performance of the RNs to that of the CNN. In this subsection, we comment on other top taggers.

In [79], ParticleNet [20, 166] and ResNeXt [167] show a better performance in the top jet classification than the CNN.[8] One may wonder if additional features should be included in the RN inputs to reproduce their performance. However, the networks on figure 5 of [79] are not trained on inputs at the same angular resolution. It is not clear if the better networks learn additional physical features. The ResNeXt and CNN in [79] use jet images with pixel size 0.025 and 0.04, respectively. We especially find that the performances of ResNeXt and CNN trained on jet images with pixel size 0.1 are similar. Figure 16 shows their ROC curves.

[8] Note that this CNN does not take $x_{\mathrm{kin}}$ as inputs and is different from the CNN in this paper.

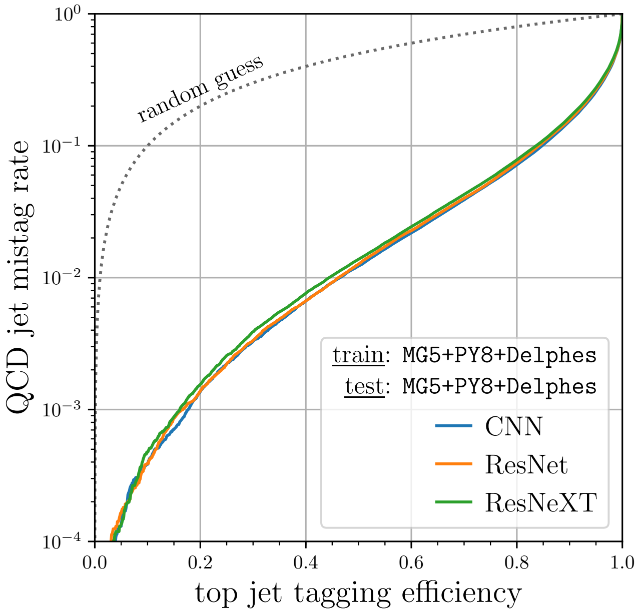


**Figure 16.** The ROC curves of the CNN and its variants: ResNet and ResNeXt.

The ResNet [168] and ResNeXt in figure 16 are the CNN after replacing the chain of the convolutional layers to ResNet [168] or ResNeXt modules described in appendix C.2. Note that the skip connections in those residual learning networks are for solving the degradation problem [168] without deteriorating the universal approximation property of the filter direction of convolutional layers. If there is no performance degradation due to the depth of the networks, all of those networks should perform similarly. If we change the pixel size from 0.1 to 0.025, the jet image size changes from 30 × 30 to 120 × 120 and we may need a CNN with more layers or larger filter sizes in order to cover the whole $(\eta, \phi)$ range. The skip connections may be required to train the network efficiently. Figure 16 shows that the CNN is sufficient for the classification in our case.

The ResNeXt in [79] shows a similar performance to the ParticleNet. The ParticleNet is a graph neural network that uses angular coordinates directly, and the angular resolution is not explicitly considered in the inputs. However, since `Delphes` provides each constituent's angular position after uniform smearing over corresponding calorimeter bin range [156], the inputs of the ParticleNet has implicit angular resolution 0.0174 and 0.1 if the constituent is from electromagnetic and hadronic calorimeter, respectively. The jet images for the ResNeXt uses pixel width 0.025, so that the loss of information due to pixelation is small. We leave further investigation between our RNs to those networks in future publications.

## 5 Reweighting distributions of IRC unsafe morphological features

In section 4, we perform the analysis using sample generated by `PY8`, but in this section, we compare the result with the analysis using another event generator and discuss the systematic uncertainties associated with simulations. Because event generation involves the modeling of soft radiation, the generated events are model-dependent, and the simulator has to be tuned to experimental data. Describing the distribution of particles in the jet in all circumstances is not trivial. Indeed, the simulated distributions of different generators are often significantly different in an extreme kinematic regime, and sometimes neither of them agrees with experimental data. The question is how precisely these simulated events should agree with the data. For the analysis mainly using high $p_T$ objects, the effects of soft physics are small. On the other hand, a neural network based jet classifier trained on jet images are capable of utilizing the pattern of soft radiation. If the agreement between observed and simulated data are "sufficiently good", we could rely on the simulated data. In reality, there are yet significant deviations between the MC predictions and experimental data, and the calibrations are necessary. Because we know that the less controlled IRC unsafe quantities, such as $N^{(0)}$ and $N^{(1)}$, play an important role in the classification, we focus on calibrating the difference between the experimental and simulated data of those quantities.

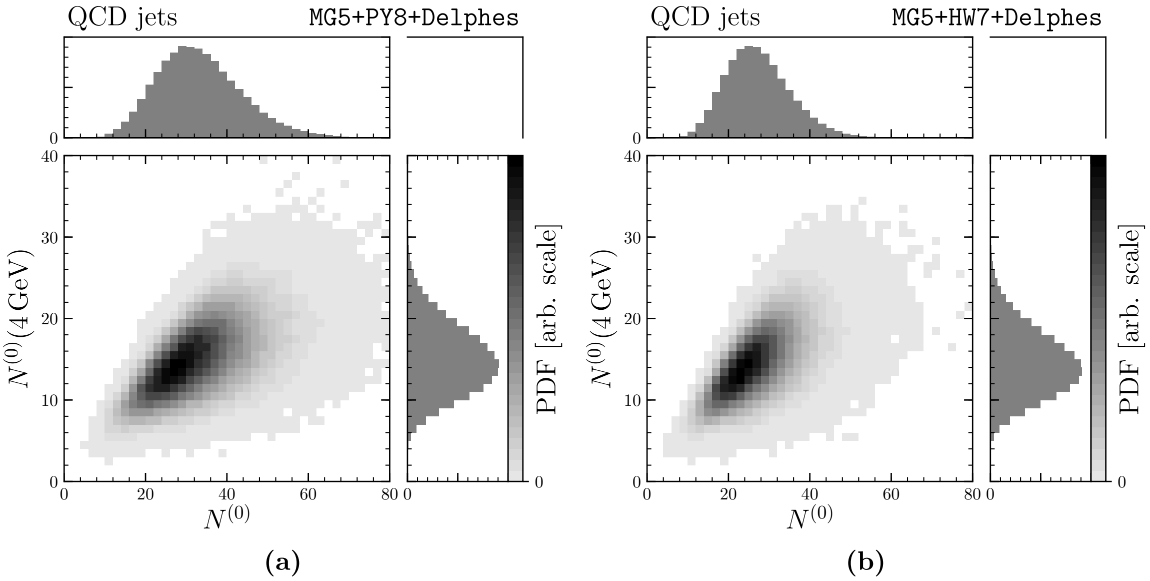
To see the systematical error coming from the mismodeling of the parton shower and hadronization, we perform the same classification analysis with different event generators and compare the results. We choose `HW7` and `PY8` for the comparison. The two event generators are quite different in modeling of the soft and collinear radiations. `HW7` uses the angular-ordered shower [169] and the cluster hadronization model [155, 170].[9] `PY8` uses $p_T$ ordered-shower [171] and the string model of hadronization [172, 173]. The comparison of the radiation pattern of QCD jets is available in various literature [100, 174, 175]. The prediction of the gluon jet distributions differs significantly in each simulator while it more or less agrees with each other for the quark jets. It is pointed out that prediction is sensitive to the color reconnection modeling.

In figure 17, we show the $(N^{(0)}, N^{(0)}(4\,\mathrm{GeV}))$ distributions of the QCD jet samples. The $N^{(0)}$ distribution simulated by `PY8` is broader than that simulated by `HW7`. The tail of the $N^{(0)}$ distribution exceeds 60 for `PY8`, but it vanishes at there for `HW7`. On the other hand, the $N^{(0)}(4\,\mathrm{GeV})$ distributions of `PY8` and `HW7` are similar, as shown in figure 10. The active pixels with $p_T > 4\,\mathrm{GeV}$ correspond to the particles from high $p_T$ partons in the shower. Predictions on those partons in the two generators tend to agree, and the predicted $N^{(0)}(4\,\mathrm{GeV})$ distributions are also similar. The $N^{(1)}/N^{(0)}$ distributions of `PY8` and `HW7` are also similar, as shown in figure 18. Therefore, $N^{(0)}$ should play an important role in the classification.

The separation of the top jets and QCD jets is worse for `HW7` compared with `PY8` discussed in previous sections. The AUC of the top jet vs. QCD jet classification predicted by `HW7` is smaller than that predicted by `PY8`. In figure 19, we show the ROC curves of each classifier trained on `HW7` events. The performance of the $\mathrm{RN}_{S_2}$ is similar to that trained on `PY8` events. Once $N^{(0)}$ is additionally considered in the classification, the performance is improved. However, the improvement from adding $N^{(0)}$ is significantly smaller in `HW7`, because the $N^{(0)}$ distributions of top jets and QCD jets are close, as shown in figure 10.

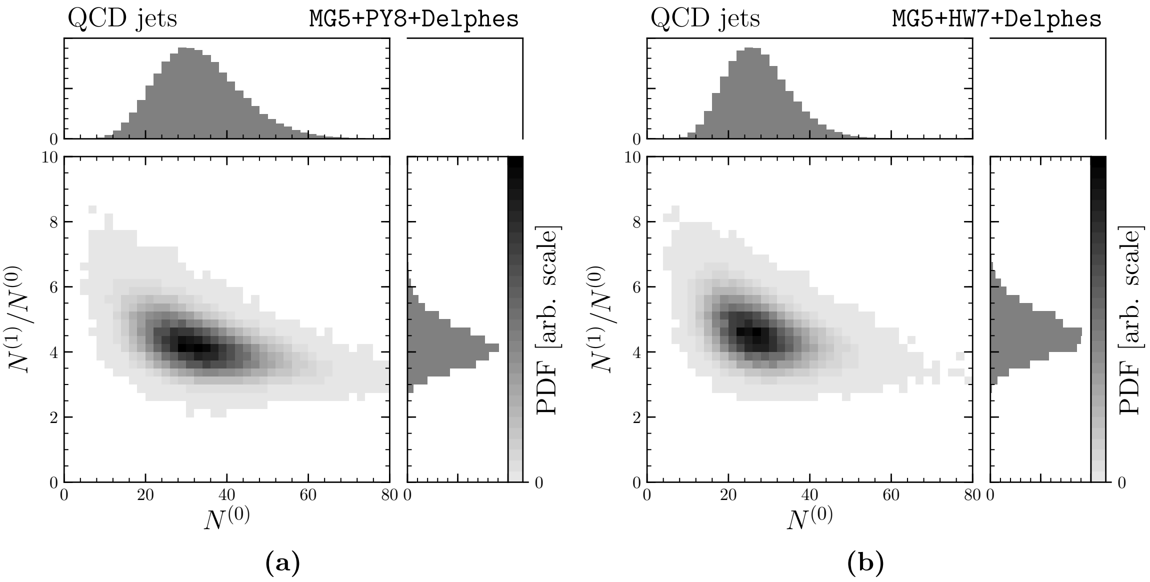
In the previous analysis, we have shown that inputs $N^{(0)}$, $N^{(0)}(4\,\mathrm{GeV})$, $N^{(1)}$, and $N^{(1)}(4\,\mathrm{GeV})$ in addition to $S_{2,ab}$ is good enough for building a neural network that fits the CNN output. At the same time, this indicates that tuning of the event generator focusing on these counting variables can be an efficient way to obtain the simulated data that gives consistent results with the experimental data.

[9] Dipole shower also can be used, but we do not study the model in this paper.



**(a)** **(b)**

**Figure 17.** $(N^{(0)}, N^{(0)}(4\,\mathrm{GeV}))$ distributions for (a) `PY8` and (b) `HW7`.



**(a)** **(b)**

**Figure 18.** $(N^{(0)}, N^{(1)}/N^{(0)})$ distributions for (a) `PY8` and (b) `HW7`.

If the difference between the simulated and experimental data is not too large, reweighting simulated events is useful for reducing the difference. We consider reweighting based on the marginal distribution of interested variables $\boldsymbol{x}$. Let $\rho_{\mathrm{true}}(\boldsymbol{x})$ and $\rho_{\mathrm{MC}}(\boldsymbol{x})$ be the $\boldsymbol{x}$ distributions with true and simulated events, respectively. The new weight $w_{\mathrm{new}}^{[i_Y]}$ of the event $i_Y$ is given as follows,

$$w_{\mathrm{new}}^{[i_Y]} = \frac{\rho_{\mathrm{true}}(\boldsymbol{x}^{[i_Y]})}{\rho_{\mathrm{MC}}(\boldsymbol{x}^{[i_Y]})} \cdot w_{\mathrm{old}}^{[i_Y]}, \tag{5.1}$$

where $w_{\mathrm{old}}^{[i_Y]}$ is the weight before reweighting.

