Circuits and Systems
Mekelweg 4,
2628 CD Delft
The Netherlands
https://sps.ewi.tudelft.nl/

SPS-2024-00

M.Sc. Thesis

Physics-informed machine learning for nowcasting extreme rainfall

Junzhe Yin

Abstract

The thesis explores an innovative technique for enhancing the precision of short-term weather forecasts, particularly in predicting extreme weather phenomena, which present a notable challenge for existing models such as PySTEPS due to their volatile behavior. Leveraging precipitation and meteorological data sourced from the Royal Netherlands Meteorological Institute (KNMI), the research innovates through the development of a physics-informed neural network. Central to this approach is the implementation of a Physics-Informed Discriminator GAN (PID-GAN), a method that embeds physical principles directly into the adversarial training regime. The architecture is marked by the integration of a Vector Quantization Generative Adversarial Network (VQ-GAN) and a Transformer as the generator, complemented by a temporal discriminator as the discriminator component. Results from this study indicate a notable advancement over traditional numerical weather prediction and cutting-edge deep learning models, underscoring the PID-GAN model's superiority in delivering accurate precipitation nowcasting metrics.

TU Delft

Faculty of Electrical Engineering, Mathematics and Computer Science

Delft University of Technology

Physics-informed machine learning for nowcasting extreme rainfall

Thesis

submitted in partial fulfillment of the
requirements for the degree of

Master of Science

in

Electrical Engineering

by

Junzhe Yin
born in Baoji, China

This work was performed in:

Circuits and Systems Group
Department of Microelectronics
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology

Delft University of Technology

Delft University of Technology
Department of
Microelectronics

The undersigned hereby certify that they have read and recommend to the Faculty of Electrical Engineering, Mathematics and Computer Science for acceptance a thesis entitled **"Physics-informed machine learning for nowcasting extreme rainfall"** by **Junzhe Yin** in partial fulfillment of the requirements for the degree of **Master of Science**.

Dated: March 25,2024

Chairman: ______________________
prof.dr.ir. J. Dauwels

Advisor: ______________________
prof.dr.ir. J. Dauwels

Committee Members: ______________________
prof.dr.ir. J. Dauwels

____
prof.dr.ir. L. Abelmann

____
Prof.dr.ir. R. Uijlenhoet

Abstract

This research embarks on an exploratory journey to enhance the precision of short-term weather forecasts, with a particular emphasis on predicting extreme weather phenomena such as heavy rain, which poses a substantial challenge to existing models like PySTEPS due to their inherently volatile behavior. By leveraging precipitation and meteorological data sourced from the Royal Netherlands Meteorological Institute (KNMI), this study pioneers the development of a physics-informed neural network framework. At the heart of this novel approach is the implementation of a Physics-Informed Discriminator Generative Adversarial Network (PID-GAN), a methodology that seamlessly integrates physical principles directly into the adversarial training regime. This innovative architecture, characterized by the amalgamation of a Vector Quantization Generative Adversarial Network (VQ-GAN) and a Transformer as the generator, complemented by a temporal discriminator, signifies a groundbreaking advancement in the field of precipitation nowcasting. Our findings illuminate the PID-GAN model's superiority in delivering not only significantly improved nowcasting metrics but also in its capacity to capture and model the complexities and non-linearities of weather patterns more effectively than traditional numerical weather prediction and cutting-edge deep learning models. This research not only underscores the potential of incorporating deep learning models into nowcasting but also sets a new benchmark for the forecasting of extreme weather events, thereby contributing profoundly to the ongoing efforts in mitigating their impacts on society. The thesis explores an innovative technique for enhancing the precision of short-term weather forecasts, particularly in predicting extreme weather phenomena, which present a notable challenge for existing models such as PySTEPS due to their volatile behavior. Leveraging precipitation and meteorological data sourced from the Royal Netherlands Meteorological Institute (KNMI), the research innovates through the development of a physics-informed neural network. Central to this approach is the implementation of a Physics-Informed Discriminator GAN (PID-GAN), a method that embeds physical principles directly into the adversarial training regime. The architecture is marked by the integration of a Vector Quantization Generative Adversarial Network (VQ-GAN) and a Transformer as the generator, complemented by a temporal discriminator as the discriminator component. Results from this study indicate a notable advancement over traditional numerical weather prediction and cutting-edge deep learning models, underscoring the PID-GAN model's superiority in delivering accurate precipitation nowcasting metrics.

Acknowledgments

Firstly, my profound gratitude is extended to Prof. Dr. Ir. Justin Dauwels, my thesis advisor, for his exceptional guidance, unwavering support, and insightful critiques throughout my research journey. His deep expertise and mentorship have been crucial in directing my thesis to successful completion. Equally, I am indebted to Dr. Cristian Meo, my daily supervisor, who provided day-to-day guidance, insightful feedback, continuous encouragement, and constructive suggestions, all of which have been instrumental in refining my work. I also want to thank Dr. Ruben Imhoff for his ability to offer innovative solutions to complex problems, enriching my research with his invaluable insights. Special thanks go to my colleagues, Ankush Roy and Zeina Bou Cher, for their encouragement and engaging discussions. Their friendship and support have greatly enhanced this academic venture, making it a truly memorable experience. In conclusion, I express my heartfelt appreciation to everyone who has contributed to the successful completion of this thesis. Your support and encouragement have been a cornerstone of this journey, and I am deeply thankful for your invaluable contributions.

Junzhe Yin
Delft, The Netherlands
March 25,2024

Nomenclature

Abbreviations

Abbreviation	Definition
AENN	Adversarial Extrapolation Neural Net
AR Transformer	AutoRegressive Transformer
AUC	Area Under the Curve
AWS	Automatic weather station
CSI	Critical Success Index
CNN	Convolutional Neural Networks
ConvGRU	Convolutional Gated Recurrent Unit
FA	False Alarm
FAR	False Alarm Ratio
FSS	Fractions Skill Score
HR	Hate Rate
KNMI	Royal Netherlands Meteorological Institute
MAE	Mean Absolute Error
MSE	Mean Squared Error
NWP	Numerical Weather Prediction
GANs	Generative Adversarial Networks
PIML	Physical-Informed Machine Learning
PIDL	Physics-Informed deep learning
PCC	Pearson Correlation Coefficient
POD	Probability of Detection
ROC	Receiver Operating Characteristic
RT dataset	Real Time Dataset
RNN	Recurrent Neural Networks
VQ-GAN	Vector Quantized Generative Adversarial Network
EVL	Extreme Value Loss

Contents

Abstract **v**

Acknowledgments **vii**

Nomenclature **ix**

1 Introduction **1**
1.1 Background 1
1.2 Literature review 2
1.2.1 Precipitation Nowcasting Deep learning Model 2
1.2.2 Physics-informed machine learning 4
1.3 Research Purpose 5
1.4 Benchmarks 5
1.4.1 PySTEPS 5
1.4.2 Nuwä-EVL 6

2 Data and Problem Statement **7**
2.1 Data 7
2.1.1 Radar Rainfall Product 7
2.1.2 Automatic weather stations Hourly data 10
2.1.3 ERA5 reanalyses 10
2.2 Problem Statement 11

3 Fundamentals of Precipitation Physics **13**
3.1 Moisture Conservation Equation 13
3.1.1 The Seven Basic Equations in Weather Forecasting Models . . . 13
3.1.2 The wind component 15
3.1.3 The specific humidity 15
3.1.4 Makkink equation 16
3.1.5 Simplified version of Moisture Conservation Equation 16

4 Method **19**
4.1 Proposed Model 19
4.1.1 Background 19
4.1.2 Model 22
4.2 Managing extreme precipitation events 31
4.3 Evaluation metrics 32
4.3.1 Pearson's Correlation Coefficient (PCC) 32
4.3.2 Mean absolute error (MAE) 32
4.3.3 Critical success index (CSI) 32
4.3.4 False alarm ratio (FAR) 33
4.3.5 Fractions skill score (FSS) 33

4.3.6 Hit Rate(HR) . . . 33
4.3.7 False Alarm Rate (FA) . . . 34
4.3.8 Receiver Operating Characteristic (ROC) Curve . . . 34
4.3.9 Precision-Recall Curves . . . 34
4.3.10 Physical Consistency (MSE) . . . 35

5 Results 37
5.1 Nowcasting performance on the Whole Netherlands . . . 38
5.1.1 Evaluation on the different lead-time . . . 38
5.1.2 Results for post-pressing and average . . . 42
5.1.3 Summary of the nowcasting performance . . . 44
5.2 Extreme events detection . . . 45
5.2.1 Evaluation of fixed threshold of extreme events . . . 46
5.2.2 Comprehensive Assessment of Extreme Events Identification Capability . . . 47

6 Conclusion and Further Research 51
6.1 Conclusion . . . 51
6.2 Further Research . . . 52
6.2.1 Data . . . 52
6.2.2 Model . . . 52

Bibliography 55

A Appendix: Interpolated Map for the meteorological data 60

B Appendix:AWS data 62

C Reconstruction Examples 64

D Nowcasting results Examples 65

E Extreme events detection on 12 Dutch catchments 66

F Comparison of Generation time of the different models 72

List of Figures

2.1 The Netherlands map displaying the 12 catchments highlighted in green and the research area delineated by the large circle [10]. 7
2.2 Sample visualizations of radar datasets, with areas lacking data depicted in grey. 8
2.3 Catchment-level rainfall intensity analysis 10
2.4 Red box indicates the study area in this thesis 11

4.1 Proposed model structure 22
4.2 The proposed generator framework begins with the initial phase, where the VQ-GAN acquires a codebook. Each code, or an amalgamation of multiple codes, within this codebook corresponds to a specific pattern observed in the radar image. This allows the radar information to be condensed into a series of indices. These indices, in turn, are methodically modelled by the autoregressive transformer. During prediction, a set of conditional indices is fed into the transformer, which has been trained to sequentially produce probabilistic distributions for the subsequent prediction tokens in an autoregressive manner. 23
4.3 Encoder model structure 24
4.4 Decoder model structure 24
4.5 Discriminator model structure 25
4.6 Radar image samples with different spatial resolutions, original image have resolutions 256 × 256 and Interpolated image have resolutions 128 × 128. 26
4.7 Causal Attention 27
4.8 Autoregressive transformer structure 28
4.9 Training stage of transformer 28
4.10 Generation stage of transformer 29
4.11 PID-GAN model structure 30
4.12 Temporal Discriminator structure 30

5.1 Evaluation of the 3-hour prediction: (continuous metrics) Subfigure (a) presents the Pearson Correlation Coefficient (PCC), and Subfigure (b) displays the Mean Absolute Error (MAE). This figure illustrates how the metrics vary with the forecasting lead time and includes a legend showing average scores over 3 hours. 39
5.2 Evaluation of the 3-hour prediction: (categorical scores) (CSI and FAR with different thresholds: a,b for 1mm; c, d for 2mm and e, f for 8mm). This figure illustrates how the metrics vary with the forecasting lead time and includes a legend showing average scores over 3 hours. 40

