MEL2040 PROJECT
DATA-DRIVEN ANALYSIS OF FLUID FLOWS

Fluid Mechanics

Avinash Kumar (B22ME014)

REVIEW OF MACHINE LEARNING IN FLUIDS

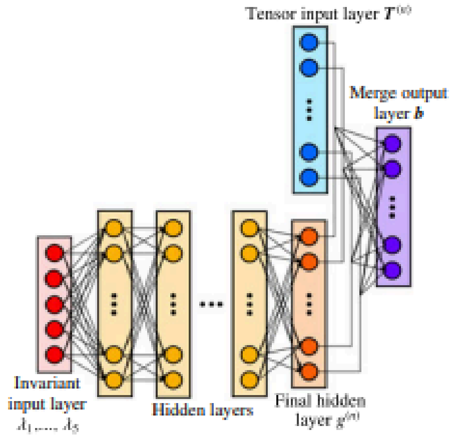
This article outlines the impact of deep neural networks (DNNs) on fluid dynamics, particularly in turbulence modeling. It emphasizes DNNs' superior accuracy in predicting complex flows and suggests improvements such as addressing potential challenges and providing clearer explanations and training methods.

Focus on Fluids journals.cambridge.org/focus

Deep learning in fluid dynamics

J. Nathan Kutz†

Department of Applied Mathematics, University of Washington, Seattle, WA 98195, USA



It was only a matter of time before deep neural networks (DNNs) – deep learning – made their mark in turbulence modelling, or more broadly, in the general area of high-dimensional, complex dynamical systems. In the last decade, DNNs have become a dominant data mining tool for big data applications. Although neural networks have been applied previously to complex fluid flows, the article featured here (Ling *et al.*, *J. Fluid Mech.*, vol. 807, 2016, pp. 155–166) is the first to apply a true DNN architecture, specifically to Reynolds averaged Navier Stokes turbulence models. As one often expects with modern DNNs, performance gains are achieved over competing state-of-the-art methods, suggesting that DNNs may play a critically enabling role in the future of modelling complex flows.

Key words: computational methods, low-dimensional models, turbulence modelling

1. Introduction

Neural networks were inspired by the Nobel prize winning work of Hubel and Wiesel on the primary visual cortex of cats (Hubel & Wiesel 1962). Their seminal experiments showed that neuronal networks where organized in hierarchical layers of cells for processing visual stimulus. The first mathematical model of a neural network, termed the Neocognitron in 1980 (Fukushima 1980), had many of the characteristic features of today's deep neural networks (DNNs), which are typically between 7–10 layers, but more recently have been scaled to hundreds of layers for certain applications. The recent success of DNNs has been enabled by two critical components: (i) the continued growth of computational power and (ii) exceptionally large labelled data sets which take advantage of the power of a multi-layer (deep) architecture. Indeed, although the theoretical inception of DNNs has an almost

† Email address for correspondence: kutz@uw.edu

NEW IDEAS

1. Turbulence Modeling using Generative Adversarial Networks (GANs): Utilize GANs to develop generative models of turbulent flow fields, addressing the limitations of classical turbulence models.

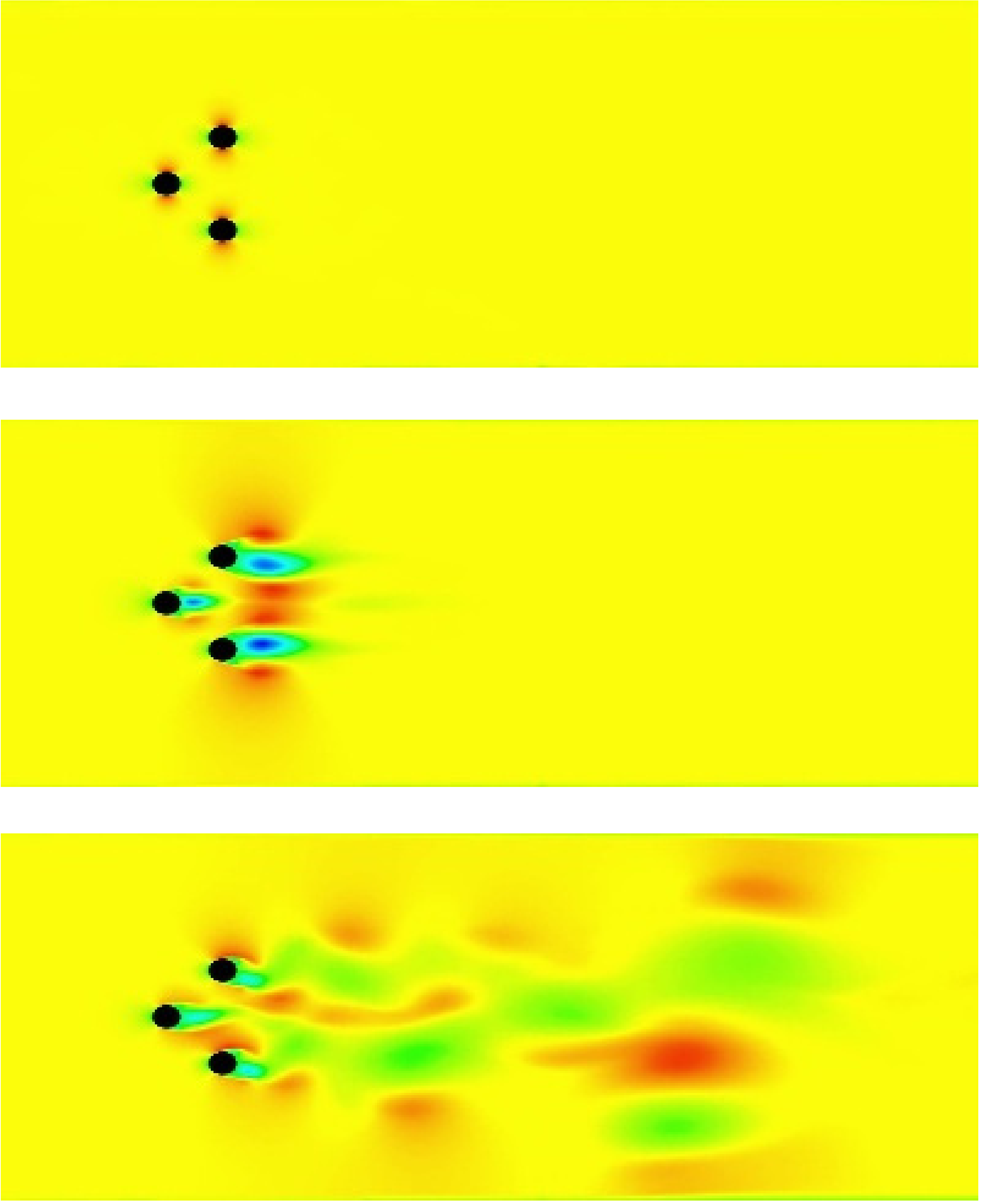
2. Physics-Informed Neural Networks (PINNs) for Fluid Flow Simulations: Employ PINNs to efficiently simulate fluid flows by learning the underlying physics from data, particularly useful for complex geometries and boundary conditions.

3. Anomaly Detection in Fluid Flow Data Using Autoencoders: Use autoencoders to detect anomalies in fluid systems, crucial for condition monitoring and fault detection, by learning a low-dimensional representation of normal flow data and identifying deviations.

4. Reduced-Order Modeling of Fluid Flows Using Variational Autoencoders (VAEs): Employ VAEs to derive compact, probabilistic representations of fluid flow data for efficient reduced-order modeling, overcoming limitations of traditional techniques like POD and DMD.

IMAGE GENRATION

- We take image with fram rate 40fps and we got around 1300 image as our dataset
- We also need to resize the image to 150x400 pixel, as there is some Memory error maybe because of large dataset