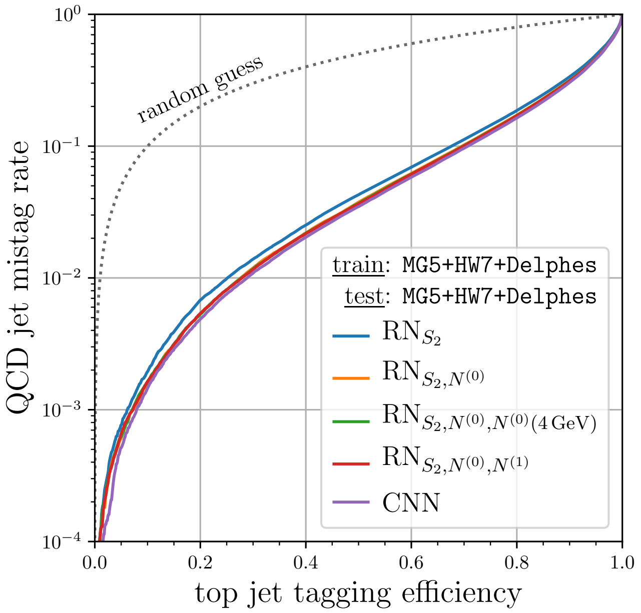


**Figure 19.** The ROC curves of the networks trained on `HW7` samples.

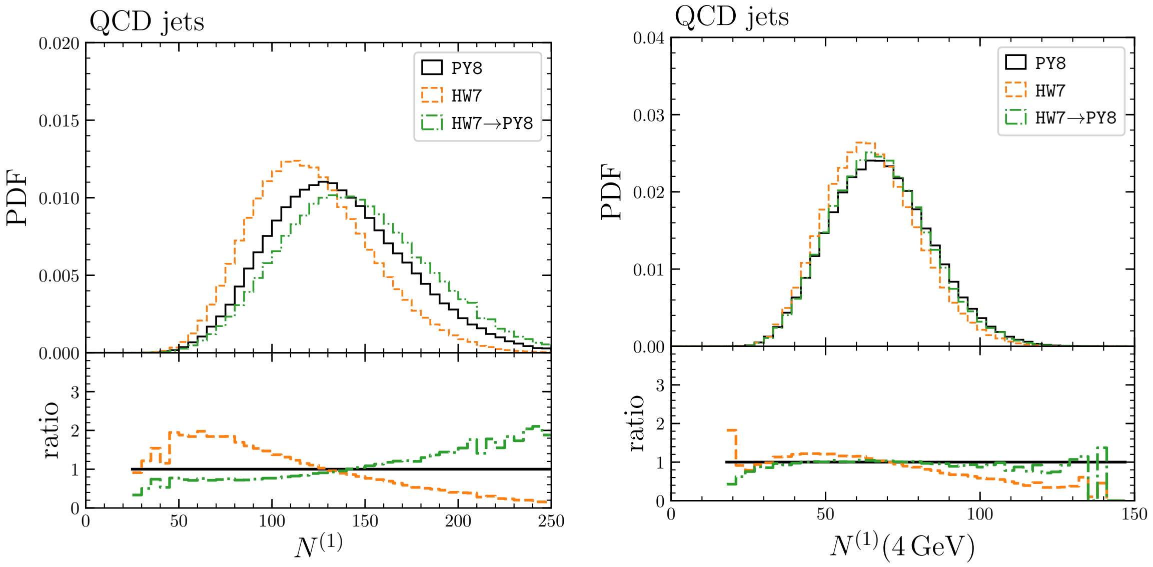


**Figure 20.** The $N^{(1)}$ and $N^{(1)}(4\,\mathrm{GeV})$ distributions of `PY8`, `HW7`, and reweighted `HW7` samples.

Let us perform an exercise to correct $(N^{(0)}, N^{(0)}(4\,\mathrm{GeV}))$ distribution, assuming that either one of the distributions $\rho_{\texttt{PY8}}$ and $\rho_{\texttt{HW7}}$ simulated by `PY8` and `HW7` is $\rho_{\mathrm{true}}$ while the other is $\rho_{\mathrm{MC}}$. We consider the reweighting of these two variables in order to consider a non-trivial case that some of the variables are correlated. The reweighting factor $\rho_{\mathrm{true}}(N^{(0)}, N^{(0)}(4\,\mathrm{GeV}))/\rho_{\mathrm{MC}}(N^{(0)}, N^{(0)}(4\,\mathrm{GeV}))$ is calculated using the normalized histogram of $(N^{(0)}, N^{(0)}(4\,\mathrm{GeV}))$, as stated in appendix E. The $N^{(1)}$ distribution in figure 20 still disagree after the reweighing, but the deviation is minor. Because the sample size is limited, we do not attempt to correct all those distribution in this paper.

Figure 21 shows the $\hat{y}^0$ distributions for QCD jet samples, of the models trained on `PY8` samples. The orange dashed, black solid, and green dot-dashed histograms are the $\hat{y}^0$ distributions with `HW7`, `PY8`, and reweighted `HW7` samples, respectively.

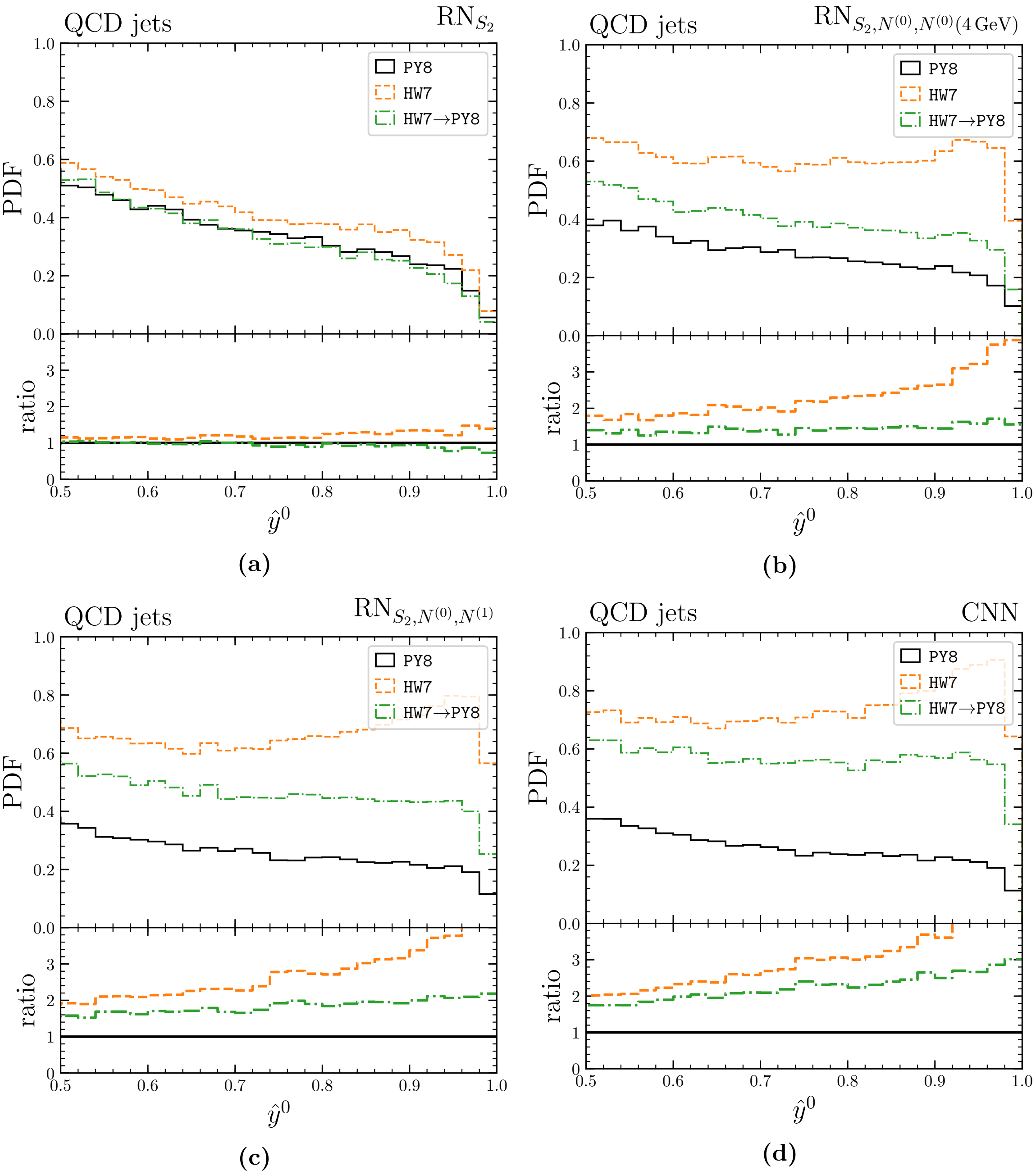
Figure 21(a) shows the $\hat{y}^0$ distributions of $\mathrm{RN}_{S_2}$. This classifier does not use $N^{(0)}$ and $N^{(0)}(4\,\mathrm{GeV})$ explicitly, but the distribution of reweighted `HW7` samples comes quite close to that of `PY8` samples. The score difference comes from the difference of $S_2$ distribution. The $S_{2,\mathrm{soft}}$ distribution of `HW7` samples is 10% smaller than that of `PY8` samples. The reweighting reduces the difference because $N^{(0)}$ and $S_{2,\mathrm{soft}}$ are correlated. The Pearson correlation coefficient between $N^{(0)}$ and $S^{(i)}_{2,\mathrm{soft}}$ is 0.3 for both `PY8` and `HW7`.[10] The bin-by-bin ratio of the average $\langle S^{(i)}_{2,\mathrm{soft}}\rangle$ between `HW7` and `PY8` samples is about 0.9. The average $\langle S^{(i)}_{2,\mathrm{soft}}\rangle$ of `HW7` samples increases after the reweighting and the $S_{2,\mathrm{soft}}$ distributions get closer to each other. On the other hand, the reweighting increases the disagreement of $p_{T,\mathbf{J}}$ distribution. The sum of the weights of the reweighted `HW7` samples with $p_{T,\mathbf{J}} \sim 500\,\mathrm{GeV}$ is about 20% larger than that of `PY8` samples. We marginalized $p_{T,\mathbf{J}}$ during the training so that the impact on the $\hat{y}^0$ distribution is minimal. Therefore, the agreement seen in figure 21(a) is mainly due to the correction of $S_{2,ab}$, and it is encouraging.

Figure 21(b) shows the $\hat{y}^0$ distributions of $\mathrm{RN}_{S_2,N^{(0)},N^{(0)}(4\,\mathrm{GeV})}$. For $\hat{y}^0 \sim 1$, the ratio of the $\hat{y}^0$ distributions of `HW7` and `PY8` exceeds 4, and the distribution of `HW7` samples even peaks near $\hat{y}^0 \sim 1$. This means that the model trained on `PY8` samples focuses on a particular region in order to get high purity top samples, but the `HW7` samples are still populated in the region. In the situation that the `HW7` distribution is "true" while `PY8` samples are used to build the top jet vs. QCD jet classifier, we overestimate the top quark event rate by dijet contamination; adding the variables whose "true distributions" are not well understood could cause the problem of this kind.

The ratio between weighted `HW7` and `PY8` distributions is constant. This is nice in order to avoid the systematics along with tightening the cut to reject QCD events. On the other hand, the ratio of the reweighted `HW7` samples is much larger than that in figure 21(a). The deviation should come from the mismodeling of the correlation between $N^{(0)}$ and other parameters. The difference is even larger if one includes $N^{(1)}$ in the inputs, as shown in figure 21(c). The ratio between the weighted `HW7` and `PY8` sample is now nearly a factor of two larger at $y \sim 1$ and even increasing. This disagreement is not surprising given the very poor sample of `HW7` in the high $N^{(0)}$ region. Finally, figure 21(d) is the $\hat{y}^0$ distribution of the CNN model. The distributions looks quite similar to those in figure 21(c) before reweighting, but the ratio of the distributions of reweighted `HW7` events and `PY8` events is larger than that of $\mathrm{RN}_{S_2,N^{(0)},N^{(1)}}$.

Figure 22 shows the $\hat{y}^0$ distributions of the model $\mathrm{RN}_{S_2,N^{(0)},N^{(0)}(4\,\mathrm{GeV})}$ trained on `HW7` events. Recall that the QCD jets in `PY8` samples cover the phase space of the QCD jets simulated by `HW7`, and the reweighting is then effective for transforming the `PY8` samples to `HW7` samples. The opposite is not true because there are QCD jets which are not in `HW7` generated samples. The reweighting is not exact because we have only a small number of

[10] The correlation between $N^{(0)}$ and $S^{(i)}_{2,\mathrm{trim}}$ is around 0.15 for the bins dominated by the cross-correlation between high $p_T$ constituents.