5.3 Evaluation of the 3-hour prediction: (spatial scores) (FSS analysis at varying spatial resolutions: Subfigures (a), (b), (c), and (d) correspond to length scales of 30 km, 20 km, 10 km, and 1 km, respectively.) This figure illustrates how the metrics vary with the forecasting lead time and includes a legend showing average scores over 3 hours. 41

5.4 The connection between the number of averages and the pixel-level evaluation, specifically the 3-hour averaged Pearson Correlation Coefficient (PCC) in sub-figure a, and the Mean Absolute Error (MAE) in sub-figure b. 42

5.5 The connection between the number of averages and the pixel-level evaluation, specifically the 3-hour averaged Critical Success Index (CSI) in sub-figure (a,c,e), and the False Alarm Ratio (FAR) in sub-figure (b,d,f). 43

5.6 The connection between the number of averages and the pixel-level evaluation, specifically the 3-hour averaged Fraction Skill Score (FSS) at varying spatial resolutions: Subfigures (a), (b), (c), and (d) correspond to length scales of 30 km, 20 km, 10 km, and 1 km, respectively. 43

5.7 Sub-figure a. The comprehensive ROC curves present the detection of extreme events over 3 hours. The points on the curve, arranged from left to right, correspond to precipitation thresholds ranging from 10mm to 0.5mm at the catchment level, where the reference threshold for the ground truth is set as 5mm/3h. Sub-figure b. The ROC curve is modified by constraining the hit rate to exceed 0.5 and the false alarm rate to be between 0.1 and 0.5. 48

5.8 Sub-figure a presents comprehensive precision-recall curves for detecting extreme events over 3 hours. The points on the curve, arranged from left to right, correspond to precipitation thresholds ranging from 10mm to 0.5mm at the catchment level, where the reference threshold for the ground truth is set as 5mm/3h. Sub-figure b. The precision-recall curve is modified by constraining the precision between 0.2 and 0.8 and the hit rate between 0.5 and 0.8. 49

5.9 Sub-figure a presents comprehensive precision-recall curves for detecting extreme events over 3 hours. The points on the curve, arranged from left to right, correspond to precipitation thresholds ranging from 10mm to 0.5mm at the catchment level, where the reference threshold for the ground truth is set as 2mm/3h. Sub-figure b. The precision-recall curve is modified by constraining the precision between 0.2 and 0.8 and the hit rate between 0.5 and 0.8. 50

A.1 Example of Interpolated Map for the meteorological data: evapotranspiration rate (EVA), Dew Point Temperature (DW-temp), East-west wind component at 100m (u100), South-north wind component at 100m (v100) 60

A.2 Example of Interpolated Map for the meteorological data: East-west wind component at 10m (Wind-U), South-north wind component at 10m (Wind-V), specific humidity (humidity) 61

C.1 Example of reconstruction of Precipitation fields by the VQ-GAN, input is the original image, reco is the reconstruction. 64

D.1 Nowcasting result of different models, t=2019/11/28/05:45 65

E.1 ROC and Precision-recall curves for Aa and Beemster 66
E.2 ROC and Precision-recall curves for Delfland and Reusel 67
E.3 ROC and Precision-recall curves for Linde and Rijnland 68
E.4 ROC and Precision-recall curves for Roggelsebeek and Dwarsdiep . . . 69
E.5 ROC and Precision-recall curves for Luntersebeek and Grote Waterleiding 70
E.6 ROC and Precision-recall curves for Hupsel Brook and Regge 71

List of Tables

2.1 12 Dutch Catchment Areas 9
2.2 Catchment-level rainfall intensity analysis 10

4.1 Confusion Matrix 33

5.1 Summary of the 3-hour averaged precipitation nowcasting skill of different models (Pixel-level evaluation). 44
5.2 Pixel-level evaluation of Physical Consistency 44
5.3 Summary of the 5% extreme event detection performance of different models (Catchment-level evaluation, RT dataset) 2mm/3h 46
5.4 Summary of the 1% extreme event detection performance of different models (Catchment-level evaluation, RT dataset) 5mm/3h 46
5.5 Catchment-level evaluation results for extreme-event forecasting, averaged over 3927 catchment-level events. 48

B.1 Description of available data parameters from AWS 62
B.2 Geographic and Altitudinal Data of AWS 63

F.1 Comparison of Generation time of the different models 72

Introduction 1

1.1 Background

Globally, more frequent extreme weather events significantly impact human life, including floods, sandstorms, tornadoes, forest fires, hurricanes, blizzards, heatwaves, and droughts, significantly impact human life. These events lead to human deaths, economic damage, destruction of homes and critical infrastructure, health emergencies, and considerable environmental harm [1]. They emphasize the urgent need for enhanced disaster readiness, robust and adaptable infrastructure, and decisive actions against climate change to protect communities and ensure long-term sustainability.

Extreme precipitation events are particularly damaging, especially in the tropical regions of South America, Southeast Asia, and parts of Europe, where they have become more frequent due to environmental degradation and climate change. The rise in these weather events causes severe flooding, soil erosion, loss of agricultural productivity, and disruption of natural ecosystems [2]. This trend emphasizes the need for comprehensive climate strategies to reduce their impact, strengthen infrastructure resilience, and adjust water management practices to protect communities and preserve the environment.

Precipitation prediction can be split into two primary categories: long-term and short-term, known as nowcasting. Due to the limitations of algorithms and future unpredictability, long-term predictions tend to be general, covering large areas with low detail and a high potential for inaccuracies. Conversely, nowcasting provides specific and reliable rainfall forecasts for localized regions within a short period, usually up to six hours, using weather radar and satellite imagery [3]. As defined by the World Meteorological Organization [4], nowcasting aims to offer prompt and accurate forecasts of upcoming rain events, enhancing readiness and response efforts for extreme weather occurrences like flash floods and landslides.

Nowcasting systems specialize in predicting weather conditions up to six hours ahead, achieving an accuracy often unattainable by Numerical Weather Prediction (NWP) models, traditionally used for broader weather forecasting. While NWP models calculate future weather by simulating atmospheric and oceanic phenomena, their earlier versions needed to be improved by low spatial-temporal resolution and lengthy processing times, rendering them less effective for immediate weather predictions [5]. Despite enhanced resolution due to improvements in computational capabilities, NWP models still cannot match the immediacy and precision of nowcasting, which is crucial for issuing timely warnings about severe weather threats such as floods and landslides.

Nowcasting relies on straightforward, high-resolution techniques that utilize current atmospheric data. It leverages recent weather radar data, typically spanning the previous 1 to 3 hours, to make precise predictions about upcoming weather. This approach is convenient in identifying localized weather patterns crucial for accurate rainfall forecasting.

In contrast, ensemble NWP systems that forecast more significant weather phenomena need help rapidly predict localized, convective weather events due to intensive computational demands and limited observational data. Radar extrapolation models gain an advantage by using real-time data and methods such as optical flow and statistical analyses for weather prediction [3]. Radar Extrapolation methods, such as PySTEPS enhance these predictions by integrating multiple forecasting techniques [6], [7], thereby reducing uncertainty. Nonetheless, NWP and radar extrapolation approaches require substantial computational resources and time to produce forecasts [8].

The studies [9], [10] assessed the PySTEPS algorithm's performance over a series of rain events, discovering that its probabilistic model notably surpassed benchmarks in key evaluation metrics. However, PySTEPS forecasts were limited by not accounting for growth and decay processes, highlighting the need to focus on nowcasting uncertainties. While statistical nowcasting methods bypass the need for historical data, their lengthy computation time and the separation of optical flow estimation from the extrapolation process pose challenges in optimizing performance.

1.2 Literature review

1.2.1 Precipitation Nowcasting Deep learning Model

The challenges inherent in optical flow-based precipitation forecasting methods have recently prompted a shift towards machine learning solutions, focusing on convolutional neural networks (CNNs) and recurrent neural networks (RNNs). These technologies have shown great promise in capturing complex spatio-temporal patterns for weather prediction, eliminating the reliance on traditional data assimilation techniques.

Shi et al. [11] marked a significant advancement by introducing a novel convolutional encoder-decoder framework integrated with ConvLSTM. This approach uses convolution operations for transitions between states and has proven more effective in managing spatio-temporal correlations than optical flow methods. However, despite its advancements, ConvLSTM faces challenges in generating accurate forecasts over long periods and extreme weather conditions due to its complex parameter set and susceptibility to blurring effects. Shi et al. [12] developed the TrajGRU model to overcome these limitations, which addresses the shortcomings of ConvLSTM's convolutional filters with location-invariant characteristics. By implementing balanced loss functions, TrajGRU enhances nowcasting accuracy for significant rainfall, offering a streamlined model with fewer parameters and improved performance compared to ConvLSTM.

Furthermore, research has validated the superiority of CNNs over RNNs, including LSTMs, especially in short-term precipitation forecasting. Qiu et al. [13] introduced a multi-task CNN model that leverages data from multiple sites to refine rainfall forecasting accuracy. This model highlights the strengths of CNNs in processing spatial information and synthesizing data from diverse locations, leading to better predictions for minor rainfall events. Additionally, RainNet has made considerable progress by utilizing extensive radar data archives for precipitation nowcasting [14]. RainNet emphasizes the reliability and precision of CNNs in capturing spatio-temporal precipitation patterns. Nonetheless, the multi-task CNN model and RainNet face limitations in accurately forecasting heavy rainfall, underscoring a vital area for ongoing research and model optimization.

Until now, models for short-term precipitation prediction have been categorized into two types: those based solely on CNNs and hybrid models that combine CNNs with RNNs. These approaches have introduced significant advancements, yet they often encounter issues with blurring in their extrapolations. Such difficulties primarily arise from employing loss functions like mean squared error (MSE) and mean absolute error (MAE), which average out all potential outcomes, thereby obscuring essential details in the forecasts. Moreover, this challenge is exacerbated by the multi-modal and skewed nature of intensity distributions in radar imagery, which needs to align better with the MSE's inclination towards favouring unimodal distributions, further complicating the accuracy of weather forecasting.

Generative Adversarial Networks (GANs) [15] have become pivotal in enhancing the accuracy and realism of weather prediction models, addressing the prevalent issue of blurring effects associated with deep learning forecasts. By leveraging adversarial training, these models can generate more detailed predictions and capture a wider array of meteorological phenomena.

A noteworthy model by Tian et al. [16], which employs a Convolutional Gated Recurrent Unit (ConvGRU) as the generator and a sophisticated five-layer convolutional neural network as the discriminator, has shown promising results. This approach, known as GA-ConvGRU, is particularly effective under conditions of high rainfall intensity, delivering forecasts that are more accurate but also stable and consistent.

Building on this innovation, Jing et al. [17] introduced the AENN model, which features a conditional generator working in tandem with both temporal and spatial discriminators. This model is distinguished by its ability to predict radar echoes with remarkable texture details and accurately model the evolution of echoes, even though it experiences a slight reduction in predictive accuracy as the forecast duration extends.