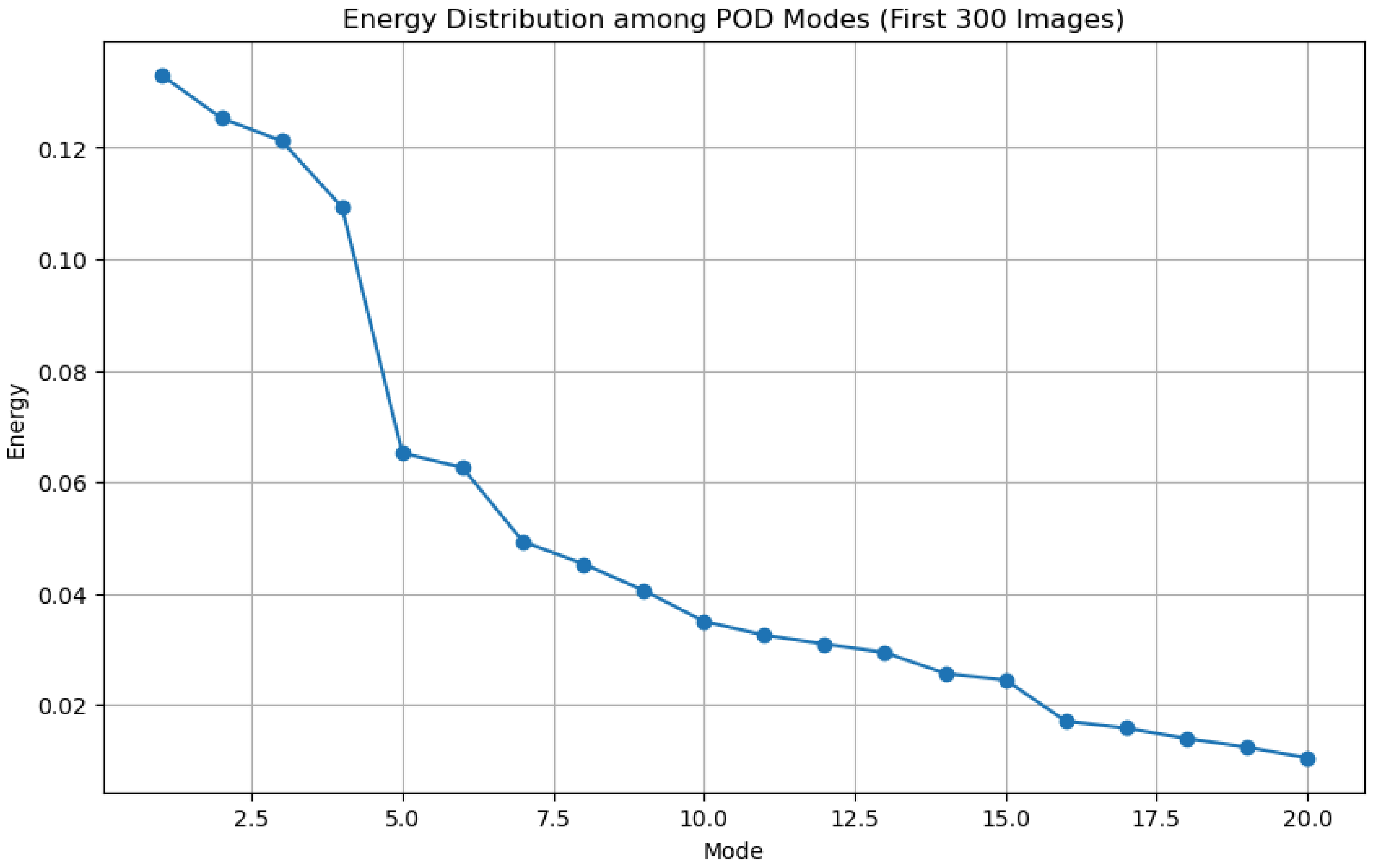
APPLYING POD

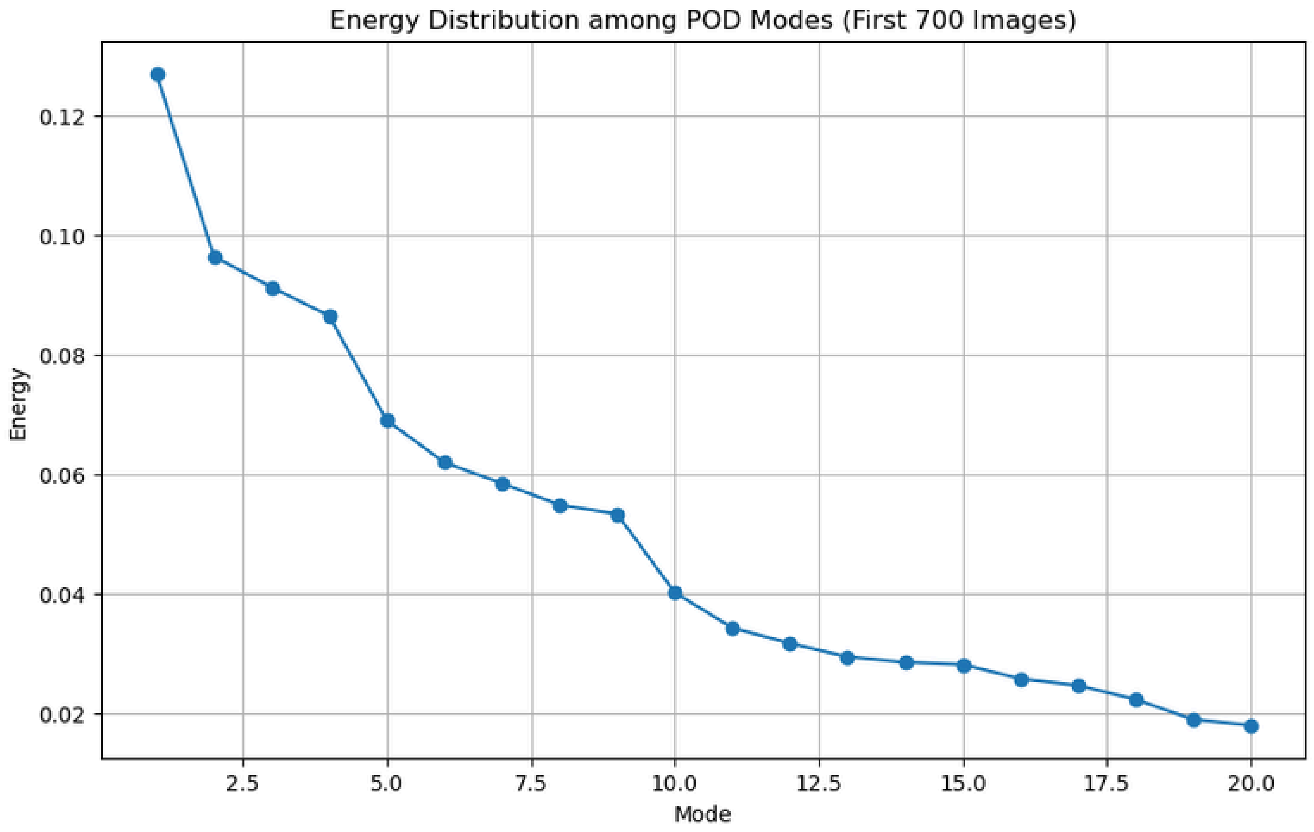
apply POD methods to all our images and taken out modes from it. modes are the concentrated information from the dataset i.e eigenvectors

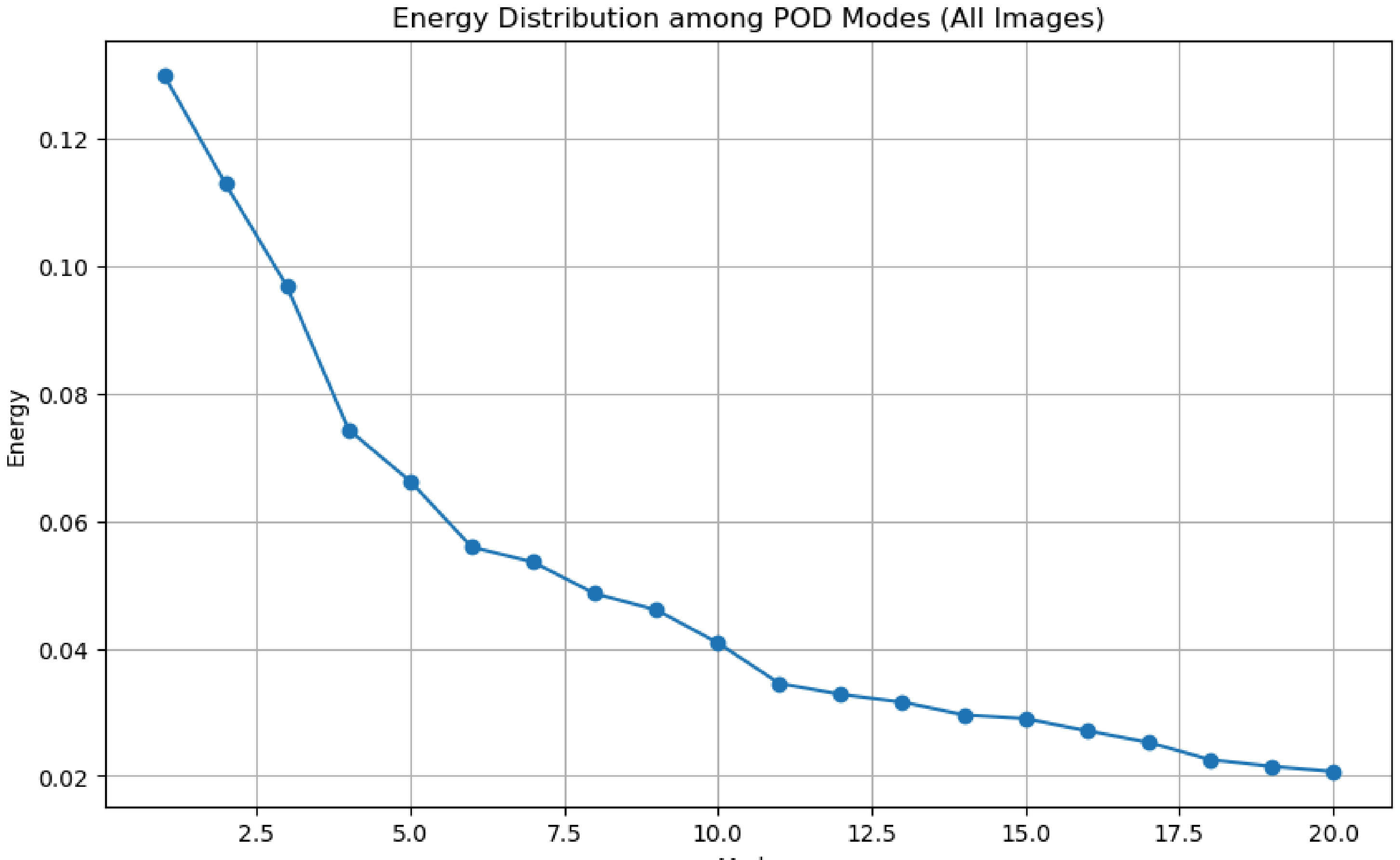
we have saved our modes in .npy format (format used to store Numpy array)

we have used SVD method for POD apllication
A = U * S * VT
U and V are orthogonal matrix and S diagonal matrix.