(a)

(b)

(c)

(d)

**Figure 21.** The $\hat{y}^0$ distributions of `PY8` and `HW7` test samples for the model trained on the `PY8` events. The neural networks used in the plots are (a) $\mathrm{RN}_{S_2}$, (b) $\mathrm{RN}_{S_2,N^{(0)},N^{(0)}(4\,\mathrm{GeV})}$, (c) $\mathrm{RN}_{S_2,N^{(0)},N^{(1)}}$, and (d) CNN.

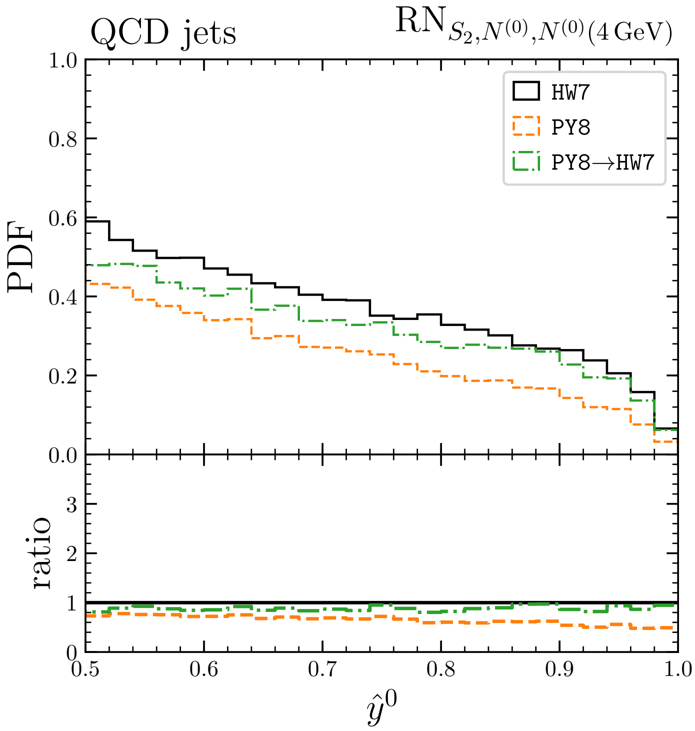


**Figure 22.** The score distribution of PY8 and HW7 test sample for the model trained by the HW7 events.

events in some phase space region, and we see some deviation in $\hat{y}$ distribution, as shown in figure 21(b). If one wishes to describe real data by assigning an appropriate weight for each simulated events, it is better to use a generator setup that covers wider phase space so that we can correct the event distribution by using experimental data afterwords.

The disagreement between PY8 and HW7 samples remains after the reweighting in this exercise. We do not proceed to reweight distributions other than the $(N^{(0)}, N^{(0)}(4\,\mathrm{GeV}))$ distribution in this paper because of the statistical limitation. Neural network-based reweighing [176] can be helpful for adjusting full phase-space, but it is beyond the scope of this paper. The difference between the two generators is too large to achieve perfect agreement simply by reweighting. Because $N^{(0)}$ and $N^{(1)}$ are important quantities for describing the neural network-based classifier, those generators may be tuned carefully to reproduce the distribution of soft activities in jet images.

## 6 Discussions

In this paper, we have identified essential quantities that the CNN on a jet image is using for the top jet vs. QCD jet classification. The discovered quantities consist of both IRC safe and IRC unsafe observables. The former includes the IRC safe two-point energy correlation, jet spectrum, as a function of the distance between two jet constituents. The latter is an IRC unsafe Minkowski sequence inspired from the Minkowski functionals that describes morphological information on the set of jet constituents. It gives a quantitative measure of the area that is occupied by the particles inside jets. The first element of the Minkowski sequence is the number of active pixels in the jet image, $N^{(0)}$, and the second element $N^{(1)}$ is the sum of the $N^{(0)}$ and the number of the pixels adjacent to the active pixels. These quantities are derivable from a jet image, and the relation network (RN) trained on these quantities (along with kinematic observables) has equivalent performance to the CNN.

The IRC safe quantities are theoretically more controlled, especially different event simulators predict consistent distributions. On the other hand, the IRC unsafe quantities are described by phenomenological models tuned by the experimental data. The classification performance of RN agrees with that of CNN only when we include IRC unsafe Minkowski sequences among the inputs. The similarity of the performance indicates that the top jet classifier based on CNN uses the geometric information of soft radiation, and we have succeeded in reproducing the CNN predictions using fewer degrees of freedom.

We also point out that the training of the RN is more stable than the CNN. The stability comes from the fact that the RN classifiers use a restricted set of derived inputs from the jet images, and the loss function of the RN is less complicated than that of CNN. We measure the variation of the training results by randomly swapping the event orders in the batch training and using a different initial parameter in the networks. The variation of the RN output is about factor 3 smaller than that of the CNN output, as we have seen in table 2.

As the IRC safe inputs, we choose the jet spectrum [19, 21], which is aggregated two-point energy correlation as the function of the $\Delta R$. We introduce the various improvements on the jet spectrum from the previous paper. In this paper, it is derived from a constrained graph network. A vertex of the graph network corresponds to a jet constituent, while each vertex carries information of the constituent momentum and the subjet ID to which the constituent belongs. The edges links between two vertices and represent the two-point correlation between the two constituents. For the classification of top jets and QCD jets, we find that the correlation among the trimmed jet and the correlation between the leading subjet and the other constituents, and their geometry are especially useful in the classification. We systematically include the three-point structure of the top quark in the two-point energy correlations after removing the leading subjet. The modularized networks process the two-point correlations separately with global kinematical inputs so that the combined network accepts significantly more inputs without inflating the parameters in the hidden layers.

The classifiers using the IRC unsafe quantities, such as soft pixels of jet image or the Minkowski sequence, could suffer from systematic uncertainties of the simulation. After the identification of the key morphological quantities, we can minimize efforts on calibration by focusing on the $N^{(0)}$ and $N^{(1)}$ distributions. The distributions may be corrected relatively easily by reweighting events to calibrate the distributions to the observed data. We demonstrate that the reweighting of the simulated events to reproduce the true $N^{(0)}$ distributions greatly reduce the systematic error of the classifiers. Such tuning of the data reduces the systematic uncertainties in the ML classifications that depend on the simulated events.

In summary, we propose an approach to replace a complex neural network using the low level inputs into a simple network using the processed inputs motivated from a physics point of view. To this end, we show surprising evidence that the CNN output depends on the geometrical measures expressed by discritized version of the Minkowski functionals. These morphological quantities improve jet classification significantly. The study of jet morphology from the data, and comparison to the prediction from event simulation might be an exciting direction to persuade. We think the variables may be further extended not only for jet physics but also for the analysis of event geometry or anomaly searches.

## Acknowledgments

The authors would like to thank Marat Freytsis, Eun-Sol Kim (in kakaobrain), Andrew J. Larkoski, Benjamin Nachman, Maurizio Pierini, David Shih, and Takashi Tsuboi for useful discussions. This work is supported by the Grant-in-Aid for Scientific Research on Scientific Research B (No. 16H03991, 17H02878) and Innovative Areas (16H06492); World Premier International Research Center Initiative (WPI Initiative), MEXT, Japan; and the Department of Science and Technology, Government of India, under Grant No. IFA18-PH 224 (INSPIRE Faculty Award). M.T. is supported in part by the JSPS Grant-in-Aid for Scientific Research No. 16H03991, 16H02176, 18K03611, and 19H04613.

## A Setup for Monte-Carlo event simulation

We generate $pp \to t\bar{t}$ and $pp \to jj$ events for top jet and QCD jet samples, respectively. The symbol $j$ represents gluon or (anti-)quark other than the top quark. The parton level events are generated by `Madgraph5 2.6.6` [152]. The center of mass energy is 13 TeV. Produced top quarks are forced to decay into $bW$ and the subsequent $W$ boson decays into two quarks including $b$-quarks. Since we are only interested in boosted top quarks, we generate events with outgoing partons whose $p_T$ is larger than 450 GeV. Numbers of the generated $pp \to t\bar{t}$ and $pp \to jj$ events with this preselection are 5 million and 10 million, respectively. The renormalization and factorization scales are set to be $H_T/2$, where $H_T$ is the sum of the transverse energy of each parton, and the parton distribution function is `NNPDF23_lo_as_0130_qed` [72–75]. Two parton shower and hadronization simulations are considered in this paper: `Pythia 8.226` [153] with Monash tune [177] and `Herwig 7.1.3` [154, 155] with default tune [178, 179]. The pile-ups are not included but the underlying events and multi-parton interactions are considered.

We use `Delphes 3.4.1` [156] for detector simulation with its default ATLAS detector configuration. Jets are reconstructed from calorimeter towers whose $(\eta, \phi)$ resolutions at electromagnetic and hadronic calorimeters in $|\eta| < 2.5$ are assumed to be $(0.0174, 1^\circ)$ and $(0.1, 10^\circ)$, respectively. Anti-$k_T$ jet clustering algorithm [180] with radius parameter $R_{\mathbf{J}} = 1.0$ implemented in `fastjet 3.3.0` [181, 182] is used to cluster these calorimeter towers into jets. The leading $p_T$ jets with its transverse momentum $p_{T,\mathbf{J}} \in [500, 600]$ GeV and mass $m_{\mathbf{J}} \in [150, 200]$ GeV are selected for the analysis. In addition, a top jet sample is required to have quarks from the originating top quark within $R_{\mathbf{J}}$ from the jet axis. After this selection, we have about 950,000 top jets and 350,000 QCD jets. Half of them are used for the training and th others are used for testing. For jet trimming, we use $k_T$

algorithm [183, 184] with radius 0.2 and keep subjets whose energy fraction is larger than 0.05. The leading $p_T$ subjet $\mathbf{J}_1$ is the highest $p_T$ anti-$k_T$ subjet [180] with radius 0.2.

Note that we have not used matched sample, so that the modeled $p_{T,\mathbf{J}}$ distribution is not precise beyond the leading order accuracy. Nevertheless, the changes due to recoiling from extra radiation are not a main interest in this paper, so we use this samples by presuming that the top jets and QCD jets are factorizable.

# B Kernel density estimation of $p_{T,\mathbf{J}}$ distribution

We use the kernel density estimation (KDE) on a finite interval $[p_{T,\mathbf{J}}^{\min}, p_{T,\mathbf{J}}^{\max}]$ to model the event-by-event weight $f_{p_{T,\mathbf{J}}}(p_{T,\mathbf{J}}; Y)$ in eq. (4.13). First, we transform $p_{T\,jet}$ into a logit $t(p_{T,\mathbf{J}})$ in order to make the domain unbounded.

$$t(p_{T,\mathbf{J}}) = \text{logit}\left(\frac{p_{T,\mathbf{J}} - p_{T,\mathbf{J}}^{\min}}{p_{T,\mathbf{J}}^{\max} - p_{T,\mathbf{J}}^{\min}}\right) = -\log\frac{p_{T,\mathbf{J}} - p_{T,\mathbf{J}}^{\min}}{p_{T,\mathbf{J}}^{\max} - p_{T,\mathbf{J}}} \tag{B.1}$$

The KDE of the sampled logits, $t(p_{T,\mathbf{J}}^{[i_{N_Y}]})$ is used to estimate the probability density function $f_{P_{T,\mathbf{J}}}(p_{T,\mathbf{J}}; Y)$.

$$f_{P_{T,\mathbf{J}}}(p_{T,\mathbf{J}}; Y) \approx \frac{t'(p_{T,\mathbf{J}})}{N_Y}\sum_{i_Y=1}^{N_Y} K_h(t(p_{T,\mathbf{J}}) - t(p_{T,\mathbf{J}}^{[i_Y]})), \quad t'(p_{T,\mathbf{J}}) = \frac{p_{T,\mathbf{J}}^{\max} - p_{T,\mathbf{J}}^{\min}}{(p_{T,\mathbf{J}}^{\max} - p_{T,\mathbf{J}})(p_{T,\mathbf{J}} - p_{T,\mathbf{J}}^{\min})} \tag{B.2}$$

where $K_h$ is a scaled kernel whose bandwith parameter is $h$. In particular, a gaussian kernel with bandwith $h = 0.25$ is used for the KDE.

$$K_h(x) = \frac{1}{h}K\left(\frac{x}{h}\right), \quad K(x) = \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{x^2}{2}\right) \tag{B.3}$$

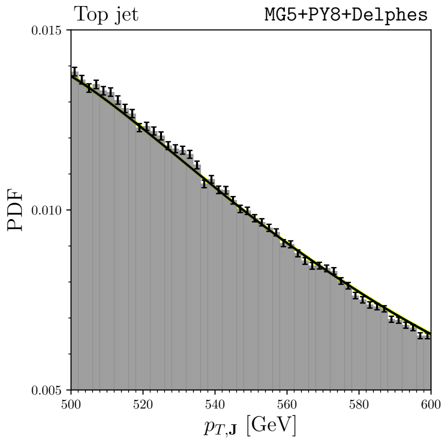
However, $t'(p_{T,\mathbf{J}})$ is singular at $p_{T,\mathbf{J}}^{\min}$ and $p_{T,\mathbf{J}}^{\max}$, and the estimation of the probability density near the boundary is less precise. Instead of using samples after the selection $p_{T,\mathbf{J}} \in [500, 600]$ GeV, we use a selection with broader $p_{T,\mathbf{J}}$ range, $[450, 650]$ GeV for KDE only in order to avoid the effects from the singularities. The KDE is then normalized for $p_{T,\mathbf{J}} \in [500, 600]$ GeV afterward. We show the normalized histogram of $p_{T,\mathbf{J}}$ and the KDE in figure 23.