Further enriching the landscape of GAN-based weather forecasting, GAN-argcPredNet melds a deep encoding-decoding framework with an innovative ConvLSTM configuration [18]. This model stands out for its adeptness at preserving spatiotemporal information, thereby improving the clarity and sophistication of features at finer scales.

Despite a noted decrease in performance with increasing rainfall intensity, it proves to be effective in accurately predicting the shape and intensity of radar echoes.

In conclusion, the advent of GAN-based models signifies a substantial breakthrough in the domain of weather forecasting. These innovative approaches not only provide more accurate predictions but also address the intrinsic difficulties of deep learning techniques, such as training instability and diminished performance in severe weather scenarios. With ongoing advancements, these models are poised to transform meteorological forecasting, enabling the prediction of weather events with unparalleled accuracy and detail.

While GAN-based nowcasting models show promise in generating more authentic predictions, there are still areas that require further enhancement. Particularly, there is room to improve the accuracy of forecasting heavy rainfall events. Additionally, the shapes of extrapolated echoes don't consistently match observed ground-truth data, and occasional fluctuations in echo intensity undermine the reliability of these forecasts.

1.2.2 Physics-informed machine learning

Physics-informed machine learning (PIML) [19] is gaining prominence as it tackles a significant challenge in applying traditional machine learning (ML) within scientific domains, notably in weather and climate forecasting. While ML models excel at parsing intricate datasets and forecasting results, they frequently overlook the physical laws governing the systems they aim to replicate [20]. This oversight can result in unreliable forecasts, particularly for situations not encountered during the training phase. PIML aims to overcome this issue by incorporating domain-specific insights and fundamental physical principles into the training process, ensuring that the models derive knowledge from data and conform to the inherent laws of nature [21]. This methodology improves the accuracy and dependability of ML deployments in scientific research, facilitating more precise predictions and a deeper understanding of complex physical systems.

Integrating physical priors into models can lead to predictions that are more accurate and physically consistent, especially in tasks requiring generalization. It has been suggested by several researchers to merge deep learning models with knowledge in physics, enhancing the models' robustness and reliability [21]–[25].

According to the case study by Kashinath et al., [21], PIML models can lead to improvements across various aspects, including enhanced physical consistency, greater accuracy, quicker training times, improved convergence, more efficient use of data, better generalization capabilities, increased interpretability, and superior scalability to handle more complex physical systems and more extensive computational infrastructures. These collective achievements underscore the significant impact of PIML on advancing weather and climate modelling.

1.3 Research Purpose

This research focuses on developing and testing a deep generative model based on physics-informed machine learning for precipitation nowcasting, specifically for predicting extreme precipitation events in regions of the Netherlands, to achieve a forecasting lead time of up to 180 minutes at 30-minute intervals. The investigation is structured around two main objectives: to achieve accurate nowcasting predictions throughout the Netherlands and to use the physical priors of precipitation to help the model generate accurate and physically consistent predictions. The dissertation is divided into two principal themes: The first theme involves formulating a novel deep generative model for precipitation nowcasting, drawing inspiration from high-resolution image synthesis research [26], and adopting a bifurcated approach that begins with a Vector Quantized Generative Adversarial Network (VQGAN) and transitions to an autoregressive transformer. The second theme explores the integration of a Physics-Informed Discriminator (PID)-GAN formulation with the deep generative model to incorporate the physical priors of precipitation [27]. Additionally, the temporal discriminator from the Adversarial Extrapolation Neural Net (AENN) model [17]is applied to achieve the GAN framework of PID-GAN.

Three pivotal questions propel this inquiry:

1. How can we develop a deep generative model that delivers accurate precipitation forecasts for the next three hours?

2. How can we identify and detect extreme precipitation events?

3. How can we inject the physical priors of precipitation into the deep generative model?

The thesis is organized in the following manner: Chapter 2 outlines the datasets employed in the study. Chapter 3 details the fundamentals of precipitation physics. In Chapter 4, we detail the architecture of the deep-learning model proposed for nowcasting. Chapter 5 offers an analysis and discussion of the findings. The conclusion and directions for future research are provided in Chapter 6.

1.4 Benchmarks

1.4.1 PySTEPS

PySTEPS is a pioneering, community-developed, open-source Python platform for advanced short-term precipitation forecasting [6]. Its modular framework stands out for incorporating diverse statistical nowcasting techniques. The methodology is anchored in the Lagrangian persistence concept, suggesting precipitation moves at a uniform velocity and direction. The forecasting initiates with the identification of the motion field from existing meteorological data, used thereafter to forecast rainfall movement. Central to PySTEPS are its principal algorithms: S-PROG, for deterministic forecasting, and STEPS, for probabilistic forecasting, where the latter synergises nowcasting with improvements from Numerical Weather Prediction (NWP) model forecasts. PySTEPS is recognized as a leading framework in statistical nowcasting due to its deterministic and probabilistic forecasts. This study delves into PySTEPS's ensemble-based probabilistic forecasting, evaluating an average from 20 ensemble forecasts, in line with Imhoff et al.'s configuration [10]. The Royal Netherlands Meteorological Institute (KNMI) validates its operational efficacy in nowcasting, endorsing its precision and reliability. Thus, our research highlights PySTEPS as a critical benchmark, evidencing its profound impact on the advancement of nowcasting technologies.

1.4.2 Nuwä-EVL

This thesis evaluates the innovative "Nuwä-EVL" model for nowcasting, which applies a deep generative technique for predicting extreme precipitation events [28]. This model, incorporating Extreme Value Loss (EVL) with an autoregressive transformer framework, excels in forecasting extreme weather occurrences, outperforming conventional methods in capturing the severity and patterns of extreme precipitation. Its advanced approach marks a significant contribution to nowcasting extreme events, making it a critical benchmark for comparing our model's efficacy in predicting extreme weather alongside PySTEPS.

Data and Problem Statement 2

2.1 Data

This section introduces the use of real-time radar data, hourly meteorological data from Automatic Weather Stations (AWS), and ERA5 reanalyses. It details their processing and significance for precipitation nowcasting in the Netherlands.

2.1.1 Radar Rainfall Product

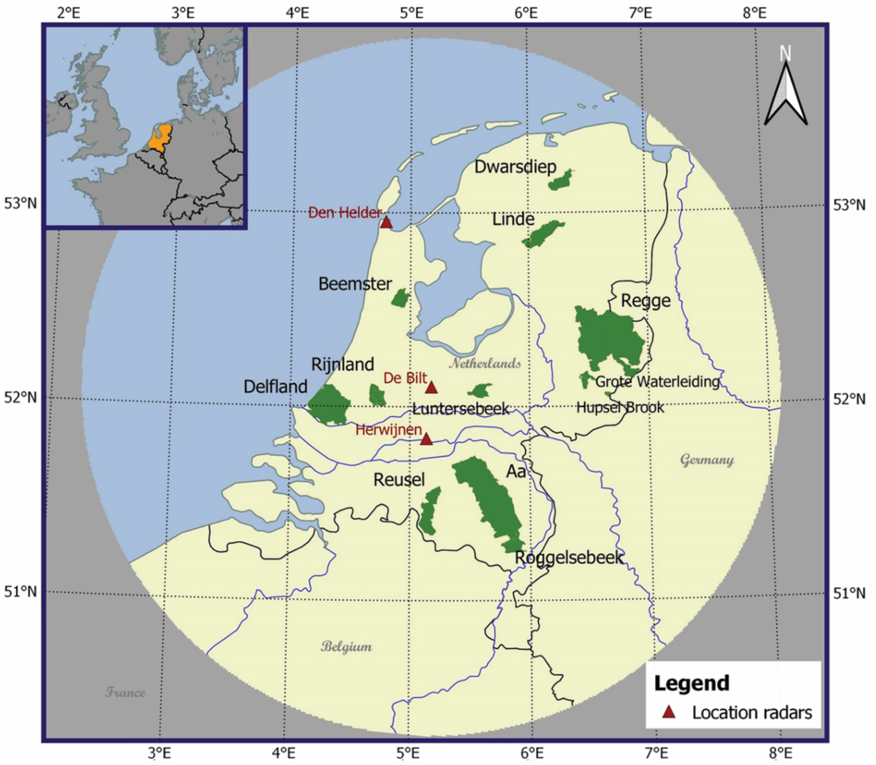


Figure 2.1: The Netherlands map displaying the 12 catchments highlighted in green and the research area delineated by the large circle [10].

KNMI manages two C-band weather radar systems, shown in Figure 2.1. The radar stations in De Bilt and Den Helder were upgraded to new installations in 2017 [29]. The currently active radar systems in Den Helder and Herwijnen are equipped with dual-polarization technology.

KNMI manipulates radar data to calculate volumetric reflectivity at a height of 1500 meters, facilitating its conversion to rainfall accumulation figures. The process begins with applying Doppler filtering and using a cloud mask derived from satellite imagery to filter out non-meteorological signals in the radar information. Following this step, the reflectivity data undergoes enhancement via range-weighted compositing. The Marshall-Palmer Z-R formula determines the precipitation rate. A specific Z-R conversion Equation (2.1) is utilized to calculate the rainfall rate from radar reflectivity data [30].

$$Z_h = 200R^{1.6}, \tag{2.1}$$

Where Z_h denotes radar reflectivity, which is measured in $mm^6 \cdot m^{-3}$, and R signifies the rainfall rate, with the unit $mm \cdot hr^{-1}$. Radar reflectivity, initially expressed in dBZ units, can be transformed into $mm^6 \cdot m^{-3}$ by using the formula $Z_h(dBZ) = 10\log_{10}(Z_h)$. The values of reflectivity less than 7dB (corresponding to a precipitation intensity less than $0.1\,mm/hr$) are disregarded, and values over 55dB (indicating a precipitation intensity exceeding $100\,mm/hr$) are standardized to 55dB during this transformation. Once refined and adjusted, the processed data is subsequently utilized to compute the quantitative precipitation estimate, which can be accessed via the KNMI website.

The RT radar dataset can provide real-time data at five-minute intervals with a spatial resolution of 1 kilometre. Each radar map is an image with dimensions of 765 by 700 pixels. However, the area for which precipitation measurement is available forms a circle with a 200 km radius centred around De Bilt, and most of the map represents unavailable data, indicated by a value of 65535. Examples of the precipitation intensity maps from the radar dataset are shown in Figure 2.2, where the grey areas denote unavailable data.