ANALYSIS OF THIS MODES



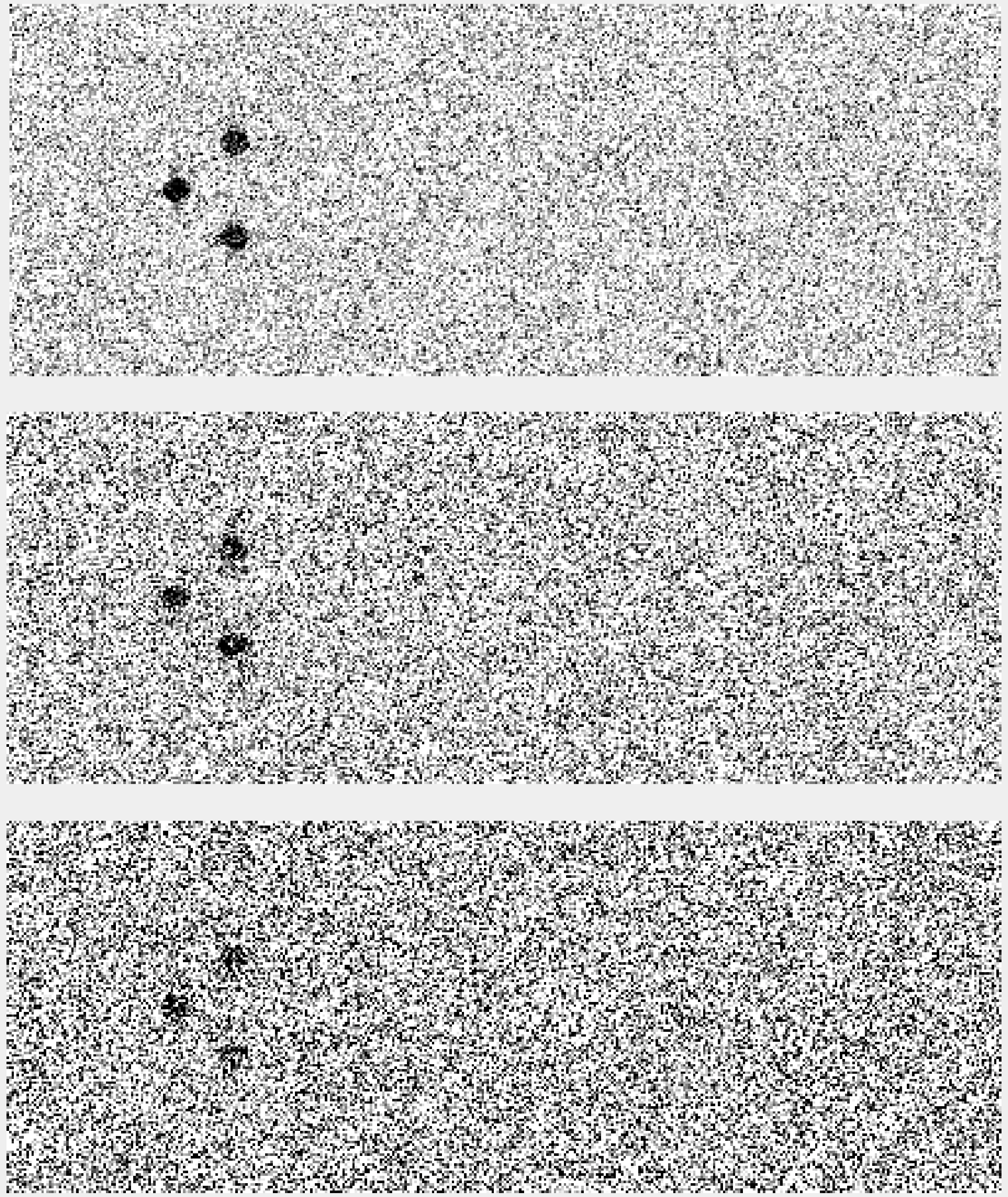




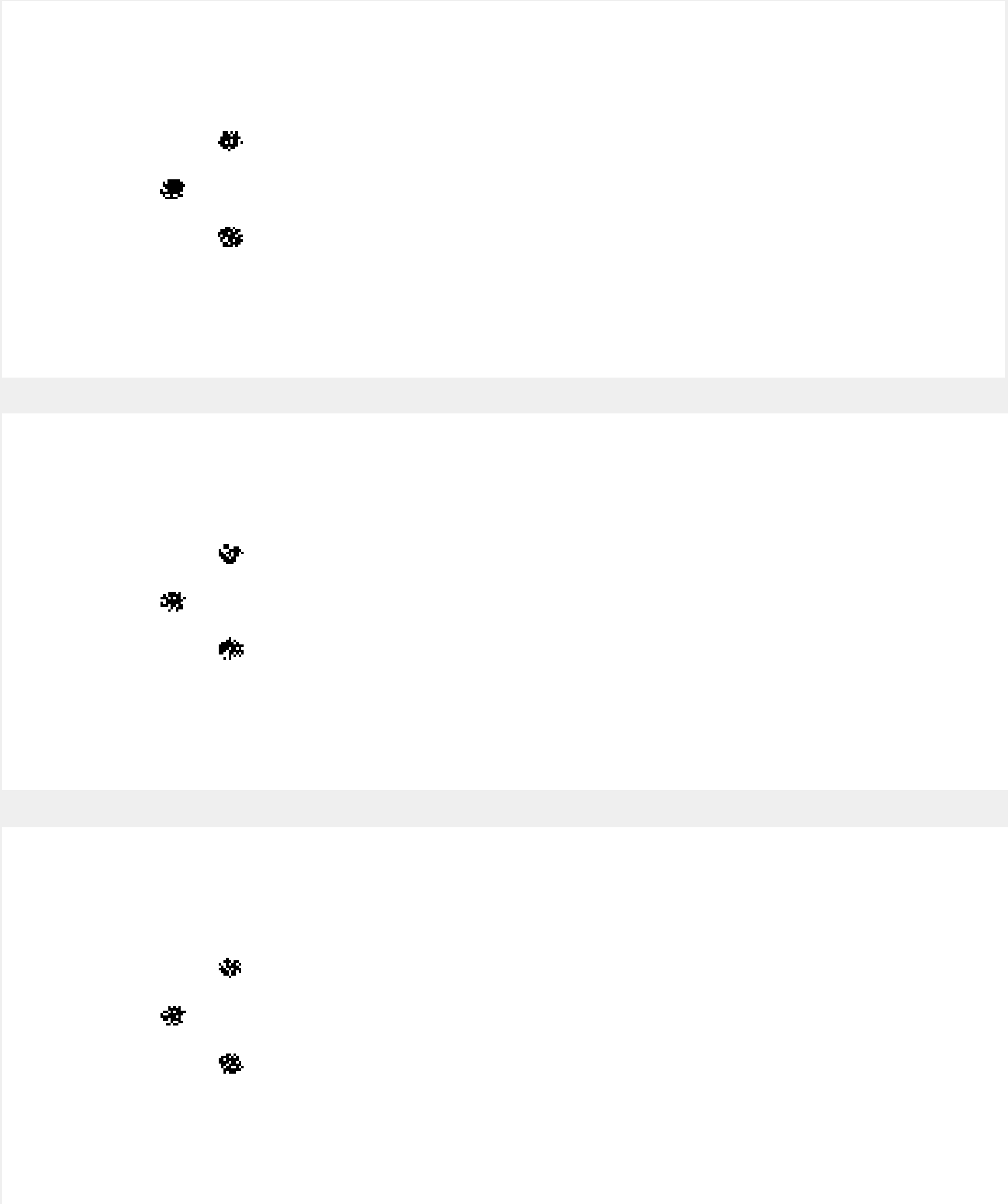
more even distributed energy

ADDING NOISE

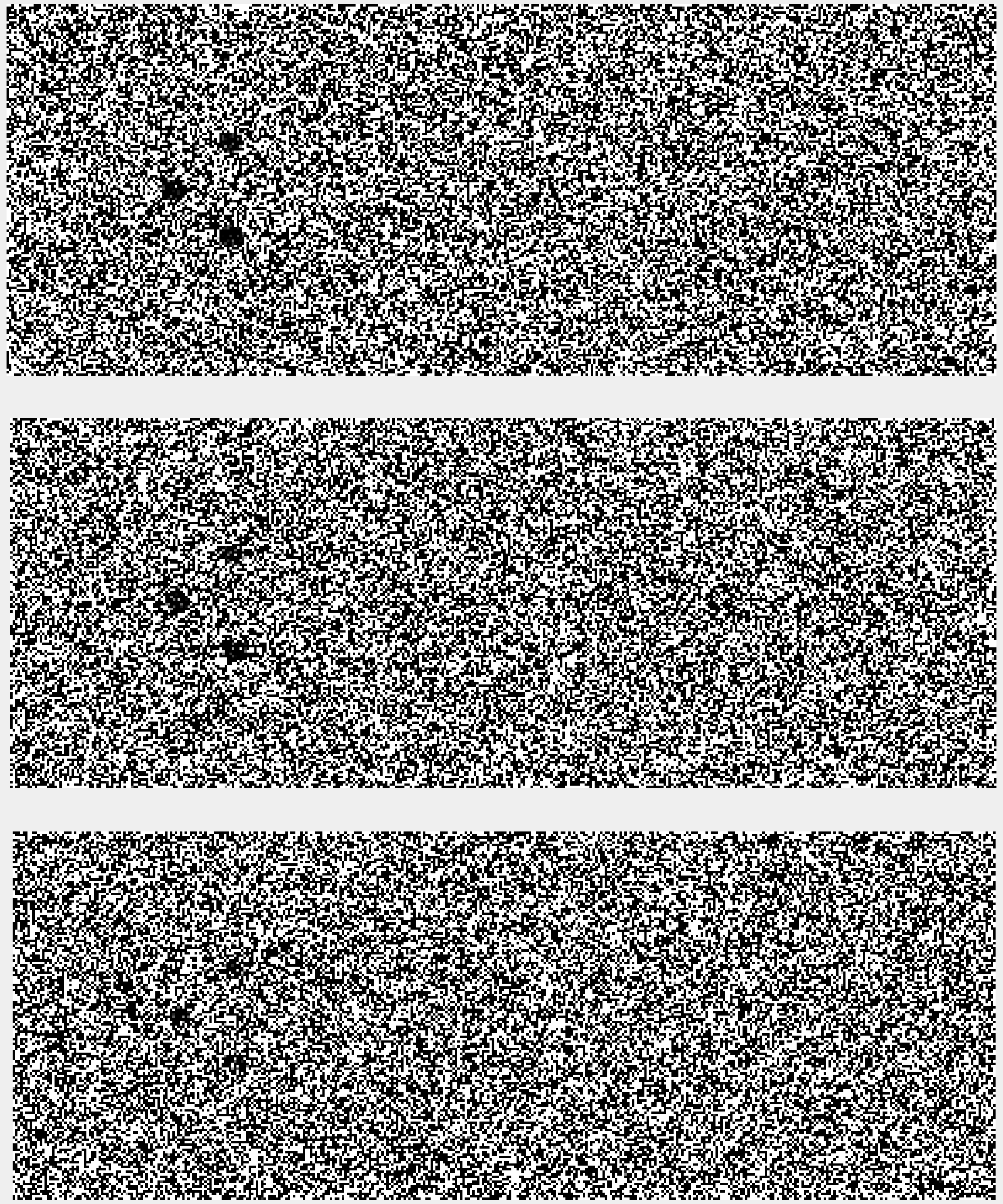
GAUSSIAN



POISSON

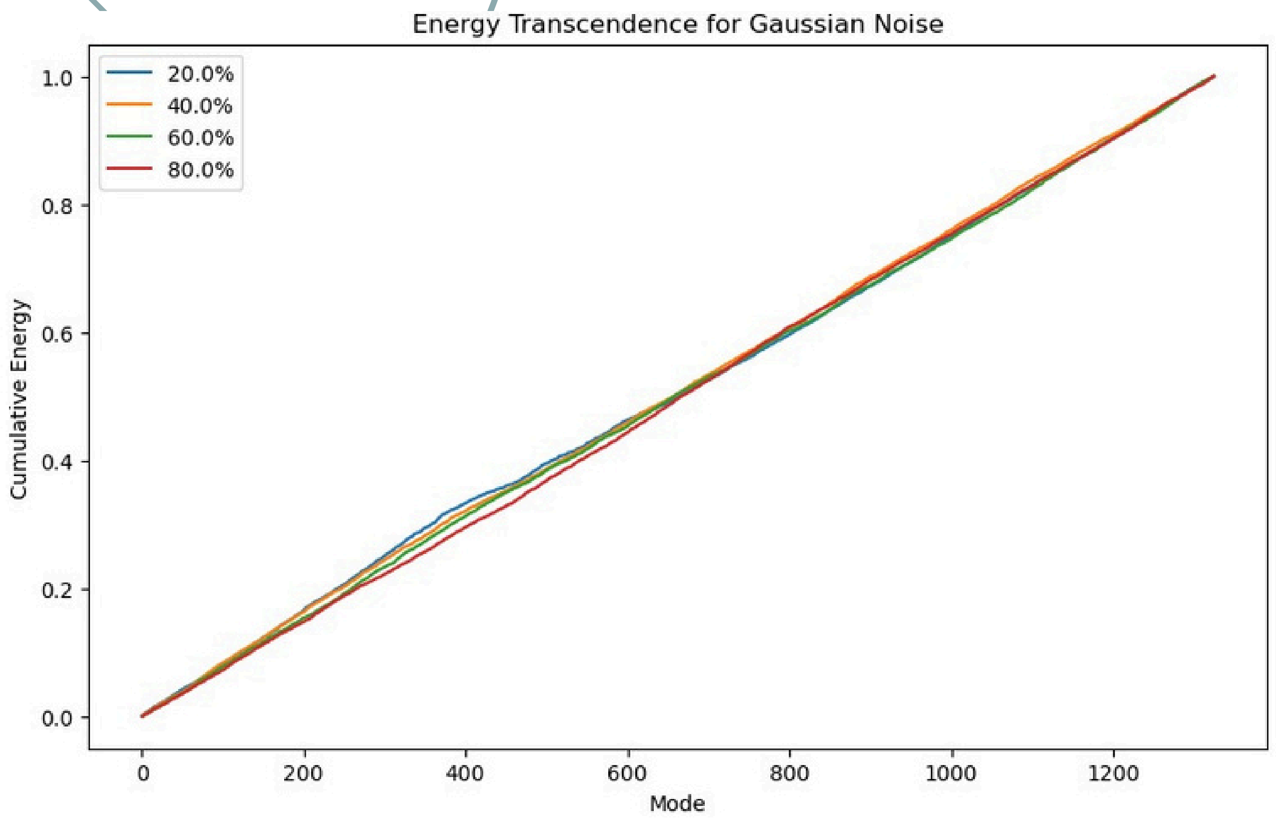


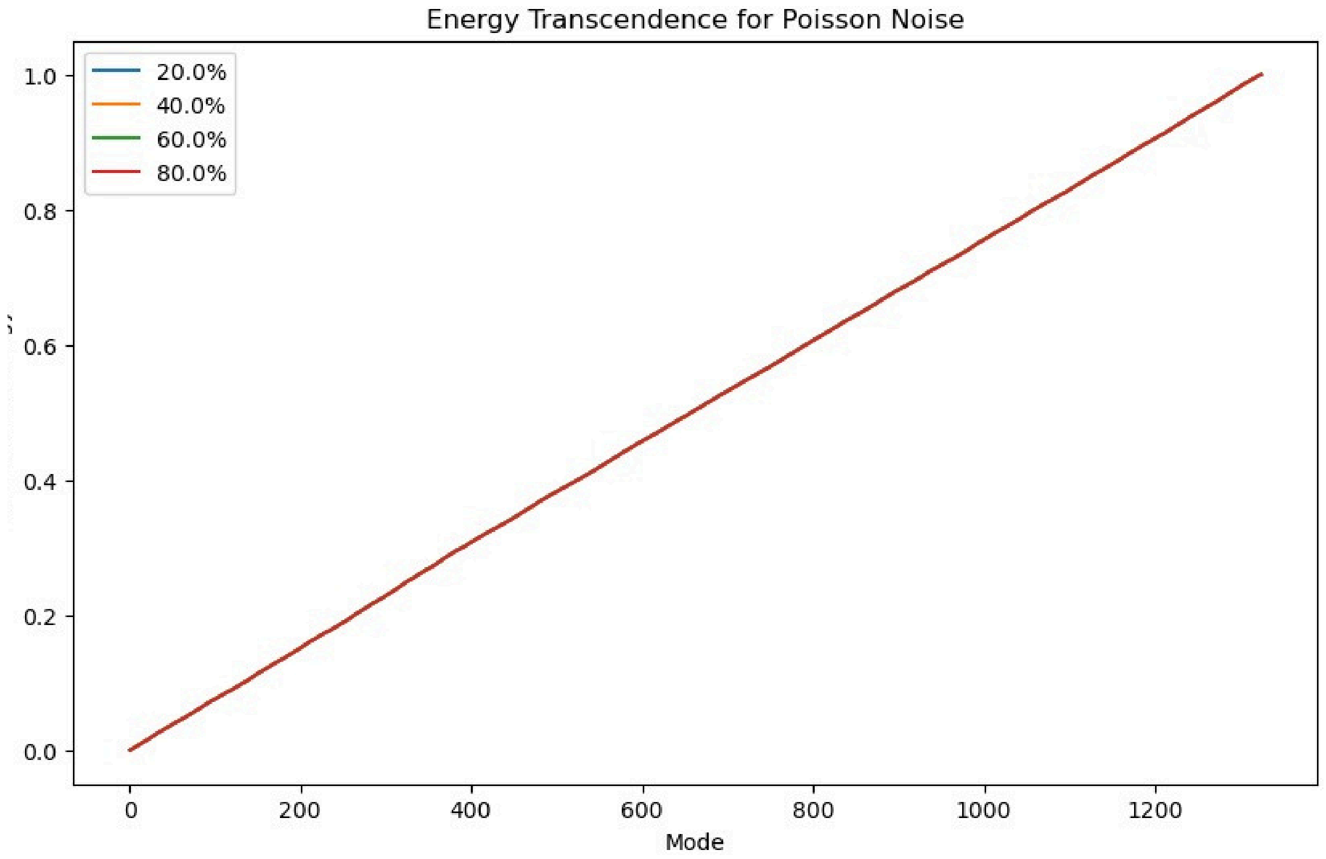