# C Network implementations

## C.1 Relation networks

The relation networks used in this paper are implemented as follows. The module for analyzing the energy correlation with jet trimming, $\boldsymbol{h}_{\text{trim}} = \text{MLP}_{\text{trim}}(\boldsymbol{x}_{\text{trim}}, \boldsymbol{x}_{\text{kin}})$, consists of two hidden layers,

$$\begin{aligned}
\boldsymbol{h}_{\text{trim}}^{(1)} &= \text{FC}(\boldsymbol{z}_{\text{trim}}, \boldsymbol{z}_{\text{kin}}), && \text{size: } 200, && \text{activation: ELU} \\
\boldsymbol{h}_{\text{trim}}^{(2)} &= \text{FC}(\boldsymbol{h}_{\text{trim}}^{(1)}), && \text{size: } 200, && \text{activation: ELU} \\
\boldsymbol{h}_{\text{trim}} &= \text{FC}(\boldsymbol{h}_{\text{trim}}^{(2)}), && \text{size: } 5, && \text{activation: linear}
\end{aligned} \tag{C.1}$$



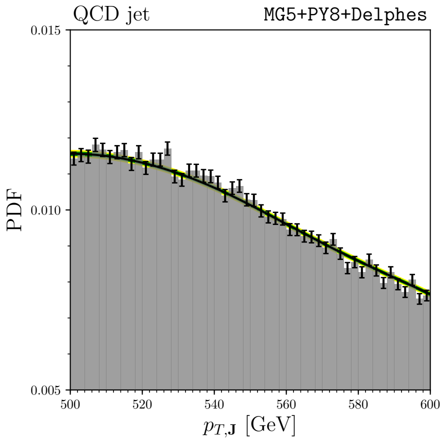


**Figure 23.** The histogram and modeled probability density distribution of $p_{T,\mathbf{J}}$ for (left) top jet samples and (right) QCD jet samples. Solid black line is the KDE. The vertical bar is the statistical uncertainty of each bin. The green and yellow bands are pointwise statistical uncertainty of the KDE calculated from squared sum of the summands.

where $\boldsymbol{z}_i$ is the standardized inputs of $\boldsymbol{x}_i$, and FC is a fully-connected layer with a given output size and activation function. Note that we do not apply $L_2$ regularization for the FCs with linear activation. The module for analyzing the energy correlation of $\mathbf{J}_1$ and $\mathbf{J} \setminus \mathbf{J}_1$ is as follows.

$$\begin{aligned}
\boldsymbol{h}^{(1)}_{\mathbf{J}_1} &= \mathrm{FC}(\boldsymbol{z}_{\mathbf{J}_1}, \boldsymbol{z}_{\mathrm{kin}}), && \text{size: } 200, && \text{activation: ELU} \\
\boldsymbol{h}^{(2)}_{\mathbf{J}_1} &= \mathrm{FC}(\boldsymbol{h}^{(1)}_{\mathbf{J}_1}), && \text{size: } 200, && \text{activation: ELU} \\
\boldsymbol{h}_{\mathbf{J}_1} &= \mathrm{FC}(\boldsymbol{h}^{(2)}_{\mathbf{J}_1}), && \text{size: } 5, && \text{activation: linear}
\end{aligned} \tag{C.2}$$

The logits $\boldsymbol{u}'$ for the binary classification is implemented as follows.

$$\begin{aligned}
\boldsymbol{h}^{(1)}_{\mathrm{logit}} &= \mathrm{FC}(\boldsymbol{h}_{\mathrm{trim}}, \boldsymbol{h}_{\mathbf{J}_1}, \boldsymbol{z}_{\mathrm{kin}}), && \text{size: } 200, && \text{activation: ELU} \\
\boldsymbol{h}^{(2)}_{\mathrm{logit}} &= \mathrm{FC}(\boldsymbol{h}^{(1)}_{\mathrm{logit}}), && \text{size: } 200, && \text{activation: ELU} \\
\boldsymbol{u}' &= \mathrm{FC}(\boldsymbol{h}^{(2)}_{\mathrm{logit}}), && \text{size: } 2, && \text{activation: linear}
\end{aligned} \tag{C.3}$$

For the relation networks with inputs $\boldsymbol{x}_{\mathrm{geometry}}$, we replace $\boldsymbol{h}^{(1)}_{\mathrm{logit}}$ of eq. (C.3) as follows.

$$\boldsymbol{h}^{(1)}_{\mathrm{logit}} = \mathrm{FC}(\boldsymbol{h}_{\mathrm{trim}}, \boldsymbol{h}_{\mathbf{J}_1}, \boldsymbol{z}_{\mathrm{geometry}}), \quad \text{size: } 200, \quad \text{activation: ELU}, \tag{C.4}$$

### C.2 Convolutional neural networks

Our convolutional neural networks are trained on the preprocessed jet images obtained as in [21]. We recluster given jet constituents by $k_T$ algorithm [183, 184] with radius parameter $R_{\mathbf{J}} = 0.2$ and translate the $(\eta, \phi)$ coordinate so that the leading $p_T$ subjet axis is at $(0, 0)$. If a subleading $p_T$ subjet exists, we rotate the $(\eta, \phi)$ corordinate about the origin so that the subjet is on the positive $y$-axis on the rotated coordinate. If a third leading $p_T$ subjet exists with a negative $x$ coordinate, we reflect the coordinate to across the $y$ axis so that the third leading $p_T$ subjet always has a positive $x$ coordinate. The preprocessed jet image $\boldsymbol{x}_{\text{image}}$ is a two-dimensional $p_T$-weighted histogram of those regularized constituents on a range $[-1.5, 1.5] \otimes [-1.5, 1.5]$ with bin size $0.1 \times 0.1$. The energy deposit of each pixel is standardized thereafter.

The vanilla CNN of this paper consists of six convolutional layers with a filter size $3 \times 3$. The standardized image $\boldsymbol{z}_{\text{image}}$ of $\boldsymbol{x}_{\text{image}}$ is fed into a chain of convolutional layers as follows.

$$\begin{aligned}
\boldsymbol{h}^{(1)}_{\text{CNN}} &= \text{CONV}(\boldsymbol{z}_{\text{image}}), && \text{size: } 30\times30\times16, && \text{filter size: } 3\times3, && \text{activation: ELU},\\
\boldsymbol{h}^{(2)}_{\text{CNN}} &= \text{CONV}(\boldsymbol{h}^{(1)}_{\text{CNN}}), && \text{size: } 30\times30\times16, && \text{filter size: } 3\times3, && \text{activation: ELU},\\
\boldsymbol{h}^{(3)}_{\text{CNN}} &= \text{CONV}(\boldsymbol{h}^{(2)}_{\text{CNN}}), && \text{size: } 30\times30\times16, && \text{filter size: } 3\times3, && \text{activation: ELU},\\
\boldsymbol{h}^{(3,\text{POOL})}_{\text{CNN}} &= \text{POOL}(\boldsymbol{h}^{(3)}_{\text{CNN}}), && \text{size: } 15\times15\times16, && \text{pool size: } 2\times2, &&\\
\boldsymbol{h}^{(4)}_{\text{CNN}} &= \text{CONV}(\boldsymbol{h}^{(3,\text{POOL})}_{\text{CNN}}), && \text{size: } 15\times15\times8, && \text{filter size: } 3\times3, && \text{activation: ELU},\\
\boldsymbol{h}^{(5)}_{\text{CNN}} &= \text{CONV}(\boldsymbol{h}^{(4)}_{\text{CNN}}), && \text{size: } 15\times15\times8, && \text{filter size: } 3\times3, && \text{activation: ELU},\\
\boldsymbol{h}^{(6)}_{\text{CNN}} &= \text{CONV}(\boldsymbol{h}^{(5)}_{\text{CNN}}), && \text{size: } 15\times15\times8, && \text{filter size: } 3\times3, && \text{activation: ELU},\\
\boldsymbol{h}^{(6,\text{POOL})}_{\text{CNN}} &= \text{POOL}(\boldsymbol{h}^{(6)}_{\text{CNN}}), && \text{size: } 7\times7\times8, && \text{pool size: } 2\times2, &&\\
\boldsymbol{h}^{(7)}_{\text{CNN}} &= \text{FC}(\boldsymbol{h}^{(6,\text{POOL})}_{\text{CNN}}), && \text{size: } 200, && \text{activation: ELU}, &&\\
\boldsymbol{h}_{\text{CNN}} &= \text{FC}(\boldsymbol{h}^{(7)}_{\text{CNN}}), && \text{size: } 100, && \text{activation: linear}, &&
\end{aligned} \tag{C.5}$$

where CONV is a two-dimensional convolutional layer with a given filter size and activation function, and POOL is a max-pooling layer with a given pool size. The output size consists of three numbers: the first two numbers represent output image width and height, and the third number is the number of filters. We simply put $\boldsymbol{h}_{\text{CNN}}$ to $\text{MLP}_{\text{logit}}$ by replacing eq. (C.3) to the following.

$$\boldsymbol{h}^{(1)}_{\text{logit}} = \text{FC}(\boldsymbol{h}_{\text{CNN}}, \boldsymbol{z}_{\text{kin}}), \qquad \text{size: } 200, \quad \text{activation: ELU} \tag{C.6}$$

The ResNet in section 4.4 consists of convolutional layers $\boldsymbol{h}^{(i+1,\text{res})}_{\text{ResNet}}$ with skip connection $\boldsymbol{h}^{(i+1,\text{shortcut})}_{\text{ResNet}}$. We define a ResNet module of input image $\boldsymbol{h}^{(i)}_{\text{ResNet}}$ as follows.

$$\begin{aligned}
\boldsymbol{h}^{(i+1,\text{res})}_{\text{ResNet}} &= \text{CONV} \circ \text{ELU} \circ \text{CONV}(\boldsymbol{h}^{(i)}_{\text{ResNet}}),\\
\boldsymbol{h}^{(i+1,\text{shortcut})}_{\text{ResNet}} &= \begin{cases} \boldsymbol{h}^{(i)}_{\text{ResNet}} & \text{if } \boldsymbol{h}^{(i)}_{\text{ResNet}} \text{ and } \boldsymbol{h}^{(i+1)}_{\text{ResNet}} \text{ has the same size,}\\ \text{CONV}_{1\times1}(\boldsymbol{h}^{(i)}_{\text{ResNet}}) & \text{otherwise,} \end{cases}\\
\boldsymbol{h}^{(i+1)}_{\text{ResNet}} &= \text{ResNet}(\boldsymbol{h}^{(i)}_{\text{ResNet}}) = \text{ELU}(\boldsymbol{h}^{(i+1,\text{res})}_{\text{ResNet}} + \boldsymbol{h}^{(i+1,\text{shortcut})}_{\text{ResNet}})
\end{aligned} \tag{C.7}$$

where $\text{CONV}_{1\times1}$ is a convolutional layer with filter size $1 \times 1$. The hyperparameters of other CONV will be specified later. All the convolutional operations above do not have any activation function. If input image size and output image size are different, we use strided convolution on $\text{CONV}(\boldsymbol{h}^{(i)}_{\text{ResNet}})$. We build a ResNet by replacing the chain of convolutional layers in eq. (C.5) to the following chain of six ResNet modules.

$$\begin{aligned}
\boldsymbol{h}^{(1)}_{\rm ResNet} &= {\rm ResNet}(\boldsymbol{z}_{\rm image}), && \text{size: } 30\times30\times16, && \text{filter size: } 3\times3,\\
\boldsymbol{h}^{(2)}_{\rm ResNet} &= {\rm ResNet}(\boldsymbol{h}^{(1)}_{\rm ResNet}), && \text{size: } 30\times30\times16, && \text{filter size: } 3\times3,\\
\boldsymbol{h}^{(3)}_{\rm ResNet} &= {\rm ResNet}(\boldsymbol{h}^{(2)}_{\rm ResNet}), && \text{size: } 15\times15\times8, && \text{filter size: } 3\times3, \quad \text{stride: } 2,\\
\boldsymbol{h}^{(4)}_{\rm ResNet} &= {\rm ResNet}(\boldsymbol{h}^{(3)}_{\rm ResNet}), && \text{size: } 15\times15\times8, && \text{filter size: } 3\times3,\\
\boldsymbol{h}^{(5)}_{\rm ResNet} &= {\rm ResNet}(\boldsymbol{h}^{(4)}_{\rm ResNet}), && \text{size: } 8\times8\times8, && \text{filter size: } 3\times3, \quad \text{stride: } 2,\\
\boldsymbol{h}^{(6)}_{\rm ResNet} &= {\rm ResNet}(\boldsymbol{h}^{(5)}_{\rm ResNet}), && \text{size: } 8\times8\times8, && \text{filter size: } 3\times3,\\
\boldsymbol{h}^{(1)}_{\rm logit} &= {\rm FC}(\boldsymbol{h}^{(6)}_{\rm ResNet}, \boldsymbol{z}_{\rm kin}), && \text{size: } 200, \quad \text{activation: ELU.}
\end{aligned} \tag{C.8}$$