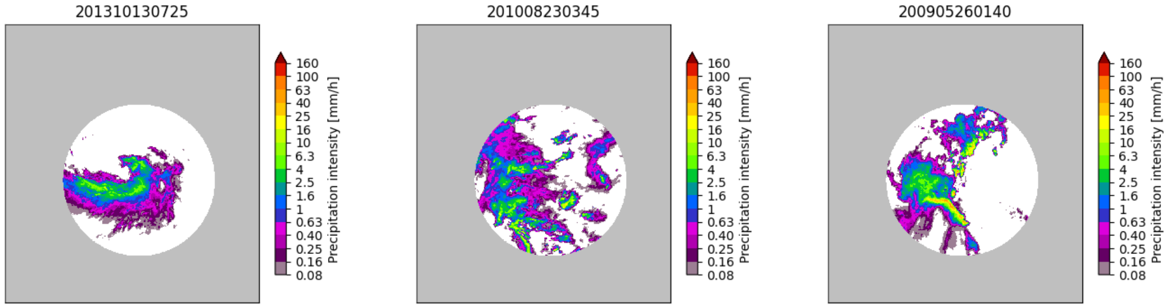


Figure 2.2: Sample visualizations of radar datasets, with areas lacking data depicted in grey.

2.1.1.1 Data analysis and Event selection

According to the study cited [10], [28], catchment areas act as focal points for the convergence of surface water runoff. Accurate short-term forecasting of weather conditions in these areas is crucial, as it enhances hydrological models essential for early flood warning systems. As a result, analysis will be conducted on the 12 catchments listed in Table 2.1, which illustrates their respective areas.

Table 2.1: 12 Dutch Catchment Areas

Number	**Catchment Name**	**Area (km^2)**
1	Regge	957
2	Aa	836
3	Delfland	379
4	Reusel	176
5	Linde	150
6	Rijnland	89
7	Roggelsebeek	88
8	Dwarsdiep	83
9	Beemster	71
10	Luntersebeek	63
11	Grote Waterleiding	40
12	Hupsel Brook	6.5

The potential inaccuracies in the RT dataset are primarily attributed to the inherent limitations of the instruments and methodologies employed for real-time rainfall data capture. Consequently, KNMI provides a more reliable radar product, the MFBS, which is refined through mean field bias and spatial adjustments and calibrated using data from 31 automatic and 325 manual rain gauges [31]. As such, the MFBS is considered a precise reference for rainfall measurements. However, due to its non-availability in real-time, this MFBS dataset is utilized for event selection in the training and testing phases of the model[10].

Table 2.2 analyses all 3-hour events from 2008 to 2014 using the MFBS dataset across the 12 Dutch catchments, conducted to calculate the average precipitation. It shows the frequency of each rainfall intensity range (Occurrence) and the corresponding percentage of the total (Percent). Most occurrences are in the lowest rainfall intensity range ($X \leq 1$ mm/3h), accounting for 91.11% of the total. The table then lists decreasing occurrences and percentages for increasing rainfall intensities, with the most minor occurrences above 9 mm/3h. Figure 2.3 of occurrence is a bar chart reflecting the data from Table 2.2, providing a visual representation of the percentage of occurrences for each rainfall intensity category. According to the distribution of precipitation intensity, there is a significant imbalance, where 91.9% catchment level events register less than 1mm/h. To handle this highly unbalanced dataset, the top 1% of the total events are selected as extreme events to generate the training, validation and testing dataset. After the final selection, 32183 events from 2008-2014 were selected as the training dataset, 3493 events as the validation dataset and 357 extreme events as the testing dataset where the extreme threshold is defined as 5mm/3h based on top 1% of the total events.

Table 2.2: Catchment-level rainfall intensity analysis

X**Average rainfall intensity (mm/3h)**	**Occurrence**	**Percent**
$0 \leq X < 1$	1,245,834	91.11%
$1 \leq X < 2$	52,492	3.84%
$2 \leq X < 5$	51,899	3.80%
$5 \leq X < 7$	9,192	0.67%
$7 \leq X < 9$	4,141	0.30%
$X \geq 9$	3,865	0.28%
Total	1,367,423	1

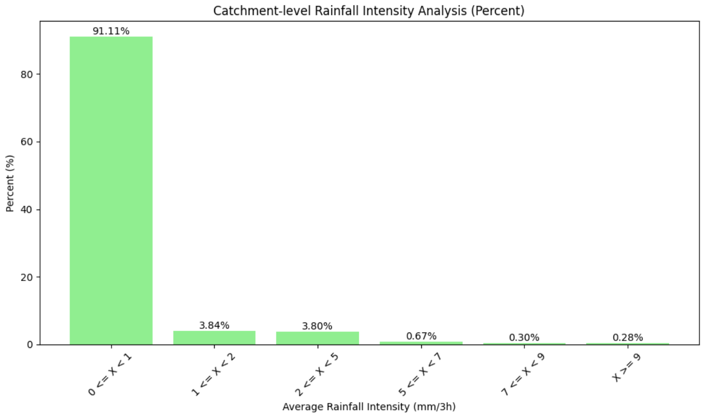


Figure 2.3: Catchment-level rainfall intensity analysis

2.1.2 Automatic weather stations Hourly data

In this thesis, another dataset was collected from automated weather stations (AWS) across the Netherlands. A total of 50 AWS, distributed throughout the country, recorded information on 22 distinct weather parameters [32]. These parameters include wind direction, hourly wind speed, temperature, dew point temperature, global radiation, air pressure, etc. Table B.1 shows the complete description of all weather parameters.

All weather parameters are measured hourly. However, inconsistencies in the hourly data can arise due to station relocations and changes in observational methods, rendering these parameters unsuitable for direct use in nowcasting tasks.

2.1.3 ERA5 reanalyses

ERA5 [33] is the latest climate reanalysis product from the European Centre for Medium-Range Weather Forecasts (ECMWF), extending back to 1940. As a successor to the ERA-Interim reanalysis, it offers an improved and comprehensive dataset constructed by merging model data with global observations, applying the data assimilation technique used in updating weather forecasts every 12 hours. A key feature of ERA5 is its provision of hourly estimates across a wide array of atmospheric, oceanic, and land-surface parameters. Each hourly estimate comes with uncertainty quantification, derived from a 10-member ensemble analysis conducted at three-hour intervals.

2.2 Problem Statement

This thesis is committed to developing a physics-informed machine learning model aimed at fulfilling three pivotal objectives:

1. Precise Precipitation Nowcasting of whole Netherlands: Nowcasting, by definition, aims to provide short-term forecasts. In this thesis, the model delivers precipitation forecasts with a 3-hour lead time at 30-minute intervals, resulting in a sequence of six frames for each event. This method utilizes radar maps from the previous 90 minutes (T-60, T-30, T minutes) as input to predict subsequent precipitation patterns for the next 3 hours (T+30, T+60, T+90, T+120, T+150, T+180 minutes). Considering the RT radar map from KNMI, which has dimensions of (765, 700) and contains largely masked areas with no valuable data, this study focuses on a 256km by 256km area (as shown in Figure 2.4), representing approximately 90% of the Netherlands' land area. This area includes all 12 catchment zones and is selected to improve model training efficiency while mitigating the impact of data limitations due to masked regions.

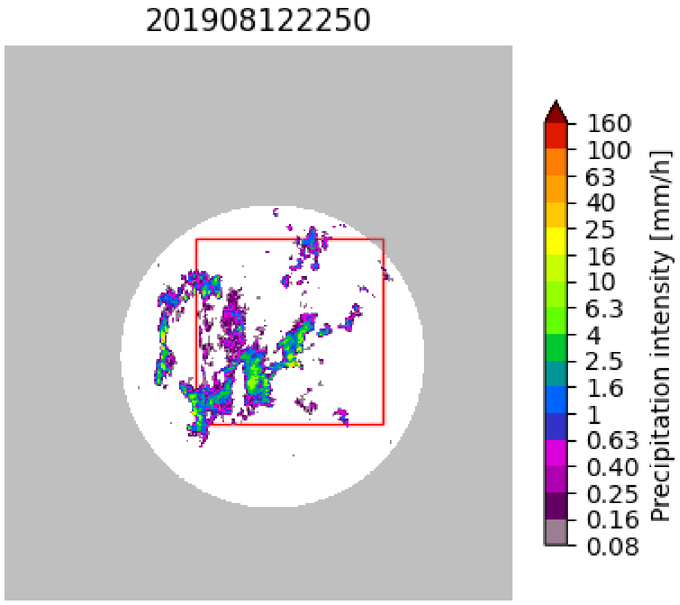


Figure 2.4: Red box indicates the study area in this thesis

2. Extreme events of catchments: In this research, extreme weather events are identified by selecting the top 1% and top 5% of all events, based on previously established criteria for extreme event selection. These criteria set corresponding

thresholds at 5 mm/3h for the top 1% and 2 mm/3h for the top 5% of events. This dual-threshold approach is adopted because relying solely on the top 1% of events, with a threshold of 5 mm/3h, is not sufficient to encompass all significant extreme events. Therefore, to capture a broader spectrum of extreme weather that could affect catchment areas, we also include those events in the top 5% category, which exceed a lower threshold of 2 mm/3h. Recognizing the critical importance of forecasting rainfall intensity for catchment areas, extreme events are thus defined as those 3-hour periods where the average precipitation surpasses these thresholds.

3. To analyze weather parameters measured hourly by AWS, we employ cubic interpolation to derive estimates at half-hour intervals (T-60, T-30, T, T+30, T+60, T+90, T+120, T+150, T+180 minutes) from the original hourly data points (T-60, T, T+60, T+120, T+180 minutes). This approach allows for a fine-grained temporal resolution in forecasting. The analysis focuses on the same 256km by 256km area covered by the radar map, ensuring consistency in the spatial domain of the study.

   To address the limited spatial coverage of the AWS and ERA5 reanalyses datasets, we applied kriging interpolation using the PyKrige package [34], achieving a spatial resolution compatible with our radar data. Here, the Kriging interpolation method is distinguished by its ability to incorporate the spatial autocorrelation and variance of the measured points, offering a robust framework for estimating values at unmeasured locations [35], [36]. Consequently, it produces a continuous surface map that aligns with the resolution of the radar data, facilitating a detailed and cohesive analysis.

   Cubic interpolation complements this by interpolating values between known data points through cubic polynomials [37], [38]. This technique ensures a smooth transition between points, which is crucial for generating more precise half-hourly weather estimates from the hourly data provided by AWS. These interpolation methods enhance the accuracy and spatial-temporal resolution of the weather parameter analysis, enabling a more nuanced understanding of the weather dynamics within the specified area.

3 Fundamentals of Precipitation Physics

3.1 Moisture Conservation Equation

Precipitation patterns, crucial components of the Earth's weather systems, are intricately shaped by the dynamics of water vapour within the atmosphere. These patterns are influenced by a myriad of factors including temperature gradients, air pressure variations, and geographic features that collectively determine the transport and condensation of moisture in the air.

To unravel the complex interplay of these elements, atmospheric scientists rely on a variety of equations that simulate the physics of the atmosphere. These equations account for the conservation of mass, momentum, and energy within the air, and they require an understanding of the thermodynamics of water vapour, which is a key component of the hydrological cycle.

In this study, a particular focus is placed on employing a sophisticated physical equation that is central to atmospheric dynamics. This equation is designed to capture the essence of fluid motion and thermodynamic processes that lead to the formation, movement, and intensity of precipitation fields. By incorporating parameters such as wind speed and direction, humidity, and temperature, the equation can simulate how a parcel of moist air moves through the atmosphere, cools, and eventually leads to rainfall or other forms of precipitation.