SPECKLE

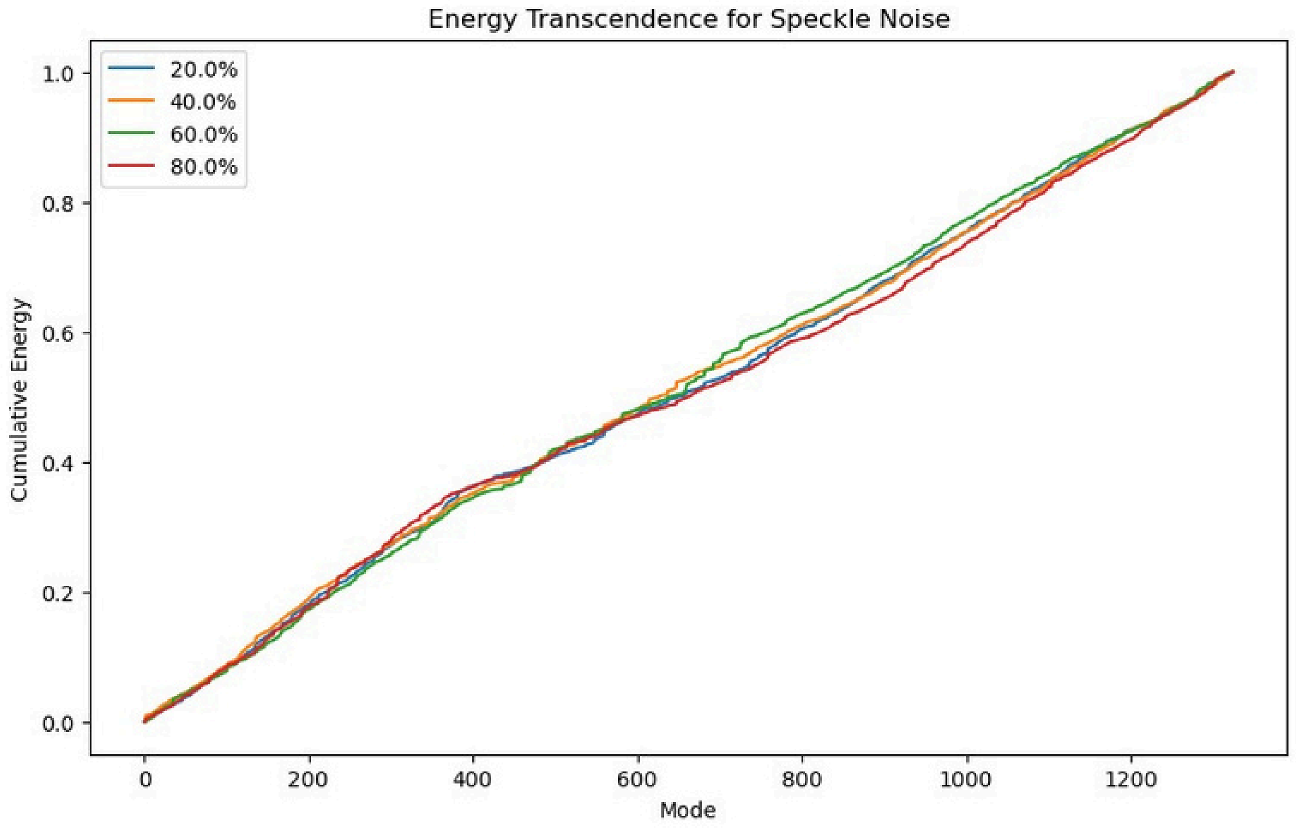


EFFECT ON POD MODES

Energy transcend through modes

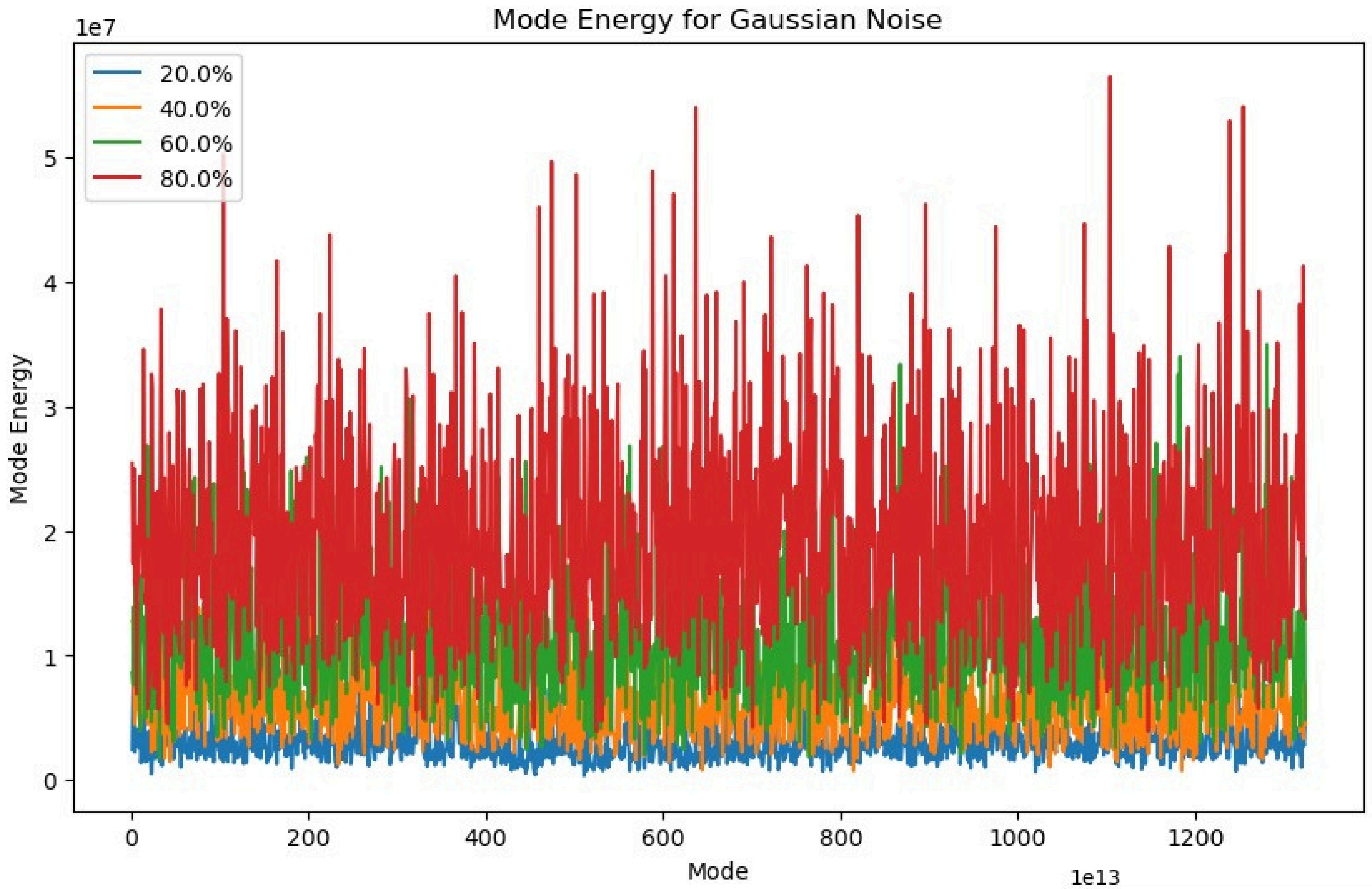


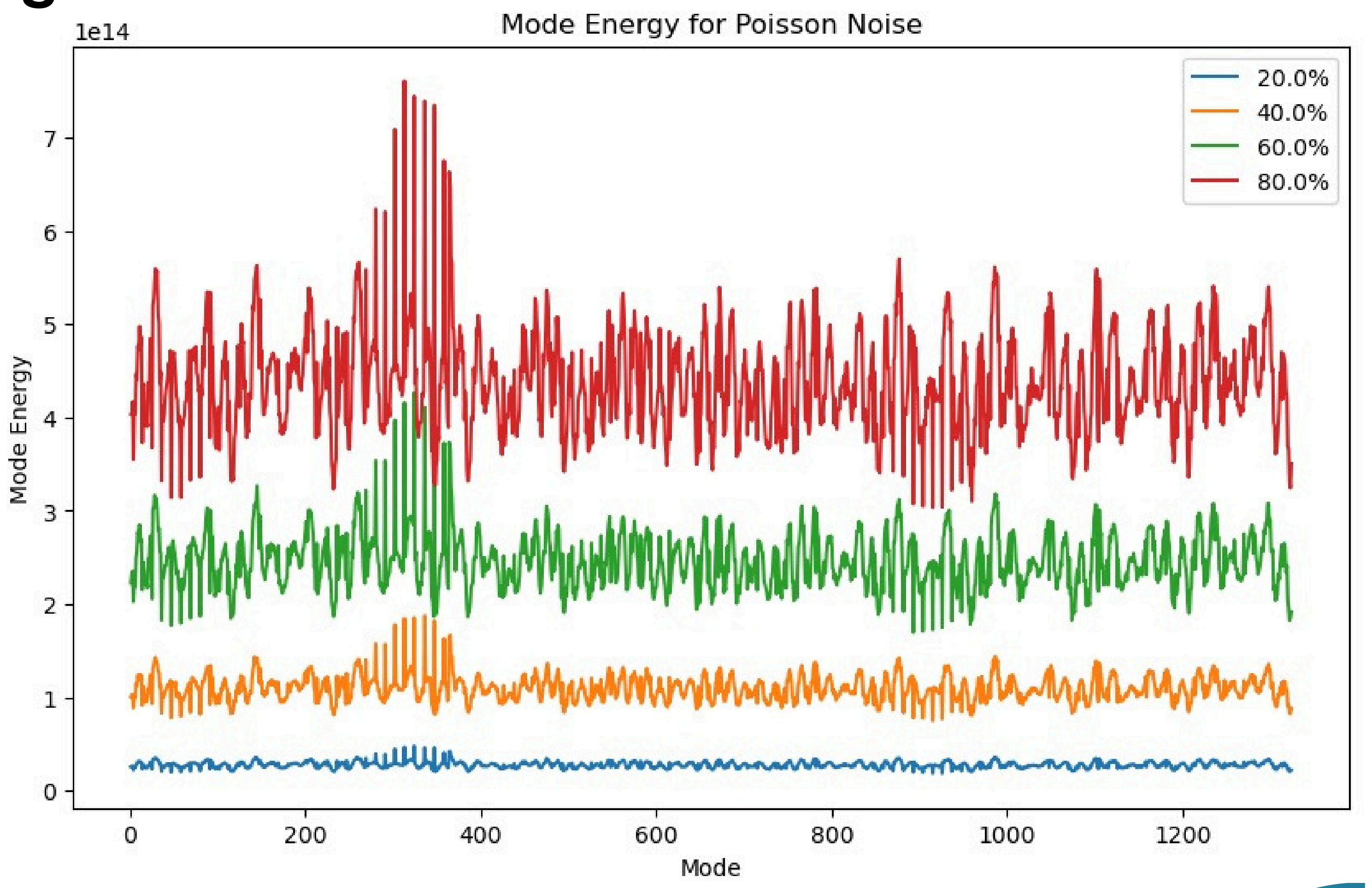


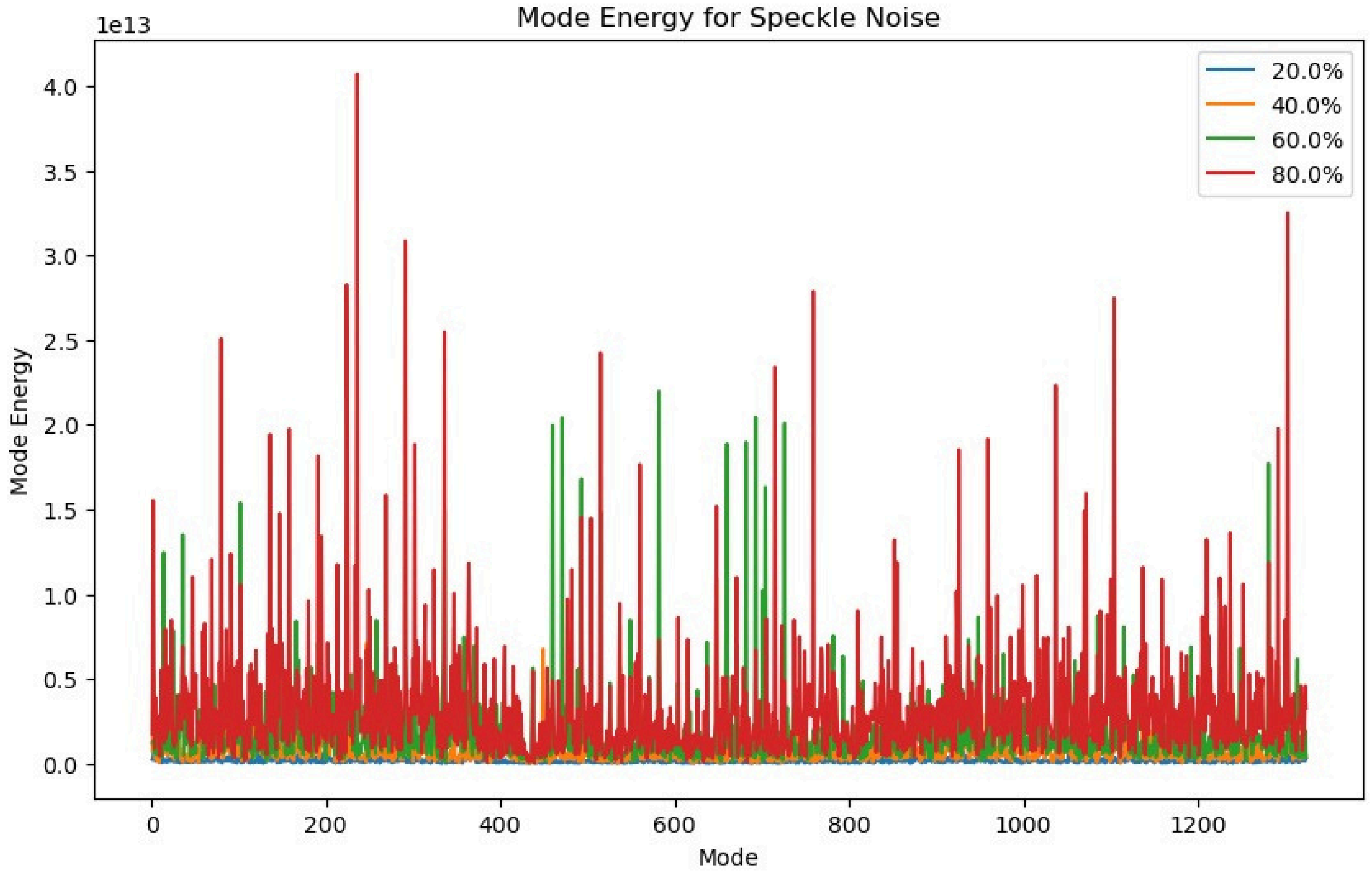


EFFECT ON POD MODES