The ResNeXt in section 4.4 uses multiple chains of convolutional layers for the residual learning parts $\boldsymbol{h}^{(i+1,\rm res)}_{\rm ResNet}$ in the ResNet. The ResNeXt module with four parallel chains of convolutional layers is defined as follows.

$$\begin{aligned}
\boldsymbol{h}^{(i+1,j)}_{\rm ResNeXt} &= {\rm ELU}\circ{\rm CONV}\circ{\rm ELU}\circ{\rm CONV}_{1\times1}(\boldsymbol{h}^{(i)}_{\rm ResNeXt}),\\
\boldsymbol{h}^{(i+1,\rm res)}_{\rm ResNeXt} &= {\rm CONV}_{1\times1}\left(\bigoplus_{j=1}^{4}\boldsymbol{h}^{(i+1,j)}_{\rm ResNeXt}\right),\\
\boldsymbol{h}^{(i+1,\rm shortcut)}_{\rm ResNeXt} &= {\rm CONV}_{1\times1}(\boldsymbol{h}^{(i)}_{\rm ResNeXt}),\\
\boldsymbol{h}^{(i+1)}_{\rm ResNet} &= {\rm ResNeXt}(\boldsymbol{h}^{(i)}_{\rm ResNeXt}) = {\rm ELU}\left(\boldsymbol{h}^{(i+1,\rm res)}_{\rm ResNeXt}+\boldsymbol{h}^{(i+1,\rm shortcut)}_{\rm ResNeXt}\right),
\end{aligned} \tag{C.9}$$

where the direct sum of the images represents a stacked image along the filter dimension. Since we use many convolutional layers already, we use three of those modules for the image analyzer.

$$\begin{aligned}
\boldsymbol{h}^{(1)}_{\rm ResNeXt} &= {\rm ResNeXt}(\boldsymbol{z}_{\rm image}), && \text{size: } 30\times30\times16, && \text{filter size: } 3\times3,\\
\boldsymbol{h}^{(2)}_{\rm ResNeXt} &= {\rm ResNeXt}(\boldsymbol{h}^{(1)}_{\rm ResNeXt}), && \text{size: } 30\times30\times16, && \text{filter size: } 3\times3,\\
\boldsymbol{h}^{(3)}_{\rm ResNeXt} &= {\rm ResNeXt}(\boldsymbol{h}^{(2)}_{\rm ResNeXt}), && \text{size: } 15\times15\times8, && \text{filter size: } 3\times3, \quad \text{stride: } 2,\\
\boldsymbol{h}^{(1)}_{\rm logit} &= {\rm FC}(\boldsymbol{h}^{(3)}_{\rm ResNeXt}, \boldsymbol{z}_{\rm kin}), && \text{size: } 200, \quad \text{activation: ELU}
\end{aligned}$$

# D Updating trainable parameters with moving averages

The moving average of a network parameter in section 4.1 is evaluated as follows. An updated parameter $\boldsymbol{\theta}^{(t)}$ at an epoch $t$ is accumulated into a moving average $\bar{\boldsymbol{\theta}}^{(t)}$,

$$\bar{\boldsymbol{\theta}}^{(t)} = \begin{cases} 0 & t < t_0 \\ \alpha\bar{\boldsymbol{\theta}}^{(t-1)} + (1-\alpha)\boldsymbol{\theta}^{(t)} & t \geq t_0 \end{cases} \tag{D.1}$$

where $\alpha = 0.9$. We accumulate only the updated parameters at the epochs after $t_0 = 50$. The solution to the recurrence relation is as follows,

$$\bar{\boldsymbol{\theta}}^{(t)} = \sum_{u=t_0}^{t}\alpha^{t-u}(1-\alpha)\boldsymbol{\theta}^{(u)}. \tag{D.2}$$

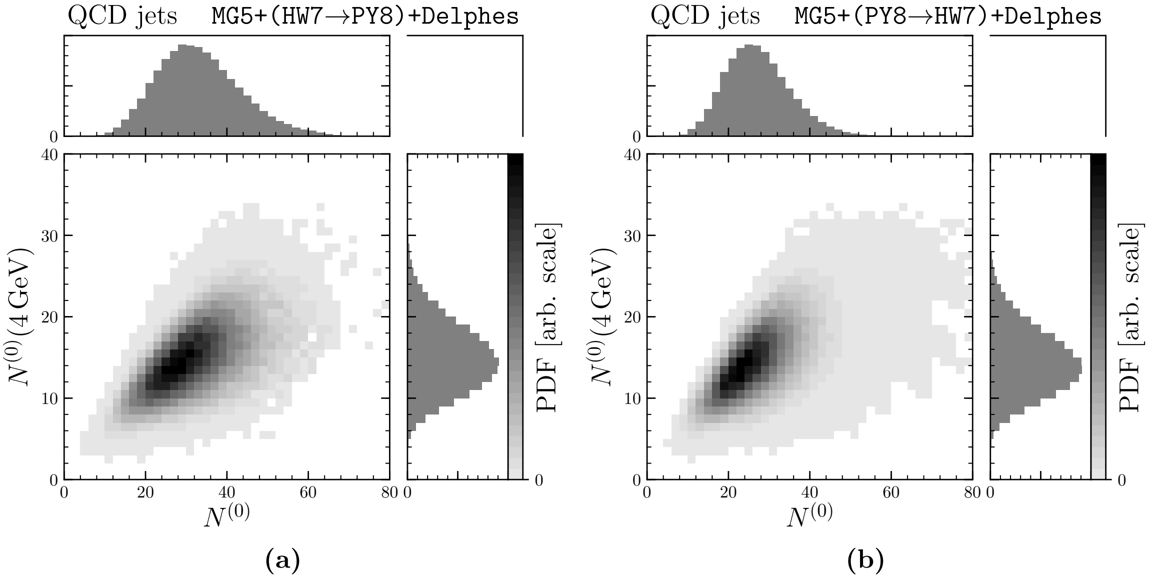


**Figure 24.** $(N^{(0)}, N^{(0)}(4\,\mathrm{GeV}))$ distribution for (a) the weighted `HW7` samples to reproduce `PY8` distribution and (b) the weighted `PY8` samples to reproduce `HW7` distribution.

As a side effect of the epoch selection, the sum of the weights in the average is not 1. As $\boldsymbol{\theta}^{(u)}$ approaches its optimum $\boldsymbol{\theta}_0$, $\bar{\boldsymbol{\theta}}^{(t)}$ approaches to $(1-\alpha^{t-t_0+1})\boldsymbol{\theta}_0$. The factor $1-\alpha^{t-t_0+1}$ should be corrected to make the moving average also converging to $\boldsymbol{\theta}_0$. We use the following unbiased moving average $\hat{\boldsymbol{\theta}}^{(t)}$ of the sequence of $\boldsymbol{\theta}^{(t)}$ for the validation and testing,

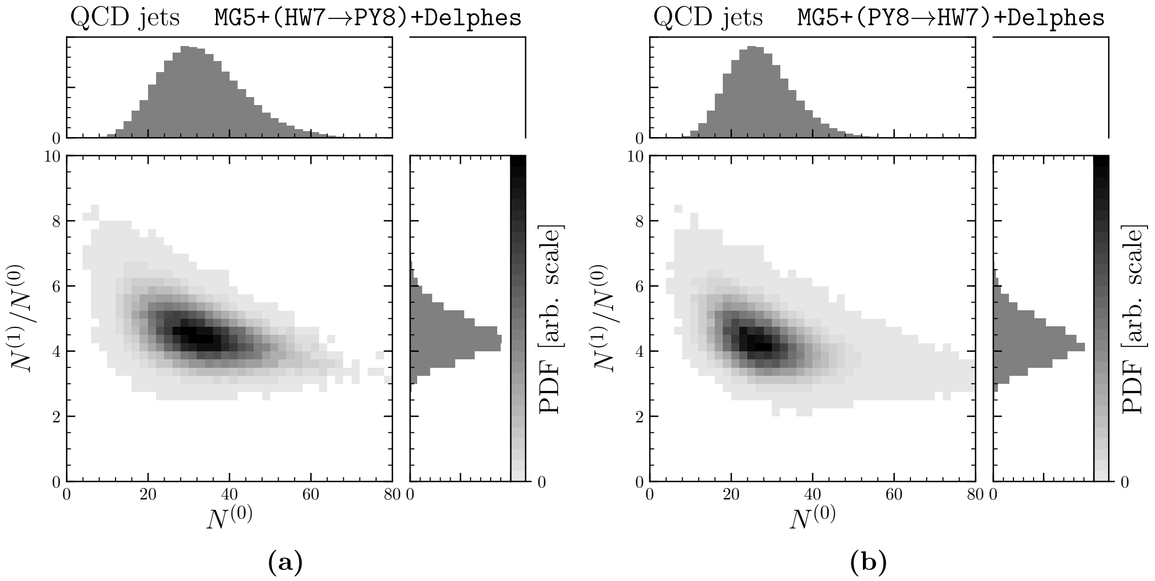
$$\hat{\boldsymbol{\theta}}^{(t)} = \frac{1}{1-\alpha^{t-t_0+1}}\bar{\boldsymbol{\theta}}^{(t)} \text{ for } t \geq t_0. \tag{D.3}$$

## E Evaluation of the reweighting factor

In section 5, we reweight the `HW7` generated events to `PY8` generated events by using $(N^{(0)}, N^{(0)}(4\,\mathrm{GeV}))$ distribution. Since the two numbers are correlated as shown in figure 17, we transform the data first and calculate the reweighting factor using normalized histograms in order to ensure the efficiency of the reweighting. The transformation of $(N^{(0)}, N^{(0)}(4\,\mathrm{GeV}))$ is defined as follows.

$$(x, y) \to (x', y') = (x, c_1 - c_2 y/x + c_3 x), \tag{E.1}$$

where $c_1 = 3/2$, $c_2 = 2$, and $c_3 = -1/60$. For each event, the reweighting factor in eq. (5.1) is calculated by the ratio of the corresponding bin values, $\rho_{\mathtt{PY8}}/\rho_{\mathtt{HW7}}$, where $\rho_A$ is the bin value of $(x', y')$ histogram with events generated by $A$. The reweighting factor for `PY8` generated events to obtain distributions of `HW7` generated events can be obtained by a similar procedure. The reweighted $(N^{(0)}, N^{(0)}(4\,\mathrm{GeV}))$ distribution and $(N^{(0)}, N^{(1)}/N^{(0)})$ distribution are shown in figure 24 and figure 25, respectively.



(a)

(b)

**Figure 25.** $N_{\text{pixel}}$ vs $N_1/N_{\text{pixel}}$ for (a) the weigthed `HW7` sample to reproduce `PY8` distribution and (b) the weighted `PY8` sample. to reproduce `HW7` distribution.

## References

[1] A.J. Larkoski, I. Moult and B. Nachman, *Jet Substructure at the Large Hadron Collider: A Review of Recent Advances in Theory and Machine Learning*, *Phys. Rept.* **841** (2020) 1 [`arXiv:1709.04464`] [INSPIRE].

[2] R. Kogler et al., *Jet Substructure at the Large Hadron Collider: Experimental Review*, *Rev. Mod. Phys.* **91** (2019) 045003 [`arXiv:1803.06991`] [INSPIRE].

[3] D. Guest, K. Cranmer and D. Whiteson, *Deep Learning and its Application to LHC Physics*, *Ann. Rev. Nucl. Part. Sci.* **68** (2018) 161 [`arXiv:1806.11484`] [INSPIRE].

[4] A. Radovic et al., *Machine learning at the energy and intensity frontiers of particle physics*, *Nature* **560** (2018) 41 [INSPIRE].

[5] M. Abdughani, J. Ren, L. Wu, J.M. Yang and J. Zhao, *Supervised deep learning in high energy phenomenology: a mini review*, *Commun. Theor. Phys.* **71** (2019) 955 [`arXiv:1905.06047`] [INSPIRE].