3.1.1 The Seven Basic Equations in Weather Forecasting Models

In the numerical weather prediction model, Seven Basic Equations represent the dimensions of atmospheric motion, energy and moisture over time and space. These equations are concurrently computed in numerical weather forecasting across numerous atmospheric data points. The initial three equations are the Reynolds Equations of the Motion, which focus on changes in wind speed in various directions. The fourth equation, the Continuity or Mass Conservation Equation, determines density change rates based on density advection in various directions and velocity shifts in those directions. The Fifth equation, the Thermodynamic Energy Equation, sums up the atmosphere's internal molecular energy and the kinetic energy responsible for wind. The Thermodynamic Energy Equation summarises the atmosphere's internal molecular energy and the kinetic energy responsible for wind. The last equation is the ideal gas law [39].

All the equations below are defined in the spherical coordinates(x-y-z), where west-to-east direction (positive u direction), south-to-north direction (positive v direction), and down-to-up direction (positive w direction).

u is the west-to-east wind velocity component, v is the south-to-north component, and w is the down-to-up component. ρ – the density of the fluid. p – the atmospheric pressure. f – the Coriolis parameter. F_x and F_y are the impact of friction in x and y direction. T – air Temperature. C_p – the specific heat at constant pressure. R_d – dry gas constant.

$$\frac{\partial u}{\partial t} = -u\frac{\partial u}{\partial x} - v\frac{\partial u}{\partial y} - w\frac{\partial u}{\partial z} - \frac{1}{\rho}\frac{\partial p}{\partial x} + fv + F_x \qquad \text{(Wind Forecast in west-to-east)} \tag{3.1a}$$

$$\frac{\partial v}{\partial t} = -u\frac{\partial v}{\partial x} - v\frac{\partial v}{\partial y} - w\frac{\partial v}{\partial z} - \frac{1}{\rho}\frac{\partial p}{\partial y} - fu + F_y \qquad \text{(Wind Forecast in south-to-north)} \tag{3.1b}$$

$$\frac{\partial w}{\partial t} = -u\frac{\partial w}{\partial x} - v\frac{\partial w}{\partial y} - w\frac{\partial w}{\partial z} - \frac{1}{\rho}\frac{\partial p}{\partial z} + (2\Omega\cos(\phi))u + F_z - g \qquad \text{(Wind Forecast in down-to-up)} \tag{3.1c}$$

$$\frac{\partial \rho}{\partial t} = -u\frac{\partial \rho}{\partial x} - v\frac{\partial \rho}{\partial y} - w\frac{\partial \rho}{\partial z} - \rho\frac{\partial u}{\partial x} - \rho\frac{\partial v}{\partial y} - \rho\frac{\partial w}{\partial z} \qquad \text{(Continuity equation)} \tag{3.1d}$$

$$\frac{\partial T}{\partial t} = \frac{1}{C_p}\frac{\partial q}{\partial t} + \frac{1}{\rho C_p}\frac{\partial p}{\partial t} - u\frac{\partial T}{\partial x} - v\frac{\partial T}{\partial y} - w\frac{\partial T}{\partial z} \qquad \text{(Thermodynamic Energy Equation)} \tag{3.1e}$$

$$\frac{\partial q}{\partial t} = -u\frac{\partial q}{\partial x} - v\frac{\partial q}{\partial y} - \omega\frac{\partial q}{\partial z} + ET - P \qquad \text{(Moisture Conservation Equation)} \tag{3.1f}$$

$$p\alpha = R_d T \qquad \text{(Ideal Gas Law0} \tag{3.1g}$$

While it is impractical to simulate the entire atmospheric field using an atmospheric motion equation fragment, certain individual equations can still serve as supplementary constraints to generate precipitation fields and distinguish between real and fake precipitation maps. Specifically, Equation (3.1f) delineates the interplay among the moisture content in the air, evaporation, and precipitation.

The evapotranspiration rate (ET) could be estimated by the Makkink Equation (3.8) [40], which is based on temperature and solar radiation data and provides precision results in cold and humid climates.

$$ET = 0.65\frac{\Delta}{\Delta + \gamma}\frac{R_s}{\lambda} \tag{3.2}$$

Therefore, the combination of the Equation (3.8) and Equation (3.1f) is:

$$\frac{\partial q}{\partial t} = -u\frac{\partial q}{\partial x} - v\frac{\partial q}{\partial y} - \omega\frac{\partial q}{\partial z} + (0.65\frac{\Delta}{\Delta + \gamma}\frac{R_s}{\lambda}) - P \tag{3.3}$$

— **u**: East-west wind component. Unit: m/s

— **v**: South-north wind component. Unit: m/s

— $\boldsymbol{\omega}$: Vertical wind component. Unit: m/s

— **T**: Temperature. Unit: °C

— **q**: Specific Humidity. Unit: $g\,g^{-1}$

— $\boldsymbol{\Delta}$: Derivative w.r.t. temperature of the saturation vapour pressure

— $\boldsymbol{R_s}$: Global radiation. Unit: J/m^2

— $\boldsymbol{\gamma}$: Psychrometric constant.

— $\boldsymbol{\lambda}$: Latent heat of vaporization. Unit: J/Kg

In order to calculate the Equation (3.11), the specific humidity q, east-west wind component u, south-north wind component v, derivative w.p.t temperature of the saturation vapour pressure Δ, Psychrometric constant γ and Latent heat of vaporization λ need to be calculated. Automatic weather stations from KNMI measure global radiation.

3.1.2 The wind component

The wind component u and v are calculated by:

$$u = FF \times \cos(\theta) \tag{3.4a}$$
$$v = FF \times \sin(\theta) \tag{3.4b}$$
$$\theta = 270 - DD \tag{3.4c}$$

FF : Wind speed from AWS data. Unit: m/s

DD : Wind direction from AWS data. Unit: °C

3.1.3 The specific humidity

The specific humidity could be calculated by the combination of the Equation (3.5) , Equation (3.7) and Equation (3.6) [39].

The specific humidity is defined as:

$$q = \frac{e\epsilon}{p} \tag{3.5}$$

e : Vapor Pressure. Unit: Pa

p : air Pressure from AWS data. Unit: Pa

$\boldsymbol{\epsilon}$: dimensionless ratio of the molecular weight of water vapor to that of dry air. (0.622)

According to Clausius–Clapeyron Equation, the vapor pressure is defined as:

$$e = 611\,\mathrm{Pa} \cdot \exp\left(\frac{\lambda M_v}{R^*}\left(\frac{1}{273\,\mathrm{K}} - \frac{1}{T_d}\right)\right) \tag{3.6}$$

$\boldsymbol{M_v}$: the molecular weight of water vapor. (0.018015 kg mol^{-1})

$\boldsymbol{R^*}$: the universal gas constant. $R = 8.314$ J mol^{-1}K^{-1}

$\boldsymbol{T_d}$: Dew Point Temperature from AWS data. Unit: Kelvin

Latent Heat of Vaporization λ is defined as:

$$\lambda = (2.501 - (2.361 \times 10^{-3})T) \times 10^6 \tag{3.7}$$

T : Air Temperature from AWS data. Unit: °C

3.1.4 Makkink equation

$$ET = 0.65 \frac{\Delta}{\Delta + \gamma} \frac{R_s}{\lambda} \tag{3.8}$$

In order to calculate the evapotranspiration rate from Makkink Equation [41], the Latent Heat of Vaporization and the Derivative w.p.t temperature of the saturation vapour pressure need to be calculated from the following equations.

The Derivative w.p.t temperature of the saturation vapour pressure Δ is calculated by:

$$\Delta = \frac{4098 e^o(T)}{(T + 237.3)^2} \tag{3.9}$$

where e^0 is the saturation vapor pressure is given by (3.6) but replace T_d by air pressure. where the unit is Pa/°C

The Psychrometric Constant is calculated by:

$$\gamma = \frac{c_p p}{\varepsilon \lambda} \times 10^{-3} = 0.00163 \frac{p}{\lambda} \tag{3.10}$$

$\boldsymbol{C_p}$: specific heat of moist air. 1.013×10^3 (J kg^{-1}°C^{-1})

3.1.5 Simplified version of Moisture Conservation Equation

By computing all parameters outlined in Equation (3.11), it can be effectively resolved and transformed into the following format:

$$\mathcal{R}_q = \frac{\partial q}{\partial t} - u\frac{\partial q}{\partial x} - v\frac{\partial q}{\partial y} - \omega\frac{\partial q}{\partial z} + (0.65 \frac{\Delta}{\Delta + \gamma} \frac{R_s}{\lambda}) - P \tag{3.11}$$

Measuring vertical wind speeds ($\boldsymbol{\omega}$) presents significant challenges due to their typically low intensity, the need for sensitive equipment, atmospheric stability factors, and the complexity and expense of precise measurement methods. Consequently, no existing datasets include vertical wind speed measurements. The most straightforward approach is to omit the term $\omega\frac{\partial q}{\partial z}$. However, neglecting the full three-dimensional dynamics of the atmosphere risks overlooking moisture transport occurring across different atmospheric layers, potentially complicating maintaining moisture balance.

To address this issue, the model introduces an assumption focusing on the horizontal wind components, denoted U (east-west direction) and V (north-south direction), at different elevation levels. The ERA5 dataset provides measurements of these horizontal wind components, specifically u_{100} and v_{100}, at a height of 100 meters. Similarly, Automatic Weather Stations (AWS) measure these components at a lower altitude, precisely at 10 meters. In order to approximate the three-dimensional moisture balance within the atmosphere effectively, a strategic decision was made to utilize data from these two distinct altitudinal points: 10 meters and 100 meters.

After implementing this assumption to account for the variations in altitudes in the atmospheric moisture analysis, the equation is modified to incorporate the influence of horizontal wind components at 10-meter and 100-meter altitudes, thereby providing a more comprehensive approximation of moisture transport without direct measurements of vertical wind speeds.

$$\mathcal{R}_q = -\frac{\partial q}{\partial t} - u_{10}\frac{\partial q}{\partial x} - v_{10}\frac{\partial q}{\partial y} - u_{100}\frac{\partial q}{\partial x} - v_{100}\frac{\partial q}{\partial y} + (0.65\frac{\Delta}{\Delta + \gamma}\frac{R_s}{\lambda}) - P \tag{3.12}$$

where the Equation (3.12) is computed at the pixel level. By incorporating the horizontal wind components, u and v, at both 10 meters and 100 meters, the revised equation offers a more nuanced representation of moisture dynamics across these two distinct atmospheric layers. This modification allows the equation to adapt to variations in altitude, resulting in a model that more accurately reflects the three-dimensional moisture balance within the atmosphere. However, it's important to note that atmospheric interactions, which significantly influence weather patterns, and extend up to the end of the troposphere, approximately 10 km in altitude. Therefore, by focusing on wind speed within the lower 100 meters, the approach inevitably entails a degree of uncertainty, given the comprehensive atmospheric interactions occurring beyond this range.