Statistical evidence of the energy changes in the modes

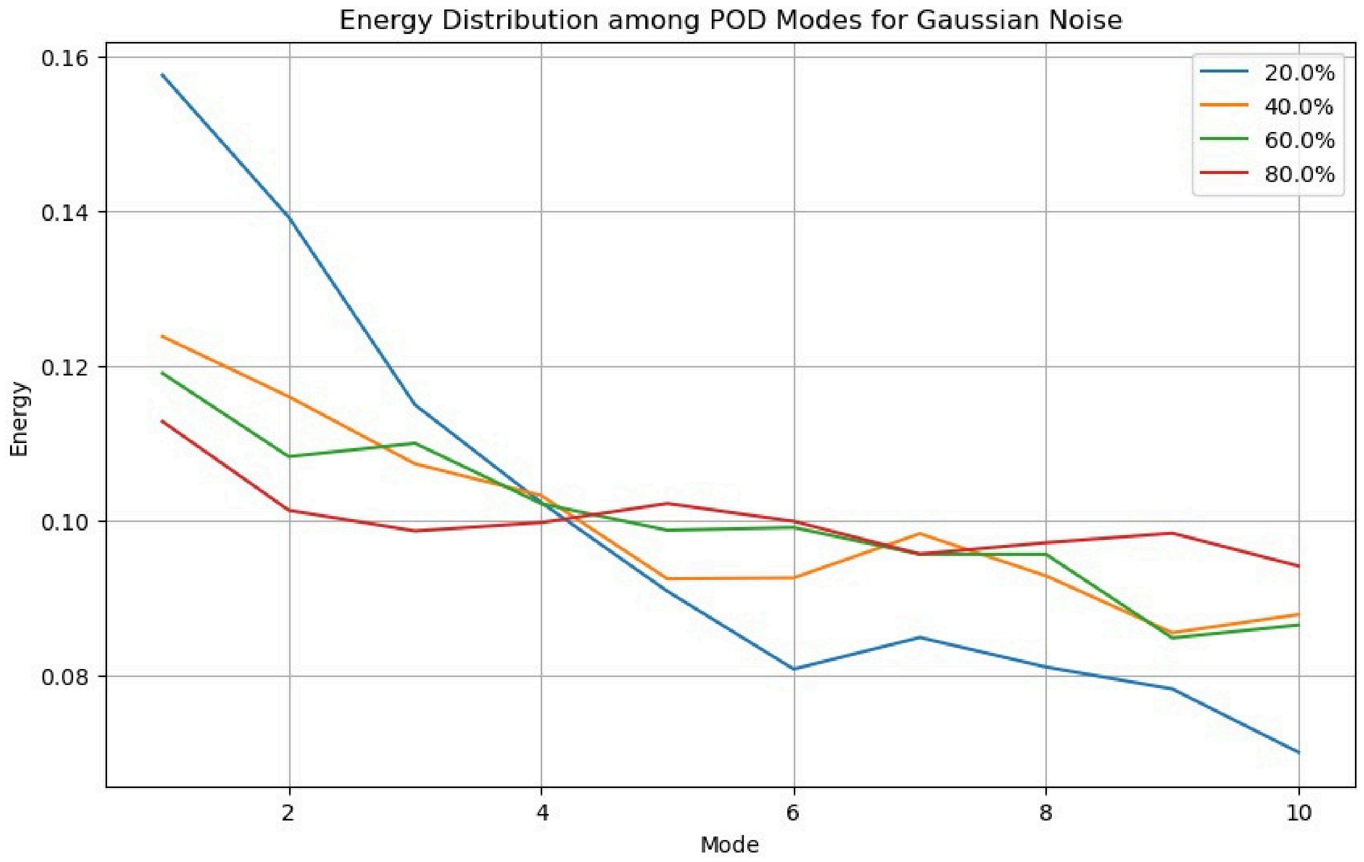


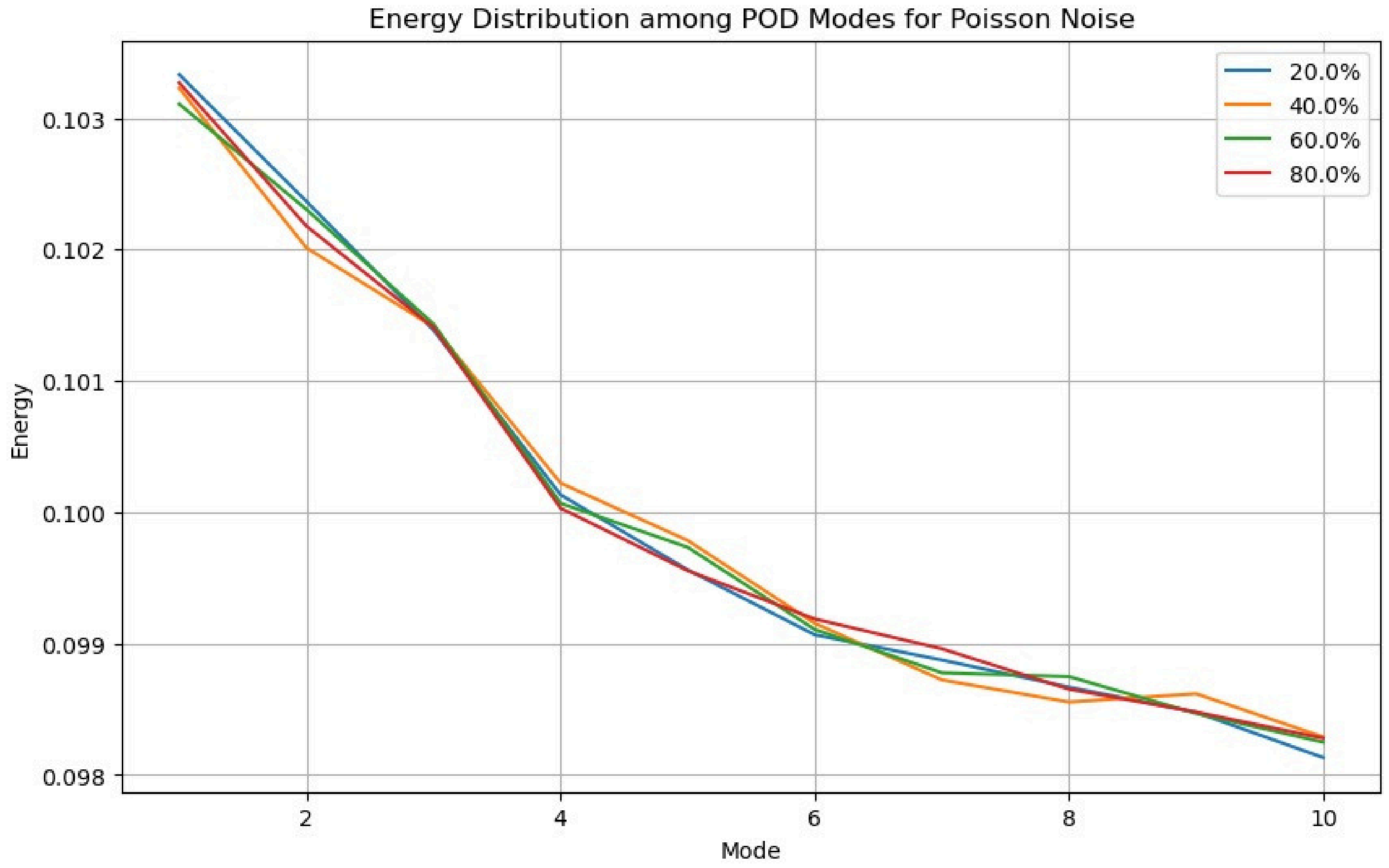


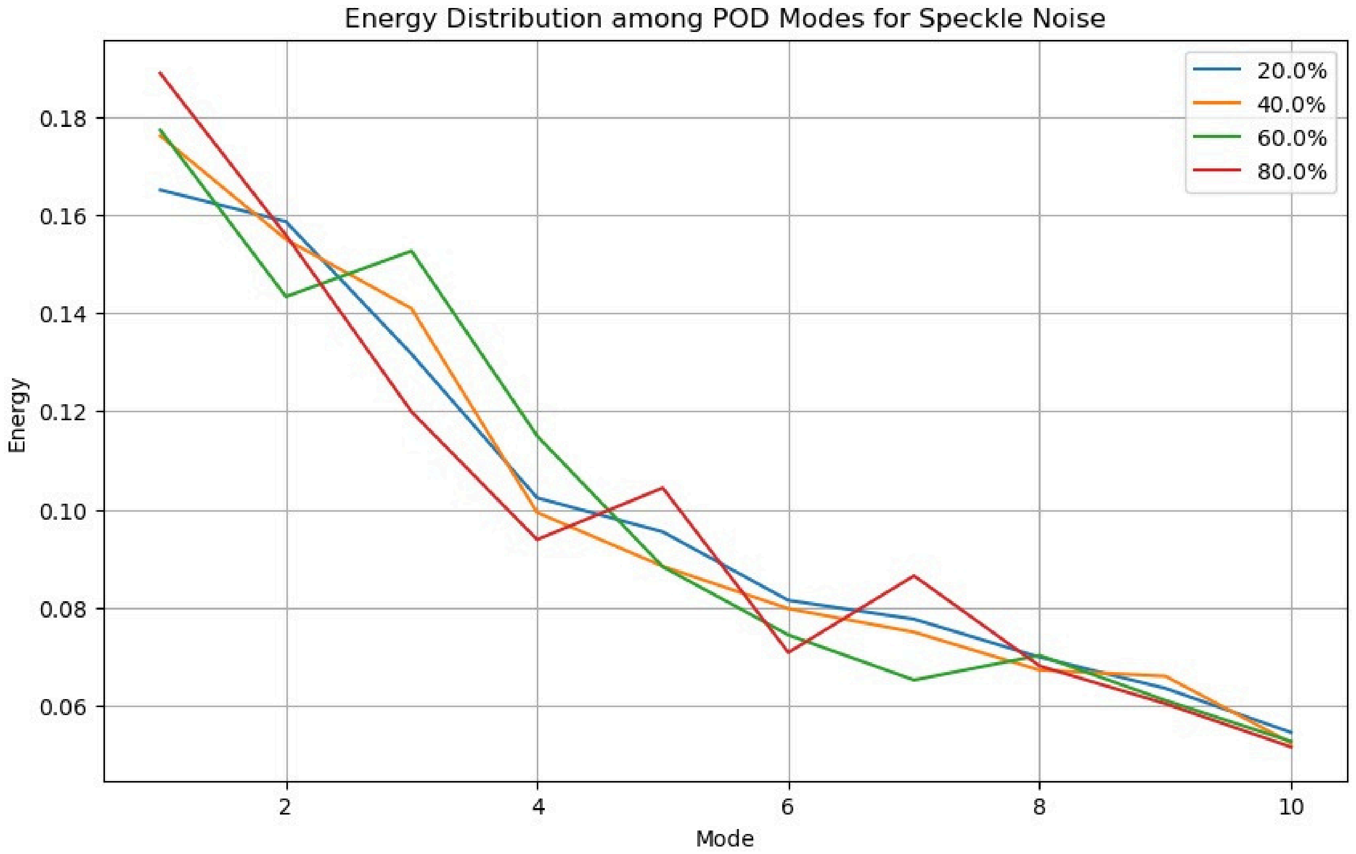


EFFECT ON POD MODES

The change on energy wrt the modes

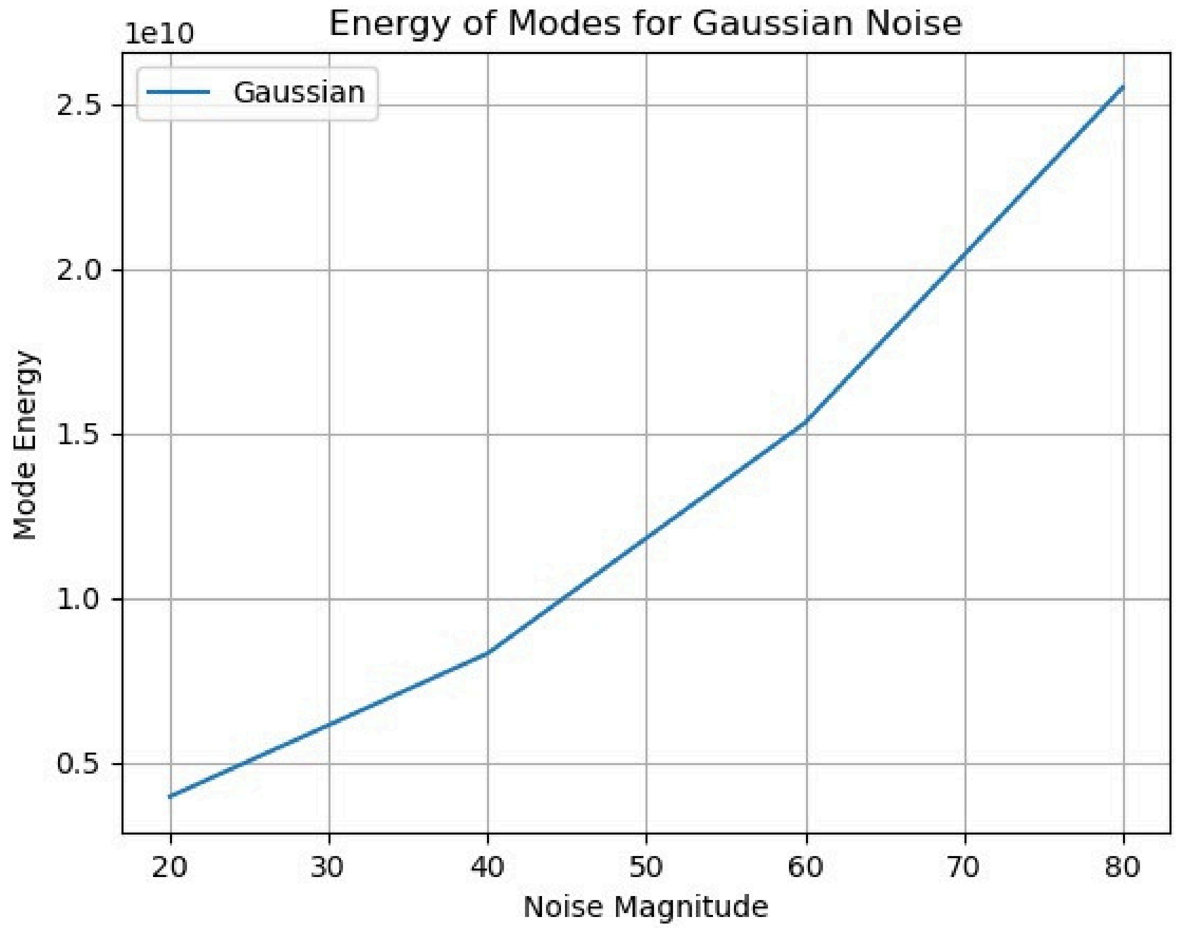


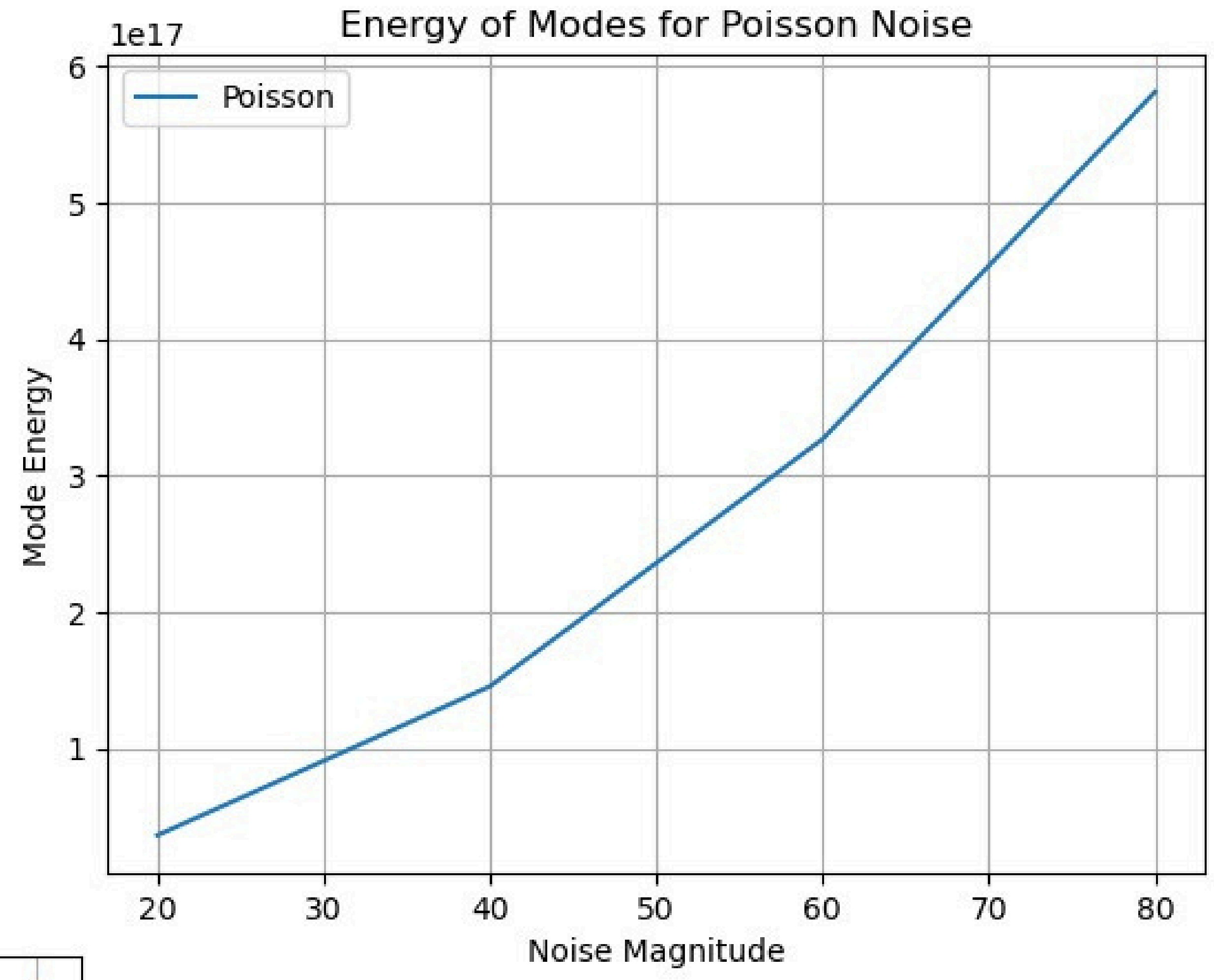