[6] L.G. Almeida, M. Backović, M. Cliche, S.J. Lee and M. Perelstein, *Playing Tag with ANN: Boosted Top Identification with Pattern Recognition*, *JHEP* **07** (2015) 086 [`arXiv:1501.05968`] [INSPIRE].

[7] L. de Oliveira, M. Kagan, L. Mackey, B. Nachman and A. Schwartzman, *Jet-images — deep learning edition*, *JHEP* **07** (2016) 069 [`arXiv:1511.05190`] [INSPIRE].

[8] P.T. Komiske, E.M. Metodiev and M.D. Schwartz, *Deep learning in color: towards automated quark/gluon jet discrimination*, *JHEP* **01** (2017) 110 [arXiv:1612.01551] [INSPIRE].

[9] A. Butter, G. Kasieczka, T. Plehn and M. Russell, *Deep-learned Top Tagging with a Lorentz Layer*, *SciPost Phys.* **5** (2018) 028 [arXiv:1707.08966] [INSPIRE].

[10] L.M. Dery, B. Nachman, F. Rubbo and A. Schwartzman, *Weakly Supervised Classification in High Energy Physics*, *JHEP* **05** (2017) 145 [arXiv:1702.00414] [INSPIRE].

[11] G. Kasieczka, T. Plehn, M. Russell and T. Schell, *Deep-learning Top Taggers or The End of QCD?*, *JHEP* **05** (2017) 006 [arXiv:1701.08784] [INSPIRE].

[12] G. Louppe, K. Cho, C. Becot and K. Cranmer, *QCD-Aware Recursive Neural Networks for Jet Physics*, *JHEP* **01** (2019) 057 [arXiv:1702.00748] [INSPIRE].

[13] T. Cheng, *Recursive Neural Networks in Quark/Gluon Tagging*, *Comput. Softw. Big Sci.* **2** (2018) 3 [arXiv:1711.02633] [INSPIRE].

[14] S. Egan, W. Fedorko, A. Lister, J. Pearkes and C. Gay, *Long Short-Term Memory (LSTM) networks with jet constituents for boosted top tagging at the LHC*, arXiv:1711.09059 [INSPIRE].

[15] E.M. Metodiev, B. Nachman and J. Thaler, *Classification without labels: Learning from mixed samples in high energy physics*, *JHEP* **10** (2017) 174 [arXiv:1708.02949] [INSPIRE].

[16] P.T. Komiske, E.M. Metodiev and J. Thaler, *Energy Flow Networks: Deep Sets for Particle Jets*, *JHEP* **01** (2019) 121 [arXiv:1810.05165] [INSPIRE].

[17] S. Macaluso and D. Shih, *Pulling Out All the Tops with Computer Vision and Deep Learning*, *JHEP* **10** (2018) 121 [arXiv:1803.00107] [INSPIRE].

[18] A. Andreassen, I. Feige, C. Frye and M.D. Schwartz, *JUNIPR: a Framework for Unsupervised Machine Learning in Particle Physics*, *Eur. Phys. J. C* **79** (2019) 102 [arXiv:1804.09720] [INSPIRE].

[19] S.H. Lim and M.M. Nojiri, *Spectral Analysis of Jet Substructure with Neural Networks: Boosted Higgs Case*, *JHEP* **10** (2018) 181 [arXiv:1807.03312] [INSPIRE].

[20] H. Qu and L. Gouskos, *ParticleNet: Jet Tagging via Particle Clouds*, *Phys. Rev. D* **101** (2020) 056019 [arXiv:1902.08570] [INSPIRE].

[21] A. Chakraborty, S.H. Lim and M.M. Nojiri, *Interpretable deep learning for two-prong jet classification with jet spectra*, *JHEP* **19** (2020) 135 [arXiv:1904.02092] [INSPIRE].

[22] A. Andreassen, I. Feige, C. Frye and M.D. Schwartz, *Binary JUNIPR: an interpretable probabilistic model for discrimination*, *Phys. Rev. Lett.* **123** (2019) 182001 [arXiv:1906.10137] [INSPIRE].

[23] Y.-C.J. Chen, C.-W. Chiang, G. Cottin and D. Shih, *Boosted W and Z tagging with jet charge and deep learning*, *Phys. Rev. D* **101** (2020) 053001 [arXiv:1908.08256] [INSPIRE].

[24] T. Cheng, *Interpretability Study on Deep Learning for Jet Physics at the Large Hadron Collider*, in *33rd Annual Conference on Neural Information Processing Systems*, 11, 2019 [arXiv:1911.01872] [INSPIRE].

[25] K.-F. Chen and Y.-T. Chien, *Deep learning jet substructure from two-particle correlations*, *Phys. Rev. D* **101** (2020) 114025 [arXiv:1911.02020] [INSPIRE].

[26] G. Kasieczka and D. Shih, *DisCo Fever: Robust Networks Through Distance Correlation*, arXiv:2001.05310 [INSPIRE].

[27] T. Heimel, G. Kasieczka, T. Plehn and J.M. Thompson, *QCD or What?*, *SciPost Phys.* **6** (2019) 030 [arXiv:1808.08979] [INSPIRE].

[28] M. Farina, Y. Nakai and D. Shih, *Searching for New Physics with Deep Autoencoders*, *Phys. Rev. D* **101** (2020) 075021 [arXiv:1808.08992] [INSPIRE].

[29] J. Hajer, Y.-Y. Li, T. Liu and H. Wang, *Novelty Detection Meets Collider Physics*, *Phys. Rev. D* **101** (2020) 076015 [arXiv:1807.10261] [INSPIRE].

[30] B.M. Dillon, D.A. Faroughy and J.F. Kamenik, *Uncovering latent jet substructure*, *Phys. Rev. D* **100** (2019) 056002 [arXiv:1904.04200] [INSPIRE].

[31] S. Diefenbacher, H. Frost, G. Kasieczka, T. Plehn and J.M. Thompson, *CapsNets Continuing the Convolutional Quest*, *SciPost Phys.* **8** (2020) 023 [arXiv:1906.11265] [INSPIRE].

[32] A. Blance, M. Spannowsky and P. Waite, *Adversarially-trained autoencoders for robust unsupervised new physics searches*, *JHEP* **10** (2019) 047 [arXiv:1905.10384] [INSPIRE].

[33] T.S. Roy and A.H. Vijay, *A robust anomaly finder based on autoencoder*, arXiv:1903.02032 [INSPIRE].

[34] J.H. Collins, K. Howe and B. Nachman, *Extending the search for new resonances with machine learning*, *Phys. Rev. D* **99** (2019) 014038 [arXiv:1902.02634] [INSPIRE].

[35] O. Amram and C.M. Suarez, *Tag N' Train: A Technique to Train Improved Classifiers on Unlabeled Data*, arXiv:2002.12376 [INSPIRE].

[36] B. Nachman and D. Shih, *Anomaly Detection with Density Estimation*, *Phys. Rev. D* **101** (2020) 075042 [arXiv:2001.04990] [INSPIRE].

[37] A. Andreassen, B. Nachman and D. Shih, *Simulation Assisted Likelihood-free Anomaly Detection*, *Phys. Rev. D* **101** (2020) 095004 [arXiv:2001.05001] [INSPIRE].

[38] D. Guest, J. Collado, P. Baldi, S.-C. Hsu, G. Urban and D. Whiteson, *Jet Flavor Classification in High-Energy Physics with Deep Neural Networks*, *Phys. Rev. D* **94** (2016) 112002 [arXiv:1607.08633] [INSPIRE].

[39] ATLAS collaboration, *Identification of Jets Containing b-Hadrons with Recurrent Neural Networks at the ATLAS Experiment*, Tech. Rep. ATL-PHYS-PUB-2017-003 (2017).

[40] CMS collaboration, *Heavy flavor identification at CMS with deep neural networks*, Tech. Rep. CMS-DP-2017-005 (2017).

[41] J. Arjona Martínez, O. Cerri, M. Pierini, M. Spiropulu and J.-R. Vlimant, *Pileup mitigation at the Large Hadron Collider with graph neural networks*, *Eur. Phys. J. Plus* **134** (2019) 333 [arXiv:1810.07988] [INSPIRE].

[42] P.T. Komiske, E.M. Metodiev, B. Nachman and M.D. Schwartz, *Pileup Mitigation with Machine Learning (PUMML)*, *JHEP* **12** (2017) 051 [arXiv:1707.08600] [INSPIRE].

[43] P.T. Komiske, E.M. Metodiev, B. Nachman and M.D. Schwartz, *Learning to Remove Pileup at the LHC with Jet Images*, *J. Phys. Conf. Ser.* **1085** (2018) 042010 [INSPIRE].

[44] J. Bendavid, *Efficient Monte Carlo Integration Using Boosted Decision Trees and Generative Deep Neural Networks*, arXiv:1707.00028 [INSPIRE].

[45] M.D. Klimek and M. Perelstein, *Neural Network-Based Approach to Phase Space Integration*, arXiv:1810.11509 [INSPIRE].

[46] S. Otten et al., *Event Generation and Statistical Sampling for Physics with Deep Generative Models and a Density Information Buffer*, arXiv:1901.00875 [INSPIRE].

[47] B. Hashemi, N. Amin, K. Datta, D. Olivito and M. Pierini, *LHC analysis-specific datasets with Generative Adversarial Networks*, arXiv:1901.05282 [INSPIRE].

[48] R. Di Sipio, M. Faucci Giannelli, S. Ketabchi Haghighat and S. Palazzo, *DijetGAN: A Generative-Adversarial Network Approach for the Simulation of QCD Dijet Events at the LHC*, *JHEP* **08** (2019) 110 [arXiv:1903.02433] [INSPIRE].

[49] A. Butter, T. Plehn and R. Winterhalder, *How to GAN LHC Events*, *SciPost Phys.* **7** (2019) 075 [arXiv:1907.03764] [INSPIRE].

[50] S. Carrazza and F.A. Dreyer, *Lund jet images from generative and cycle-consistent adversarial networks*, *Eur. Phys. J. C* **79** (2019) 979 [arXiv:1909.01359] [INSPIRE].

[51] SHiP collaboration, *Fast simulation of muons produced at the SHiP experiment using Generative Adversarial Networks*, 2019 *JINST* **14** P11028 [arXiv:1909.04451] [INSPIRE].

[52] A. Butter, T. Plehn and R. Winterhalder, *How to GAN Event Subtraction*, arXiv:1912.08824 [INSPIRE].

[53] F. Bishara and M. Montull, *(Machine) Learning Amplitudes for Faster Event Generation*, arXiv:1912.11055 [INSPIRE].

[54] E. Bothmann, T. Janßen, M. Knobbe, T. Schmale and S. Schumann, *Exploring phase space with Neural Importance Sampling*, *SciPost Phys.* **8** (2020) 069 [arXiv:2001.05478] [INSPIRE].

[55] C. Gao, J. Isaacson and C. Krause, *i-flow: High-Dimensional Integration and Sampling with Normalizing Flows*, arXiv:2001.05486 [INSPIRE].

[56] C. Gao, S. Höche, J. Isaacson, C. Krause and H. Schulz, *Event Generation with Normalizing Flows*, *Phys. Rev. D* **101** (2020) 076002 [arXiv:2001.10028] [INSPIRE].

[57] K.T. Matchev and P. Shyamsundar, *Uncertainties associated with GAN-generated datasets in high energy physics*, arXiv:2002.06307 [INSPIRE].

[58] S. Badger and J. Bullock, *Using neural networks for efficient evaluation of high multiplicity scattering amplitudes*, *JHEP* **06** (2020) 114 [arXiv:2002.07516] [INSPIRE].

[59] A. Andreassen, P.T. Komiske, E.M. Metodiev, B. Nachman and J. Thaler, *OmniFold: A Method to Simultaneously Unfold All Observables*, *Phys. Rev. Lett.* **124** (2020) 182001 [arXiv:1911.09107] [INSPIRE].

[60] M. Bellagente, A. Butter, G. Kasieczka, T. Plehn and R. Winterhalder, *How to GAN away Detector Effects*, *SciPost Phys.* **8** (2020) 070 [arXiv:1912.00477] [INSPIRE].

[61] S. Forte, L. Garrido, J.I. Latorre and A. Piccione, *Neural network parametrization of deep inelastic structure functions*, *JHEP* **05** (2002) 062 [hep-ph/0204232] [INSPIRE].

[62] S. Forte, J.I. Latorre, L. Magnea and A. Piccione, *Determination of $\alpha_s$ from scaling violations of truncated moments of structure functions*, *Nucl. Phys. B* **643** (2002) 477 [hep-ph/0205286] [INSPIRE].

[63] J. Rojo and J.I. Latorre, *Neural network parametrization of spectral functions from hadronic tau decays and determination of QCD vacuum condensates*, *JHEP* **01** (2004) 055 [hep-ph/0401047] [INSPIRE].