The Equation (3.12), referred to as the Simplified Version, is employed to assess the accuracy of precipitation predictions in adherence to the physical laws governing moisture movement. This approach addresses the complexities of atmospheric moisture transport and enhances the reliability of weather forecasting models by incorporating a detailed analysis of horizontal wind components at varying elevations.

Method 4

4.1 Proposed Model

This section begins with a comprehensive background of the proposed model's framework. Following this, the model's three distinct phases are elaborately discussed in the following subsections.

4.1.1 Background

Generative Adversarial Networks (GANs) operate through a dual-component framework comprising a generator (G) and a discriminator (D) [15]. The generator's function is to synthesize data, such as images, from a noise prior, denoted as $p_z(z)$. It aims to create outputs indistinguishable from actual data. Meanwhile, the discriminator examines these outputs to determine their authenticity, acting as a judge to differentiate between real data and the generator's fabrications. This setup creates a competitive and iterative learning process, with the generator improving its ability to mimic real data, while the discriminator enhances its skill in detecting forgeries. The generator, through a process aiming to minimize $\log(1 - D(G(z)))$, seeks to produce outputs that the discriminator will misclassify as real. On the other hand, the discriminator is trained to maximize its accuracy in distinguishing between actual data and the generator's outputs. The adversarial process propels both models to continuously improve, with the ultimate goal of reaching a point where the generator's outputs are so convincing that the discriminator is left at a threshold of uncertainty, unable to distinguish synthetic data from real samples. This dynamic interplay is the driving force behind the adversarial training methodology, pushing the boundaries of what is achievable in synthetic data generation and evaluation.

The discriminator (D) and the generator (G) are involved in a dual minimax loss function $V(G, D)$ as follows:

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \sim p_{data}(x)}[\log D(x)] + \mathbb{E}_{z \sim p_z(z)}[\log(1 - D(G(z)))]. \tag{4.1}$$

GAN models have found extensive applications across diverse fields, showcasing their versatility and efficacy. In computer vision, they are pivotal for tasks like image synthesis [42]–[44] , video generation [45], language generation [46], data augmentation [47] and weather nowcasting [17], [18], [48].

In the context of GAN applications for weather nowcasting, Jing et al. [17]developed an innovative deep learning model known as the Adversarial Extrapolation Neural

Network (AENN), which addresses the problem of blurry predictions prevalent in deep learning extrapolation models. The AENN model comprises three key elements: a conditional generator, a temporal discriminator, and a spatial discriminator. The conditional generator, incorporating an encoder, ConvLSTM, and a decoder, encodes the spatial features of the input and uses ConvLSTM to model the temporal dynamics, facilitating future frame predictions. The temporal discriminator identifies discrepancies between the predicted extrapolations and actual sequences, while the spatial discriminator focuses on differentiating between actual and predicted echo frames.

AENN outperforms other models, such as ConvLSTM and Optical flow, on the categorical score(POD, CSI and HSS). Besides, effectively addressing the issue of blurry prediction produced more accurate and realistic extrapolation echoes with internal details and accurately modelled echo evolution. Moreover, Koert Schreurs [49] conducted tests on AENN for nowcasting applications, demonstrating that it yielded greater accuracy compared to optical flow techniques like PySTEPS.

Additionally, Haoran applied the 'Nuwä' model to the task of precipitation nowcasting. This model is a robust, multimodal, pre-trained model, skilled in creating and altering visual media, including a variety of images and videos for diverse visual synthesis tasks, as described in [50]. According to the findings in [28], the combined model of Nuwä and extreme value loss (EVL) has proven to be more effective than optical flow methods like PySTEPS in terms of accuracy and predictive performance. This is particularly notable in predicting precipitation nowcasting and forecasting extreme weather events.

However, deep learning models such as AENN and 'Nuwä' exhibit several limitations in weather forecasting. Their performance notably decreases with prolonged forecast periods, impacting the accuracy of long-term predictions. Nonetheless, these models have many limitations, for instance, they are not able to produce consistent prediction over medium to long-term horizons and struggle with extreme events modelling and prediction. They also need help generalizing to various weather conditions and geographical areas not adequately represented in their training datasets. A prevalent issue with sophisticated neural networks like AENN is their lack of interpretability, which obscures the reasoning behind their predictions and makes it difficult for experts to understand their decision-making process. Additionally, these models need to consistently adhere to the fundamental physical principles governing the systems they are designed to simulate. Balancing the physical accuracy of meteorological phenomena with the models' predictive capabilities remains a complex task. Furthermore, specific models like the Nuwä, especially in precipitation forecasting, are impeded by lengthy processing times, taking up to 20 minutes for a single event prediction. Which is a considerable drawback in scenarios requiring rapid forecasting.

In response to these critical issues, researchers have been working to devise innovative and efficient methods to integrate domain expertise and fundamental physical laws into machine learning models. This effort has led to the rise of physics-informed machine learning (PIML) [51], [52], a new field that blends traditional scientific principles with advanced computational techniques.

In the previous work ”Physics-informed machine learning: case studies for weather and climate modelling” [21], a comprehensive analysis is conducted on the integration of physics and domain expertise into machine learning (ML) models, a process which substantially augments their functionality. This research categorizes various methods of integration and illustrates their applications through ten detailed case studies in the realms of emulation, downscaling, and weather and climate forecasting. These studies collectively underscore significant advancements: they bolster the physical coherence and scientific validity of predictive models, enhance efficiency by reducing the data required for effective training, expedite the training process, and amplify the models' capacity to generalize, thus ensuring their robustness in diverse and dynamically changing scenarios, including those impacted by climate change. Crucially, this fusion of physics and machine learning also enhances the transparency and interpretability of these models, thereby increasing confidence in their predictive outputs and broader applications within the scientific community.

Building on the concept of physics-informed deep learning (PIDL), a novel framework known as PID-GAN is introduced, aimed at enhancing uncertainty quantification (UQ) in deep learning (DL) applications within critical scientific fields [27]. The burgeoning necessity of integrating UQ with DL in scientific contexts underscores the importance of this development. PID-GAN ingeniously incorporates physical laws into the learning process of both the generator and discriminator models in a Generative Adversarial Network (GAN) setting. A standout feature of PID-GAN is its ability to maintain balance in generator gradients across multiple loss terms, an issue prevalent in existing methodologies. The effectiveness of PID-GAN has been empirically validated through various case studies, including physics-based Partial Differential Equations (PDEs) and scenarios with imperfect physics. This framework represents a significant stride in the field, ensuring physically consistent and generalized solutions while effectively performing UQ.

The utilization of Equation (3.12) serves as a critical tool for assessing the accuracy of precipitation forecasts, particularly in alignment with the physical principles that dictate moisture movement. This approach represents an example of integrating imperfect physics into predictive modelling. Consequently, the proposed model amalgamates two distinct yet complementary frameworks: the VQGAN + Transformer and PID-GAN. The VQGAN + Transformer, known for its proficiency in generating high-fidelity visual representations [26], is coupled with PID-GAN, which infuses physical laws into the learning process of both its generator and discriminator[27]. This synergistic combination leverages each framework's strengths, ensuring the generation of visually accurate precipitation forecasts and their adherence to fundamental physical principles, particularly those governing atmospheric moisture dynamics.

4.1.2 Model

In the proposed architecture of our model, as illustrated in Figure 4.1, a sophisticated integration of components is presented, central to which is the generator. This generator is ingeniously designed as a hybrid of VQ-GAN and Transformer networks. This unique combination leverages the VQ-GAN's capabilities in generating high-quality, detailed images through vector quantization while incorporating the Transformer's adeptness at capturing complex, long-range dependencies within data sequences. Consequently, the generator emerges as a powerful tool for synthesizing visually compelling and contextually coherent imagery. Complementing this, the model features two discriminators: a temporal discriminator and a spatial discriminator, each tasked with assessing different aspects of the generated output. The spatial discriminator, operating within the VQ-GAN framework, evaluates the fidelity and authenticity of individual images. In contrast, the temporal discriminator scrutinizes the sequential coherence of images over time, ensuring that the generated sequences are not only realistic frame-by-frame but also maintain a logical progression. An innovative aspect of the model is the integration of physical consistency scores, derived from physical data, which are then concatenated with radar image samples. This enriched information is fed into the temporal discriminator, enhancing the model's input with real-world physical constraints and thereby augmenting the temporal discriminator's ability to judge the physical plausibility of generated sequences.

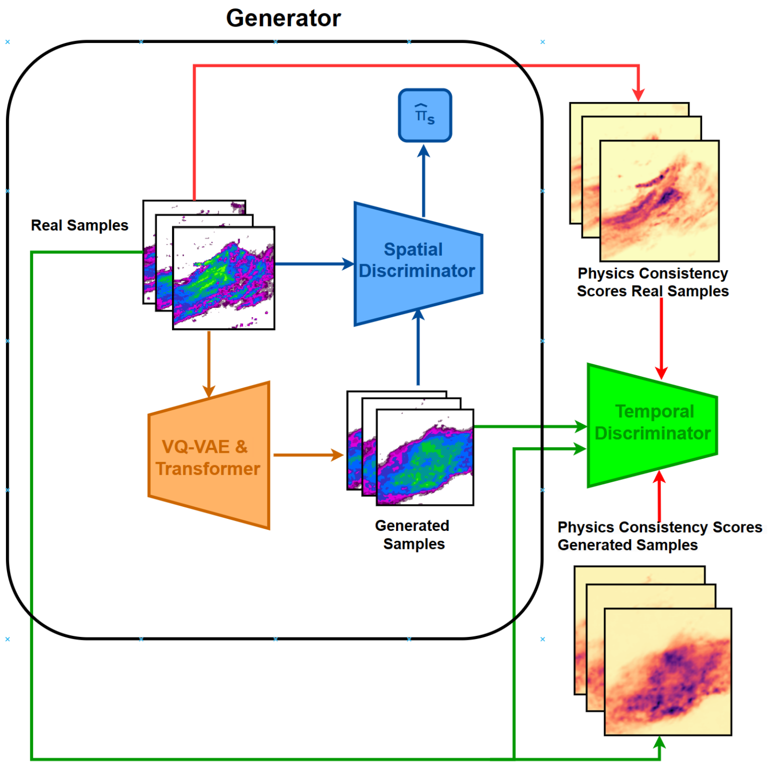


Figure 4.1: Proposed model structure

4.1.2.1 Generator

The generation model utilizes a deep generative model newly created for tasks related to the High-Resolution Image Synthesis [26]. The approach operates in two phases: initially, it learns to encode the data, and subsequently, in the second phase, it acquires a probabilistic model based on this encoding. As discussed in [53], this concept uses a Variational Autoencoder (VAE) to capture the data representation in the first stage [54], [55]. Afterwards, it involves applying a second VAE to understand and model the probabilistic distribution of the resulting data representation [56], [57]. Besides, the proposed model uses the Vector Quantization Generative Adversarial Network(VQ-GAN) [26], an approach to learning discrete representations of images, and models their distribution autoregressively with Transformer architecture.