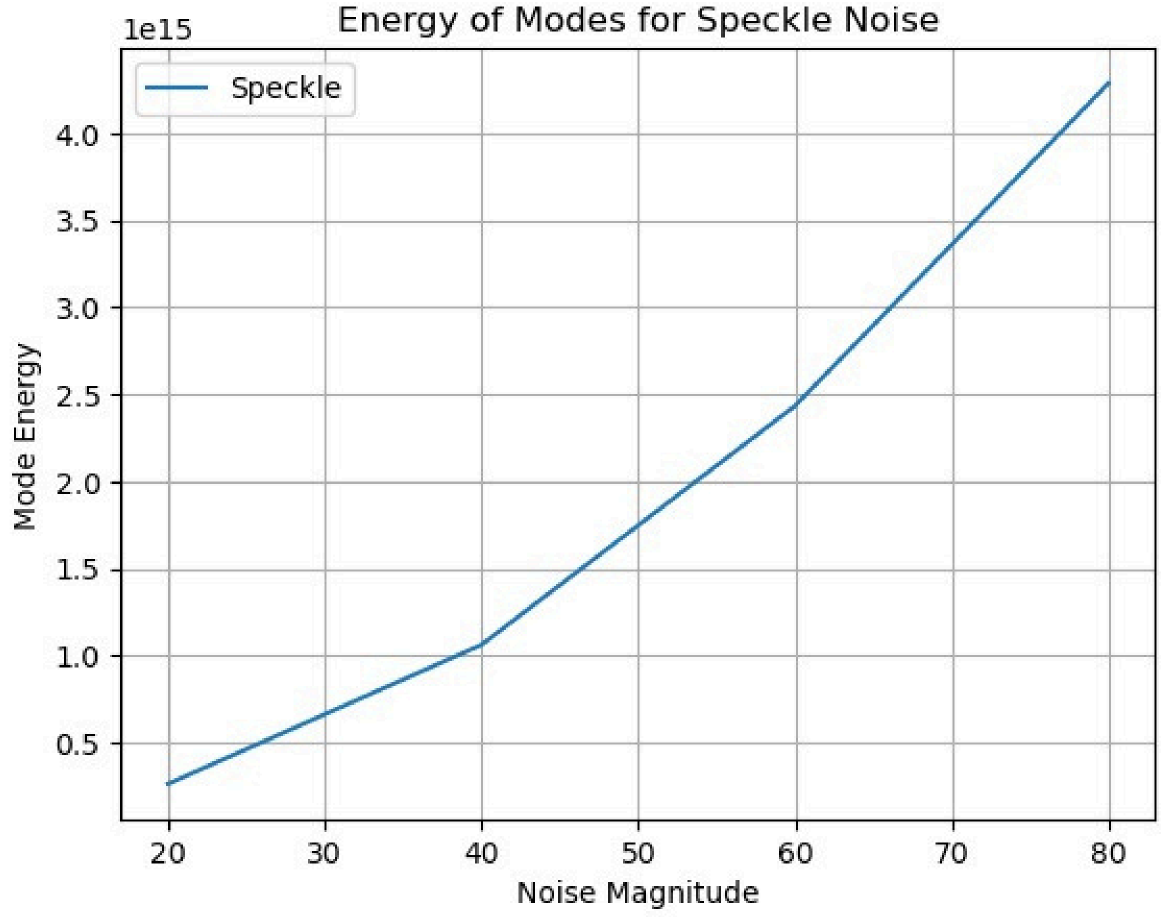


EFFECT ON POD MODES

The noise magnitude affect on the modal energy





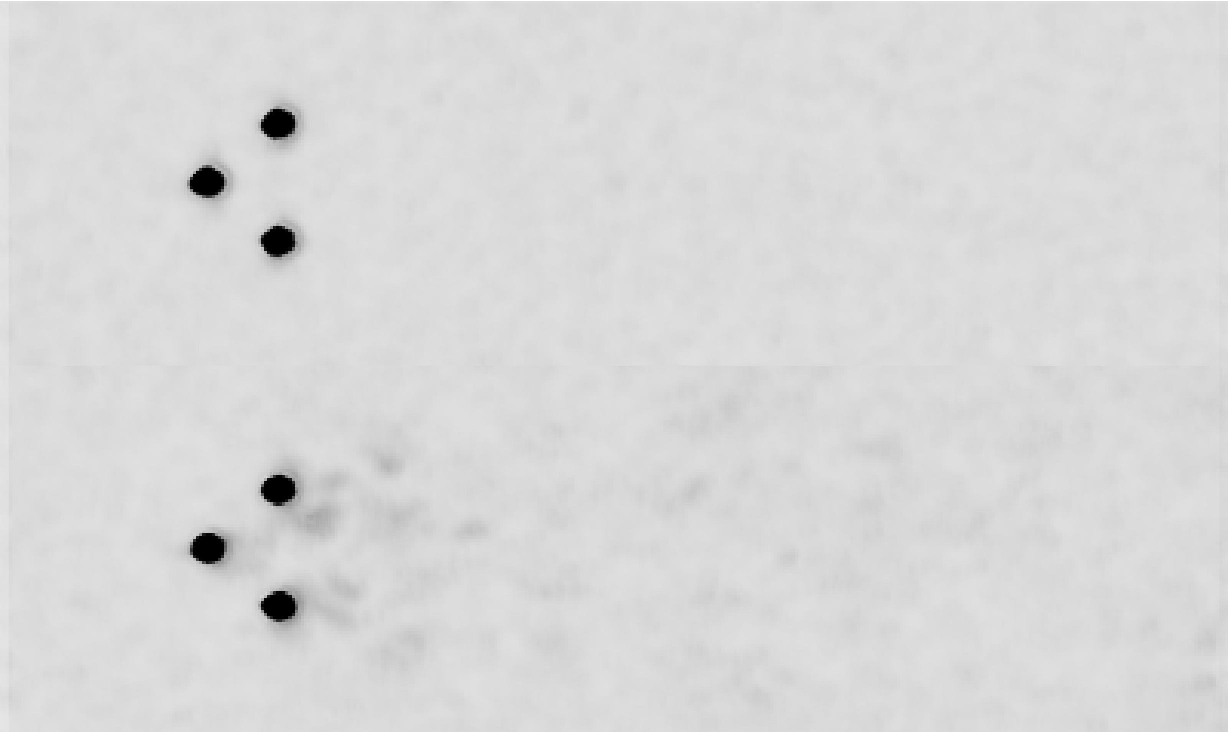


DENOISING IMAGES

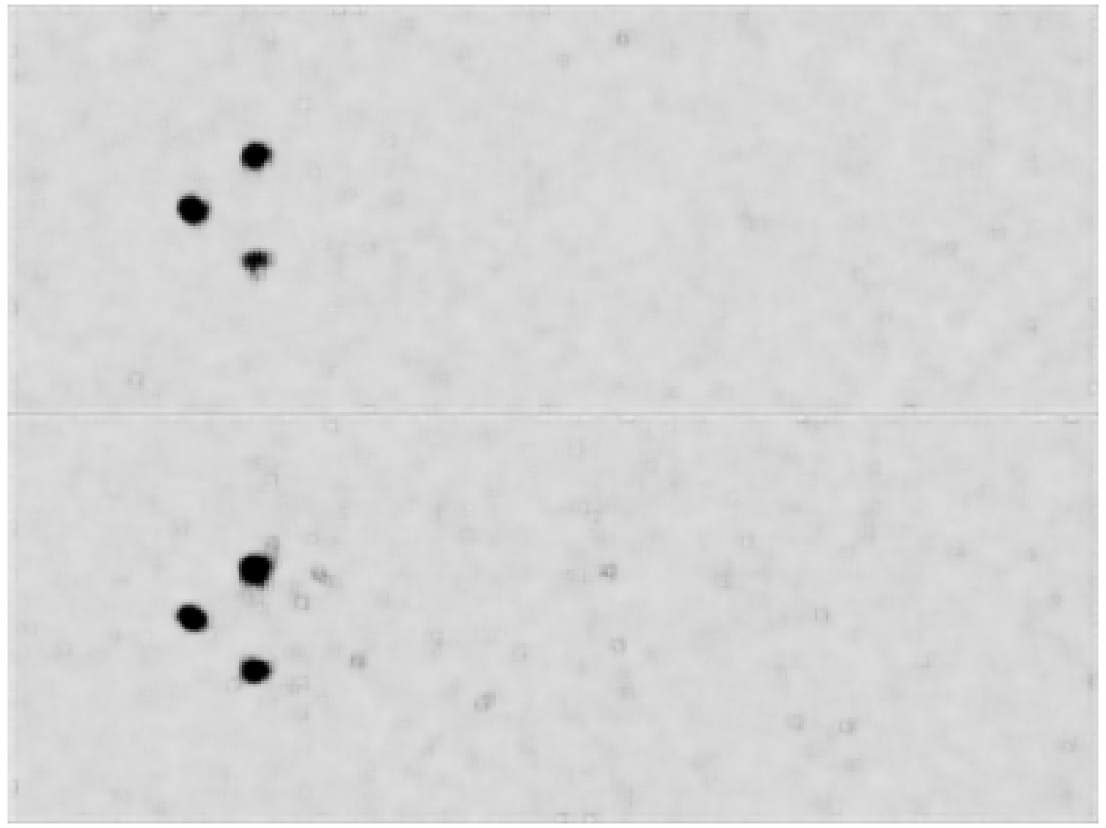
We are using CNN model to denoised images.
We trained out model with noisy and clean images using MSE loss function

i have used gaussian images with magnitude 20 and 80

denoised image of gaussian magnitude 20%



denoised image of gaussian magnitude 80%



DENOISING IMAGES

gaussian images with magnitude 20:-

```
Average PSNR: 28.6157526517026
Average SSIM: 0.8682079712361356
Average MSE: 41.7625833333333
Average MAE: 144.94845000000007
```

gaussian images with magnitude 80:-

```
Average PSNR: 26.916430985610337
Average SSIM: 0.8451103671961434
Average MSE: 41.7625833333333
Average MAE: 144.94845000000007
```

THANK YOU