[64] NNPDF collaboration, *Unbiased determination of the proton structure function $F_2^p$ with faithful uncertainty estimation*, *JHEP* **03** (2005) 080 [hep-ph/0501067] [INSPIRE].

[65] NNPDF collaboration, *Neural network determination of parton distributions: The Nonsinglet case*, *JHEP* **03** (2007) 039 [hep-ph/0701127] [INSPIRE].

[66] NNPDF collaboration, *A Determination of parton distributions with faithful uncertainty estimation*, *Nucl. Phys. B* **809** (2009) 1 [*Erratum ibid.* **816** (2009) 293] [arXiv:0808.1231] [INSPIRE].

[67] NNPDF collaboration, *Precision determination of electroweak parameters and the strange content of the proton from neutrino deep-inelastic scattering*, *Nucl. Phys. B* **823** (2009) 195 [arXiv:0906.1958] [INSPIRE].

[68] R.D. Ball et al., *A first unbiased global NLO determination of parton distributions and their uncertainties*, *Nucl. Phys. B* **838** (2010) 136 [arXiv:1002.4407] [INSPIRE].

[69] NNPDF collaboration, *Reweighting NNPDFs: the W lepton asymmetry*, *Nucl. Phys. B* **849** (2011) 112 [*Erratum ibid.* **854** (2012) 926] [*Erratum ibid.* **855** (2012) 927] [arXiv:1012.0836] [INSPIRE].

[70] R.D. Ball et al., *Impact of Heavy Quark Masses on Parton Distributions and LHC Phenomenology*, *Nucl. Phys. B* **849** (2011) 296 [arXiv:1101.1300] [INSPIRE].

[71] S. Lionetti et al., *Precision determination of $\alpha_s$ using an unbiased global NLO parton set*, *Phys. Lett. B* **701** (2011) 346 [arXiv:1103.2369] [INSPIRE].

[72] R.D. Ball et al., *Parton distributions with LHC data*, *Nucl. Phys. B* **867** (2013) 244 [arXiv:1207.1303] [INSPIRE].

[73] NNPDF collaboration, *Parton distributions with QED corrections*, *Nucl. Phys. B* **877** (2013) 290 [arXiv:1308.0598] [INSPIRE].

[74] NNPDF collaboration, *Towards the determination of the photon parton distribution function constrained by LHC data*, *PoS* **DIS2013** (2013) 279 [arXiv:1307.1131] [INSPIRE].

[75] NNPDF collaboration, *Towards an unbiased determination of parton distributions with QED corrections*, in *48th Rencontres de Moriond on QCD and High Energy Interactions*, pp. 357–360 (2013) [arXiv:1305.4179] [INSPIRE].

[76] NNPDF collaboration, *Parton distributions for the LHC Run II*, *JHEP* **04** (2015) 040 [arXiv:1410.8849] [INSPIRE].

[77] NNPDF collaboration, *Parton distributions from high-precision collider data*, *Eur. Phys. J. C* **77** (2017) 663 [arXiv:1706.00428] [INSPIRE].

[78] NNPDF collaboration, *A determination of the fragmentation functions of pions, kaons and protons with faithful uncertainties*, *Eur. Phys. J. C* **77** (2017) 516 [arXiv:1706.07049] [INSPIRE].

[79] A. Butter et al., *The Machine Learning Landscape of Top Taggers*, *SciPost Phys.* **7** (2019) 014 [arXiv:1902.09914] [INSPIRE].

[80] ATLAS collaboration, *Performance of top-quark and W-boson tagging with ATLAS in Run 2 of the LHC*, *Eur. Phys. J. C* **79** (2019) 375 [arXiv:1808.07858] [INSPIRE].

[81] J.M. Butterworth, A.R. Davison, M. Rubin and G.P. Salam, *Jet substructure as a new Higgs search channel at the LHC*, *Phys. Rev. Lett.* **100** (2008) 242001 [arXiv:0802.2470] [INSPIRE].

[82] J. Thaler and L.-T. Wang, *Strategies to Identify Boosted Tops*, *JHEP* **07** (2008) 092 [arXiv:0806.0023] [INSPIRE].

[83] D.E. Kaplan, K. Rehermann, M.D. Schwartz and B. Tweedie, *Top Tagging: A Method for Identifying Boosted Hadronically Decaying Top Quarks*, *Phys. Rev. Lett.* **101** (2008) 142001 [arXiv:0806.0848] [INSPIRE].

[84] S.D. Ellis, C.K. Vermilion and J.R. Walsh, *Techniques for improved heavy particle searches with jet substructure*, *Phys. Rev. D* **80** (2009) 051501 [arXiv:0903.5081] [INSPIRE].

[85] CMS collaboration, *A Cambridge-Aachen (C-A) based Jet Algorithm for boosted top-jet tagging*, Tech. Rep. CMS-PAS-JME-09-001 (2009).

[86] T. Plehn, G.P. Salam and M. Spannowsky, *Fat Jets for a Light Higgs*, *Phys. Rev. Lett.* **104** (2010) 111801 [arXiv:0910.5472] [INSPIRE].

[87] T. Plehn, M. Spannowsky, M. Takeuchi and D. Zerwas, *Stop Reconstruction with Tagged Tops*, *JHEP* **10** (2010) 078 [arXiv:1006.2833] [INSPIRE].

[88] M. Dasgupta, A. Fregoso, S. Marzani and G.P. Salam, *Towards an understanding of jet substructure*, *JHEP* **09** (2013) 029 [arXiv:1307.0007] [INSPIRE].

[89] A.J. Larkoski, S. Marzani, G. Soyez and J. Thaler, *Soft Drop*, *JHEP* **05** (2014) 146 [arXiv:1402.2657] [INSPIRE].

[90] D. Krohn, J. Thaler and L.-T. Wang, *Jet Trimming*, *JHEP* **02** (2010) 084 [arXiv:0912.1342] [INSPIRE].

[91] D.E. Soper and M. Spannowsky, *Finding physics signals with shower deconstruction*, *Phys. Rev. D* **84** (2011) 074002 [arXiv:1102.3480] [INSPIRE].

[92] D.E. Soper and M. Spannowsky, *Finding top quarks with shower deconstruction*, *Phys. Rev. D* **87** (2013) 054012 [arXiv:1211.3140] [INSPIRE].

[93] D.E. Soper and M. Spannowsky, *Finding physics signals with event deconstruction*, *Phys. Rev. D* **89** (2014) 094005 [arXiv:1402.1189] [INSPIRE].

[94] F.V. Tkachov, *Measuring multi-jet structure of hadronic energy flow or What is a jet?*, *Int. J. Mod. Phys. A* **12** (1997) 5411 [hep-ph/9601308] [INSPIRE].

[95] J. Thaler and K. Van Tilburg, *Identifying Boosted Objects with N-subjettiness*, *JHEP* **03** (2011) 015 [arXiv:1011.2268] [INSPIRE].

[96] M. Jankowiak and A.J. Larkoski, *Jet Substructure Without Trees*, *JHEP* **06** (2011) 057 [arXiv:1104.1646] [INSPIRE].

[97] M. Jankowiak and A.J. Larkoski, *Angular Scaling in Jets*, *JHEP* **04** (2012) 039 [arXiv:1201.2688] [INSPIRE].

[98] A.J. Larkoski, *QCD Analysis of the Scale-Invariance of Jets*, *Phys. Rev. D* **86** (2012) 054004 [arXiv:1207.1437] [INSPIRE].

[99] J. Gallicchio and M.D. Schwartz, *Quark and Gluon Jet Substructure*, *JHEP* **04** (2013) 090 [arXiv:1211.7038] [INSPIRE].

[100] A.J. Larkoski, G.P. Salam and J. Thaler, *Energy Correlation Functions for Jet Substructure*, *JHEP* **06** (2013) 108 [arXiv:1305.0007] [INSPIRE].

[101] A.J. Larkoski, I. Moult and D. Neill, *Power Counting to Better Jet Observables*, *JHEP* **12** (2014) 009 [arXiv:1409.6298] [INSPIRE].

[102] I. Moult, L. Necib and J. Thaler, *New Angles on Energy Correlation Functions*, *JHEP* **12** (2016) 153 [arXiv:1609.07483] [INSPIRE].

[103] P.T. Komiske, E.M. Metodiev and J. Thaler, *Energy flow polynomials: A complete linear basis for jet substructure*, *JHEP* **04** (2018) 013 [arXiv:1712.07124] [INSPIRE].

[104] H. Chen, M.-X. Luo, I. Moult, T.-Z. Yang, X. Zhang and H.X. Zhu, *Three Point Energy Correlators in the Collinear Limit: Symmetries, Dualities and Analytic Results*, arXiv:1912.11050 [INSPIRE].

[105] C. Frye, A.J. Larkoski, J. Thaler and K. Zhou, *Casimir Meets Poisson: Improved Quark/Gluon Discrimination with Counting Observables*, *JHEP* **09** (2017) 083 [arXiv:1704.06266] [INSPIRE].

[106] J. Gallicchio and M.D. Schwartz, *Seeing in Color: Jet Superstructure*, *Phys. Rev. Lett.* **105** (2010) 022001 [arXiv:1001.5027] [INSPIRE].

[107] J. Gallicchio, J. Huth, M. Kagan, M.D. Schwartz, K. Black and B. Tweedie, *Multivariate discrimination and the Higgs + $W/Z$ search*, *JHEP* **04** (2011) 069 [arXiv:1010.3698] [INSPIRE].

[108] A. Hook, M. Jankowiak and J.G. Wacker, *Jet Dipolarity: Top Tagging with Color Flow*, *JHEP* **04** (2012) 007 [arXiv:1102.1012] [INSPIRE].

[109] S. Choi, S.J. Lee and M. Perelstein, *Infrared Safety of a Neural-Net Top Tagging Algorithm*, *JHEP* **02** (2019) 132 [arXiv:1806.01263] [INSPIRE].

[110] S. Bollweg, M. Haußmann, G. Kasieczka, M. Luchmann, T. Plehn and J. Thompson, *Deep-Learning Jets with Uncertainties and More*, *SciPost Phys.* **8** (2020) 006 [arXiv:1904.10004] [INSPIRE].

[111] G. Kasieczka, M. Luchmann, F. Otterpohl and T. Plehn, *Per-Object Systematics using Deep-Learned Calibration*, arXiv:2003.11099 [INSPIRE].

[112] M. Gori, G. Monfardini and F. Scarselli, *A new model for learning in graph domains*, in proceedings of *2005 IEEE International Joint Conference on Neural Networks*, vol. 2, pp. 729–734, vol. 2, July 2005 [DOI].

[113] F. Scarselli, M. Gori, A.C. Tsoi, M. Hagenbuchner and G. Monfardini, *The graph neural network model*, *IEEE Trans. Neural Networks* **20** (2009) 61.

[114] D. Raposo, A. Santoro, D.G.T. Barrett, R. Pascanu, T.P. Lillicrap and P.W. Battaglia, *Discovering objects and their relations from entangled scene representations*, *CoRR* **abs/1702.05068** (2017) [arXiv:1702.05068].

[115] A. Santoro et al., *A simple neural network module for relational reasoning*, in *Advances in Neural Information Processing Systems 30*, I. Guyon et al. eds., pp. 4967–4976m Curran Associates, Inc. (2017).

[116] P.W. Battaglia et al., *Relational inductive biases, deep learning, and graph networks*, *CoRR* **abs/1806.01261** (2018) [arXiv:1806.01261].

[117] I. Henrion et al., *Neural message passing for jet physics*, in *Proceedings of the Deep Learning for Physical Sciences (DLPS) 2017 workshop at the 31st Annual Conference on Neural Information Processing Systems (NeurlIPS)*, 2017.

[118] S.R. Qasim, J. Kieseler, Y. Iiyama and M. Pierini, *Learning representations of irregular particle-detector geometry with distance-weighted graph networks*, *Eur. Phys. J. C* **79** (2019) 608 [arXiv:1902.07987] [INSPIRE].

[119] M. Abdughani, J. Ren, L. Wu and J.M. Yang, *Probing stop pair production at the LHC with graph neural networks*, *JHEP* **08** (2019) 055 [arXiv:1807.09088] [INSPIRE].

[120] E.A. Moreno et al., *JEDI-net: a jet identification algorithm based on interaction networks*, *Eur. Phys. J. C* **80** (2020) 58 [arXiv:1908.05318] [INSPIRE].

[121] E.A. Moreno et al., *Interaction networks for the identification of boosted $H \to b\bar{b}$ decays*, arXiv:1909.12285 [INSPIRE].

[122] J. Ren, L. Wu and J.M. Yang, *Unveiling CP property of top-Higgs coupling with graph neural networks at the LHC*, *Phys. Lett. B* **802** (2020) 135198 [arXiv:1901.05627] [INSPIRE].

[123] V. Mikuni and F. Canelli, *ABCNet: An attention-based method for particle tagging*, *Eur. Phys. J. Plus* **135** (2020) 463 [arXiv:2001.05311] [INSPIRE].