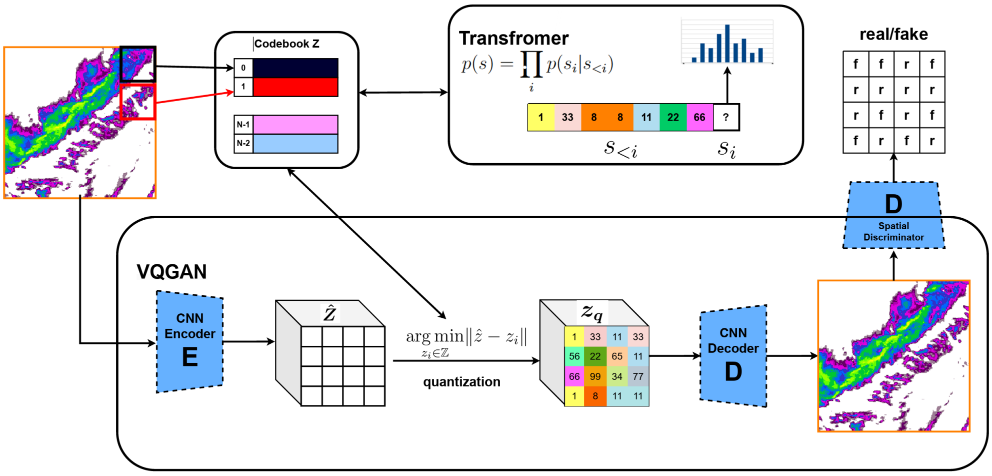


Figure 4.2: The proposed generator framework begins with the initial phase, where the VQ-GAN acquires a codebook. Each code, or an amalgamation of multiple codes, within this codebook corresponds to a specific pattern observed in the radar image. This allows the radar information to be condensed into a series of indices. These indices, in turn, are methodically modelled by the autoregressive transformer. During prediction, a set of conditional indices is fed into the transformer, which has been trained to sequentially produce probabilistic distributions for the subsequent prediction tokens in an autoregressive manner.

VQ-GAN

The initial introduction of the Vector Quantization Generative Adversarial Network(VQ-GAN) in [26] was primarily aimed at the application related to High-Resolution Image Synthesis. According to their finding, VQ-GAN can produce more realistic and high-resolution images, which is especially useful for tasks like image generation compared to the Vector Quantization Variational Autoencoder(VQ-VAE). Therefore, this model is used in our proposed model, which is used to map the original radar images into a lower-dimensional discrete latent space. This compact, meaningful latent space representation can enhance the capabilities of the subsequent transformer-based model.

The VQ-GAN model architecture, as illustrated in Figure 4.2, comprises four distinct elements: the CNN encoder, the CNN decoder, the codebook, and the PatchGAN discriminator (spatial discriminator) [58]. Additionally, the detailed model structures of the encoder and decoder are shown in Figures 4.3 and 4.4, respectively. The detailed model structure of the PatchGAN discriminator is presented in Figure 4.5.

The detailed input flow through the encoder and decoder unfolds as follows: Initially, the input passes through the CNN encoder, transforming it into a lower-dimensional latent representation. This representation is then quantized using a codebook, compressing the input data into discrete codes. Subsequently, this quantized representation is fed into the CNN decoder, which reconstructs the input data from its compressed state. This process ensures that essential information is retained while the dimensionality of the input data is reduced. After this, the reconstructed images pass through the discriminator, which divides them into patches to distinguish whether each patch is real or fake.

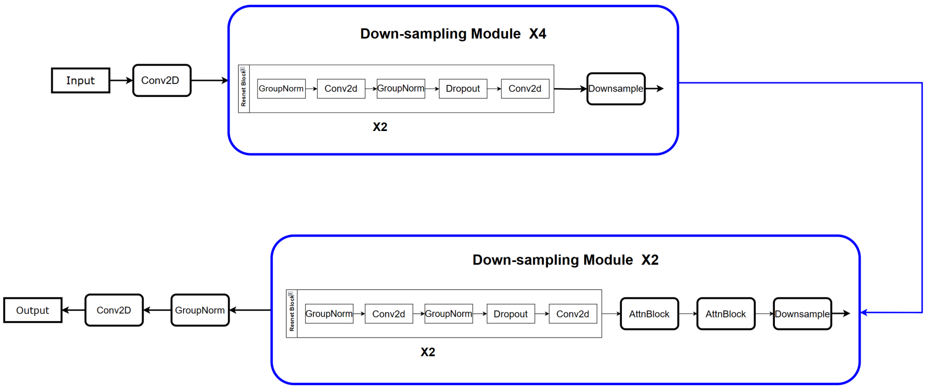


Figure 4.3: Encoder model structure

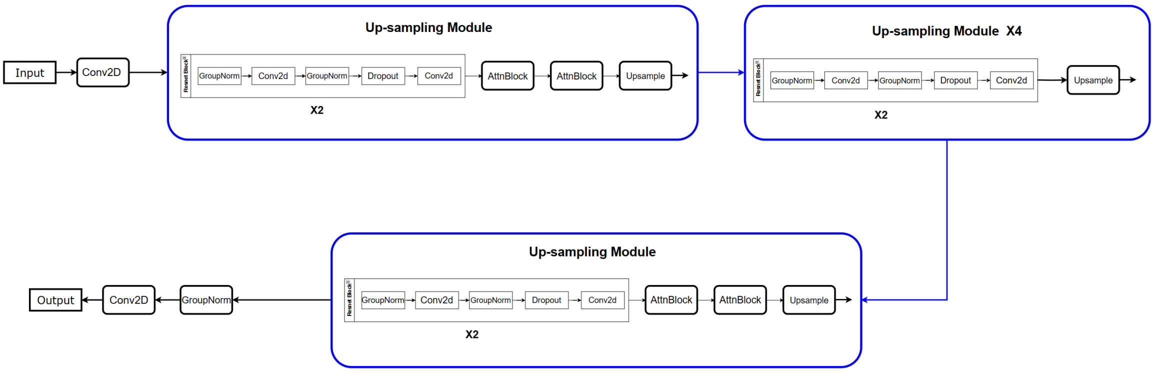


Figure 4.4: Decoder model structure

The loss function for the VQ-GAN model is described as follows, where E,G,$\mathcal{Z}$ and D represent the encoder, decoder, codebook and discriminator.

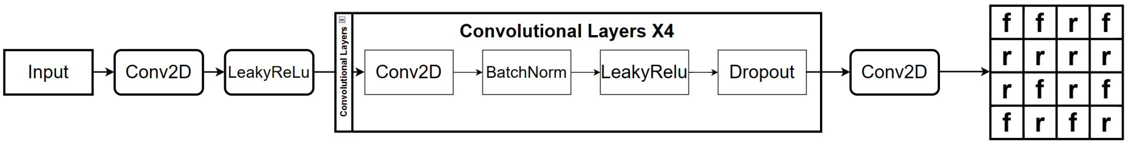


Figure 4.5: Discriminator model structure

$$\begin{aligned}\mathcal{L}(E, G, \mathcal{Z}) = \|x - \hat{x}\|_1 + \|sg[E(x)] - z_q\|_2^2 + \|sg[z_q] - E(x)\|_2^2 \\ + \mathcal{L}_{perceptual}(x, \hat{x}) + \lambda_{GAN}\mathcal{L}_{GAN}(\{E, G, \mathcal{Z}\}, D)\end{aligned} \tag{4.2}$$

where the adversarial loss $\mathcal{L}_{GAN}(\{E, G, \mathcal{Z}\}, D)$ and the adaptive weight λ_{GAN} are computed according to

$$\mathcal{L}_{GAN}(\{E, G, \mathcal{Z}\}, D) = [\log D(x) + \log(1 - D(\hat{x}))] \tag{4.3}$$

$$\lambda_{GAN} = \frac{\nabla_G [\mathcal{L}_{rec}]}{\nabla_G [\mathcal{L}_{GAN}] + \delta} \tag{4.4}$$

1. $\|x - \hat{x}\|_1$: The reconstruction loss, comparing the original input x with the reconstructed input $\hat{x}$.

2. $\|sg[E(x)] - z_q\|_2^2$: The commitment loss, where $sg[.]$ represents the stop-gradient operation. This term calculates the squared Euclidean distance between the encoder output $E(x)$ and the quantized latent space vectorz_q. It guides the encoder to produce outputs close to the discrete codes in the codebook and indirectly updates the codebook during training. It ensures that the discrete codes effectively represent the information content of the input data.

3. $\|sg[z_q] - E(x)\|_2^2$: This term calculates the squared Euclidean distance between the quantized latent space vector and the output of the encoder. It ensures the quantized latent space vector aligns closely with the encoder's output, guiding the encoder to produce representations suitable for quantization. By minimizing this term, the encoder learns to generate outputs aligned with the discrete codes.

4. $\mathcal{L}_{perceptual}(x, \hat{x})$: perceptual loss, which measures high-level differences between the original input x and reconstructed input $\hat{x}$.

5. λ_{GAN}: The adaptive weight is calculated by ∇_G, which is the gradient of the loss function concerning the last layer of the decoder.δ is a scalar value for numerical stability.

This training approach significantly reduces the input data's dimensionality, thereby allowing transformer models to be utilised in the following steps.

To enhance computational efficiency, the spatial resolution of the input radar images was reduced from 256 × 256 to 128 × 128 through downsampling. This process, as depicted in Figure 4.6, maintains the semantic integrity of the radar maps, demonstrating that critical information is preserved despite the reduction in resolution. Furthermore, this downsampling strategy contributes to more effective GPU memory management during the training of the VQ-GAN. Reconstructed precipitation fields, in comparison to their corresponding ground truth images, are showcased in Figure C.1.

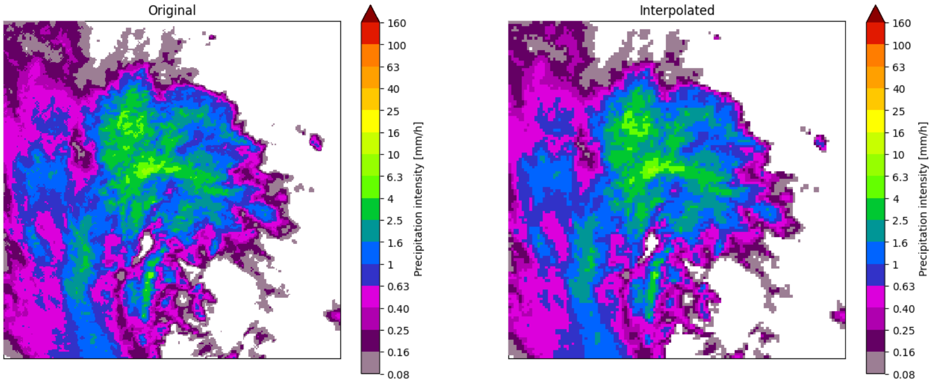


Figure 4.6: Radar image samples with different spatial resolutions, original image have resolutions 256 × 256 and Interpolated image have resolutions 128 × 128.