[124] T. Bister, M. Erdmann, J. Glombitza, N. Langner, J. Schulte and M. Wirtz, *Identification of Patterns in Cosmic-Ray Arrival Directions using Dynamic Graph Convolutional Neural Networks*, arXiv:2003.13038 [INSPIRE].

[125] C. Basham, L.S. Brown, S.D. Ellis and S.T. Love, *Energy correlations in electron-positron annihilation: testing QCD*, *Phys. Rev. Lett.* **41** (1978) 1585 [INSPIRE].

[126] C.L. Basham, L.S. Brown, S.D. Ellis and S.T. Love, *Energy Correlations in electron-Positron Annihilation in Quantum Chromodynamics: Asymptotically Free Perturbation Theory*, *Phys. Rev. D* **19** (1979) 2018 [INSPIRE].

[127] C. Basham and S.T. Love, *Energy Correlations in Electron-Positron Annihilation: Sensitivity of Quantum Chromodynamics Tests to Gluon Spin*, *Phys. Rev. D* **20** (1979) 340 [INSPIRE].

[128] C. Basham, L.S. Brown, S.D. Ellis and S.T. Love, *Energy Correlations in Perturbative Quantum Chromodynamics: A Conjecture for All Orders*, *Phys. Lett. B* **85** (1979) 297 [INSPIRE].

[129] K.R. Mecke, T. Buchert and H. Wagner, *Robust morphological measures for large scale structure in the universe*, *Astron. Astrophys.* **288** (1994) 697 [astro-ph/9312028] [INSPIRE].

[130] J. Schmalzing, M. Kerscher and T. Buchert, *Minkowski functionals in cosmology*, *Proc. Int. Sch. Phys. Fermi* **132** (1996) 281 [astro-ph/9508154] [INSPIRE].

[131] J. Schmalzing and T. Buchert, *Beyond genus statistics: A Unifying approach to the morphology of cosmic structure*, *Astrophys. J. Lett.* **482** (1997) L1 [astro-ph/9702130] [INSPIRE].

[132] J. Schmalzing and K.M. Gorski, *Minkowski functionals used in the morphological analysis of cosmic microwave background anisotropy maps*, *Mon. Not. Roy. Astron. Soc.* **297** (1998) 355 [astro-ph/9710185] [INSPIRE].

[133] S. Winitzki and A. Kosowsky, *Minkowski functional description of microwave background gaussianity*, *New Astron.* **3** (1998) 75.

[134] M. Kerscher et al., *Non-Gaussian morphology on large scales: Minkowski functionals of the reflex cluster catalogue*, *Astron. Astrophys.* **377** (2001) 1 [astro-ph/0105150] [INSPIRE].

[135] C. Beisbart, T. Buchert and H. Wagner, *Morphometry of spatial patterns*, *Physica A* **293** (2001) 592 [astro-ph/0007459] [INSPIRE].

[136] T. Matsubara, *Statistical perturbation theory of cosmic fields. 1. Basic formalism and second order theory*, astro-ph/0006269 [INSPIRE].

[137] C. Hikage, E. Komatsu and T. Matsubara, *Primordial Non-Gaussianity and Analytical Formula for Minkowski Functionals of the Cosmic Microwave Background and Large-scale Structure*, *Astrophys. J.* **653** (2006) 11 [astro-ph/0607284] [INSPIRE].

[138] D. Göring, M.A. Klatt, C. Stegmann and K. Mecke, *Morphometric analysis in gamma-ray astronomy using Minkowski functionals — Source detection via structure quantification*, *Astron. Astrophys.* **555** (2013) A38 [arXiv:1304.3732] [INSPIRE].

[139] M.A. Klatt and K. Mecke, *Morphometric analysis in gamma-ray astronomy using Minkowski functionals: II. Joint structure quantification*, arXiv:1710.03542 [INSPIRE].

[140] M.A. Klatt and K. Mecke, *Morphometric analysis in gamma-ray astronomy using Minkowski functionals: III. Sensitivity increase via a refined structure quantification*, arXiv:1710.03543 [INSPIRE].

[141] P. Chingangbam, V. Ganesan, K.P. Yogendran and C. Park, *On Minkowski Functionals of CMB polarization*, *Phys. Lett. B* **771** (2017) 67 [arXiv:1705.04454] [INSPIRE].

[142] P. Pranav et al., *Topology and Geometry of Gaussian random fields I: on Betti Numbers, Euler characteristic and Minkowski functionals*, *Mon. Not. Roy. Astron. Soc.* **485** (2019) 4167 [arXiv:1812.07310] [INSPIRE].

[143] K.R. Mecke, *Morphological characterization of patterns in reaction-diffusion systems*, *Phys. Rev. E* **53** (1996) 4794.

[144] K.R. Mecke, *Additivity, convexity, and beyond: Applications of minkowski functionals in statistical physics*, in *Statistical Physics and Spatial Statistics*, K.R. Mecke and D. Stoyan eds., Berlin, Heidelberg, pp. 111–184, Springer Berlin Heidelberg (2000).

[145] H. Mantz, K. Jacobs and K. Mecke, *Utilizing minkowski functionals for image analysis: a marching square algorithm*, *J. Stat. Mech.* **2008** (2008) P12015.

[146] J. Lin, M. Freytsis, I. Moult and B. Nachman, *Boosting $H \to b\bar{b}$ with Machine Learning*, *JHEP* **10** (2018) 101 [arXiv:1807.10768] [INSPIRE].

[147] Y.-T. Chien, *Telescoping jets: Probing hadronic event structure with multiple $R$'s*, *Phys. Rev. D* **90** (2014) 054008 [arXiv:1304.5240] [INSPIRE].

[148] Y.-T. Chien, A. Emerman, S.-C. Hsu, S. Meehan and Z. Montague, *Telescoping jet substructure*, arXiv:1711.11041 [INSPIRE].

[149] H. Hadwigeb, *Integralsätze im konvexring*, *Abh. Math. Semin. Univ. Hambg.* **20** (1956) 136.

[150] A. Cole and G. Shiu, *Persistent Homology and Non-Gaussianity*, *JCAP* **03** (2018) 025 [arXiv:1712.08159] [INSPIRE].

[151] A. Cole and G. Shiu, *Topological Data Analysis for the String Landscape*, *JHEP* **03** (2019) 054 [arXiv:1812.06960] [INSPIRE].

[152] J. Alwall et al., *The automated computation of tree-level and next-to-leading order differential cross sections and their matching to parton shower simulations*, *JHEP* **07** (2014) 079 [arXiv:1405.0301] [INSPIRE].

[153] T. Sjöstrand et al., *An introduction to PYTHIA 8.2*, *Comput. Phys. Commun.* **191** (2015) 159 [arXiv:1410.3012] [INSPIRE].

[154] J. Bellm et al., *HERWIG 7.0/HERWIG++ 3.0 release note*, *Eur. Phys. J. C* **76** (2016) 196 [arXiv:1512.01178] [INSPIRE].

[155] M. Bahr et al., *HERWIG++ Physics and Manual*, *Eur. Phys. J. C* **58** (2008) 639 [arXiv:0803.0883] [INSPIRE].

[156] DELPHES 3 collaboration, *DELPHES 3, A modular framework for fast simulation of a generic collider experiment*, *JHEP* **02** (2014) 057 [arXiv:1307.6346] [INSPIRE].

[157] D. Clevert, T. Unterthiner and S. Hochreiter, *Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs)*, in *The International Conference on Learning Representations (ICLR)*, (2016) [arXiv:1511.07289].

[158] S.J. Hanson and L.Y. Pratt, *Comparing biases for minimal network construction with back-propagation*, in *Advances in Neural Information Processing Systems 1*, D.S. Touretzky ed., pp. 177–185. Morgan-Kaufmann (1989).

[159] A.S. Weigend, D.E. Rumelhart and B.A. Huberman, *Generalization by weight-elimination with application to forecasting*, in *Advances in Neural Information Processing Systems 3*, R.P. Lippmann, J.E. Moody and D.S. Touretzky eds., pp. 875–882, Morgan-Kaufmann (1991).

[160] A. Krogh and J.A. Hertz, *A simple weight decay can improve generalization*, in *Advances in Neural Information Processing Systems 4*, J.E. Moody, S.J. Hanson and R.P. Lippmann, eds., pp. 950–957, Morgan-Kaufmann (1992).

[161] D.P. Kingma and J. Ba, *Adam: A Method for Stochastic Optimization*, arXiv:1412.6980 [INSPIRE].

[162] K. He, X. Zhang, S. Ren and J. Sun, *Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification*, arXiv:1502.01852 [INSPIRE].

[163] F. Chollet et al., *Keras*, https://keras.io (2015).

[164] M. Abadi et al., *TensorFlow: Large-scale machine learning on heterogeneous systems*, https://www.tensorflow.org/ (2015).

[165] F.A. Dreyer, L. Necib, G. Soyez and J. Thaler, *Recursive Soft Drop*, *JHEP* **06** (2018) 093 [arXiv:1804.03657] [INSPIRE].

[166] Y. Wang, Y. Sun, Z. Liu, S.E. Sarma, M.M. Bronstein and J.M. Solomon, *Dynamic Graph CNN for Learning on Point Clouds*, arXiv:1801.07829 [INSPIRE].

[167] S. Xie, R. Girshick, P. Dollar, Z. Tu and K. He, *Aggregated residual transformations for deep neural networks*, in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (2017). [arXiv:1611.05431].

[168] K. He, X. Zhang, S. Ren and J. Sun, *Deep Residual Learning for Image Recognition*, arXiv:1512.03385 [INSPIRE].

[169] S. Gieseke, P. Stephens and B. Webber, *New formalism for QCD parton showers*, *JHEP* **12** (2003) 045 [hep-ph/0310083] [INSPIRE].

[170] B.R. Webber, *A QCD Model for Jet Fragmentation Including Soft Gluon Interference*, *Nucl. Phys. B* **238** (1984) 492 [INSPIRE].

[171] T. Sjöstrand and P.Z. Skands, *Transverse-momentum-ordered showers and interleaved multiple interactions*, *Eur. Phys. J. C* **39** (2005) 129 [hep-ph/0408302] [INSPIRE].

[172] B. Andersson, G. Gustafson, G. Ingelman and T. Sjöstrand, *Parton Fragmentation and String Dynamics*, *Phys. Rept.* **97** (1983) 31 [INSPIRE].

[173] T. Sjöstrand, *Jet Fragmentation of Nearby Partons*, *Nucl. Phys. B* **248** (1984) 469 [INSPIRE].

[174] A.J. Larkoski, J. Thaler and W.J. Waalewijn, *Gaining (Mutual) Information about Quark/Gluon Discrimination*, *JHEP* **11** (2014) 129 [arXiv:1408.3122] [INSPIRE].

[175] P. Gras et al., *Systematics of quark/gluon tagging*, *JHEP* **07** (2017) 091 [arXiv:1704.03878] [INSPIRE].

[176] A. Andreassen and B. Nachman, *Neural Networks for Full Phase-space Reweighting and Parameter Tuning*, *Phys. Rev. D* **101** (2020) 091901 [arXiv:1907.08209] [INSPIRE].

[177] P. Skands, S. Carrazza and J. Rojo, *Tuning PYTHIA 8.1: the Monash 2013 Tune*, *Eur. Phys. J. C* **74** (2014) 3024 [arXiv:1404.5630] [INSPIRE].

[178] J. Bellm et al., *Minimum-bias and underlying-event tunes*, https://herwig.hepforge.org/tutorials/mpi/tunes.html (2015).

[179] S. Gieseke, C. Rohr and A. Siodmok, *Colour reconnections in HERWIG++*, *Eur. Phys. J. C* **72** (2012) 2225 [arXiv:1206.0041] [INSPIRE].

[180] M. Cacciari, G.P. Salam and G. Soyez, *The anti-$k_t$ jet clustering algorithm*, *JHEP* **04** (2008) 063 [arXiv:0802.1189] [INSPIRE].

[181] M. Cacciari, G.P. Salam and G. Soyez, *FastJet User Manual*, *Eur. Phys. J. C* **72** (2012) 1896 [arXiv:1111.6097] [INSPIRE].

[182] M. Cacciari and G.P. Salam, *Dispelling the $N^3$ myth for the $k_t$ jet-finder*, *Phys. Lett. B* **641** (2006) 57 [hep-ph/0512210] [INSPIRE].

[183] S. Catani, Y.L. Dokshitzer, M.H. Seymour and B.R. Webber, *Longitudinally invariant $K_t$ clustering algorithms for hadron hadron collisions*, *Nucl. Phys. B* **406** (1993) 187 [INSPIRE].

[184] S.D. Ellis and D.E. Soper, *Successive combination jet algorithm for hadron collisions*, *Phys. Rev. D* **48** (1993) 3160 [hep-ph/9305266] [INSPIRE].