The VQ-GAN model configuration and input data flow regarding the changing of the dimensionality are described as follows:

1. VQ-GAN Model Configuration:
   - Vocabulary size(V): 1024
   - Token dimension(D): 2048
   - Encoded feature map dimensions[h', w']: [8,8]
2. VQ-GAN Model training Configuration:
   - Batch size(B): 128
   - Time (T): 9
   - Input image height(H):128
   - Input image weight(W):128
   - Learning Rate: 0.0001
   - Discussion of dimensionality changes.
3. Input dimension flow:
   - Codebook size: [V,D]
   - Input x size: [$B \times T$,C,H,W]
   - Encoder output size: [$B \times T$,D,h',w']
   - Token size: [$B \times T$, $h' \times w'$]
   - Decoder input size: [$B \times T$,D,h',w']
   - Reconstruction size: [$B \times T$,C,H,W]

Transformer

The transformer architecture distinctively employs attention mechanisms to model interactions among its inputs without considering the relative positions of these inputs to one another [59]. Recent studies have demonstrated that transformer technology, initially designed for natural language processing tasks [60], [61], also shows remarkable effectiveness in processing image and video data [50], [62], [63]. The architecture of a transformer comprises an encoder and a decoder, each consisting of layers that incorporate an attention mechanism, allowing inputs at different positions to interact. This setup is complemented by position-wise feed-forward networks, which process each position independently. Specifically, the self-attention mechanism within a transformer processes an intermediate representation through three position-wise linear layers, generating three sets of representations: queries (Q), keys (K), and values (V). The attention weights are calculated by Eq. 4.5 based on these components to ensure that the context of each input is comprehensively captured, irrespective of its position.

$$\mathrm{Attn}(Q, K, V) = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) V \tag{4.5}$$

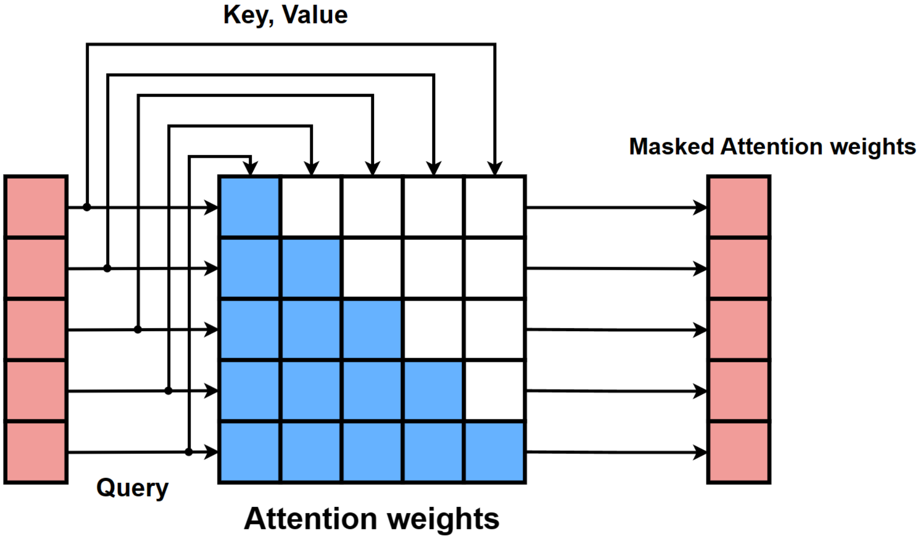


Figure 4.7: Causal Attention

Furthermore, the model integrates a causal attention mechanism, as illustrated in Figure 4.7. In this mechanism, masking is employed to guarantee autoregressive generation, ensuring the model only considers previous and current tokens when predicting sequence elements. This is operationalized by masking the upper triangular section of the attention weights matrix (indicated by the white blocks in the figure) to preclude the influence of future-position tokens. Following this, the transformer performs a linear, point-wise transformation to derive the logits that facilitate the prediction of the ensuing sequence element, thus maintaining the logical temporal ordering of the data.

As shown in Figure 4.8, The transformer utilizes multiple layers of causal self-attention, taking in the latent representation produced by the VQ-GAN as its input to learn the data distribution effectively using a cross-entropy loss function. This loss function measures the difference between the model's predicted probability distribution over the discrete latent space and the true latent representation of the observed data.

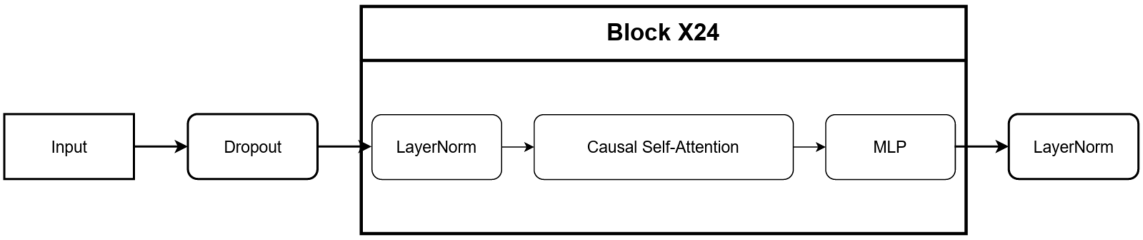


Figure 4.8: Autoregressive transformer structure

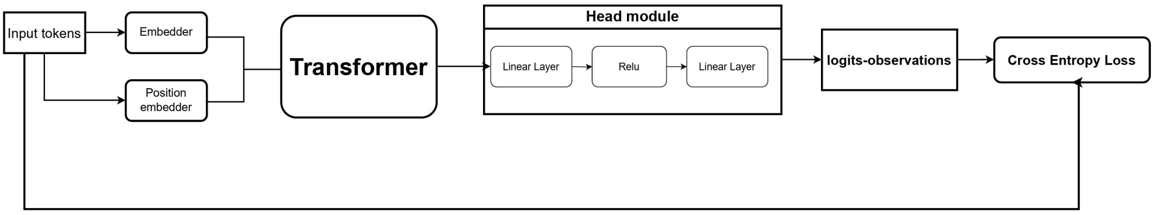


Figure 4.9: Training stage of transformer

As shown in Figure 4.9, the diagram illustrates the data flow within the transformer during the training phase. The original data undergoes quantized encoding, resulting in a format expressed as $\mathbf{z}_q = q(\mathbf{E}(x))$. This process yields a sequence $\mathbf{s}$ within the set $0, \ldots, |\mathbf{Z}| - 1^{h \times w}$, which corresponds to the indices from the VQ-GAN codebook. These indices (tokens) are then passed to an embedder that transforms the discrete tokens into continuous vectors. Positional embeddings are subsequently added, infusing the sequence with order information essential for the transformer. The transformer processes the embedded tokens, and the resulting high-level features are further refined by the head module, which outputs logits—the model's raw, unnormalized predictions for each token. The logits are evaluated against actual observations using a cross-entropy loss function, which measures the discrepancy between the predicted probabilities and the observed data's distribution. This loss informs model parameter updates through backpropagation, aiming to minimize the difference and enhance the model's predictive accuracy during training.

$$\mathcal{L}_{\text{Transformer}} = \mathbb{E}_{x \sim p(x)}[-\log p(\mathbf{z})] \tag{3}$$

where $p(\mathbf{z}) = \prod_{i=1}^{N} p(z_i|z_{<i})$, indicating that, given the indices $s_{<i}$, the transformer is trained to predict the distribution of the possible next indices z_i.

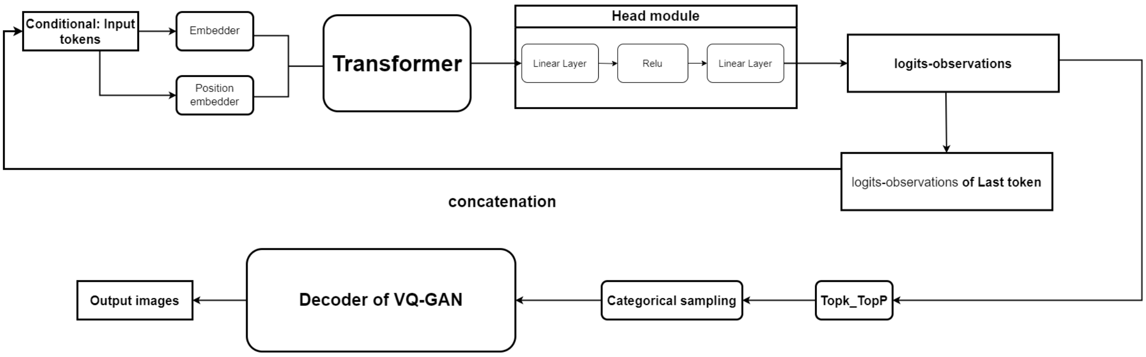


Figure 4.10: Generation stage of transformer

In the generation stage depicted in Figure 4.10, the model begins with conditional input tokens that serve as the starting point for generating new data. A Key-Value (KV) cache is employed here to expedite the process at each step [64]. By storing the keys and values calculated during the self-attention phase, the model circumvents unnecessary recalculations for these elements in subsequent steps, which is particularly advantageous in autoregressive models. As the transformer predicts one token at each step, the KV cache allows each new output token to be conditioned on previously generated tokens without recomputing the full attention map, thus significantly enhancing the efficiency of the generation process.

After adopting the same methodology as in the training stage to produce logits observations, the logits of the last token are sliced and concatenated with the previous tokens to form a new sequence, which is then processed by the transformer in subsequent steps. The transformer is designed to continue this process for a total number of steps determined by the number of prediction frames multiplied by the dimensions of the encoded feature map.

Once the logits for the entire sequence of prediction frames have been generated, top-k and top-p sampling techniques are applied to narrow down the sampling pool to the k most likely tokens or to a subset of tokens that together add up to a specified probability p, enhancing the quality and coherence of the generated data.

For the final output, categorical sampling is used, which selects from the logits distribution to produce the final token in the sequence. This sequence of tokens is then passed to the decoder of the VQ-GAN, translating the discrete sequence back into a continuous representation and culminating in the predicted images.

Physics-Informed Discriminator(PID)-GAN

The Physics-Informed Discriminator (PID) formulation is introduced[27], which incorporates principles of physics directly into the adversarial training process, as illustrated in Figure 4.11. An additional metric, known as the physics consistency score (η), is calculated for each prediction to evaluate its adherence to physical laws. These scores are subsequently integrated as supplemental inputs into the discriminator. This innovative approach enables the discriminator to distinguish between real and fake samples by learning from the data distribution and enhancing its discernment capabilities through additional physics-based supervision.

The physics consistency score for each prediction concerning the physical constraint is calculated based on the following equation:

$$\eta_k = e^{-\lambda \mathcal{R}^{(k)}(x, \hat{x})} \tag{4.6}$$

where $R(x, x)$ is the raw output of the physical equation (3.12).

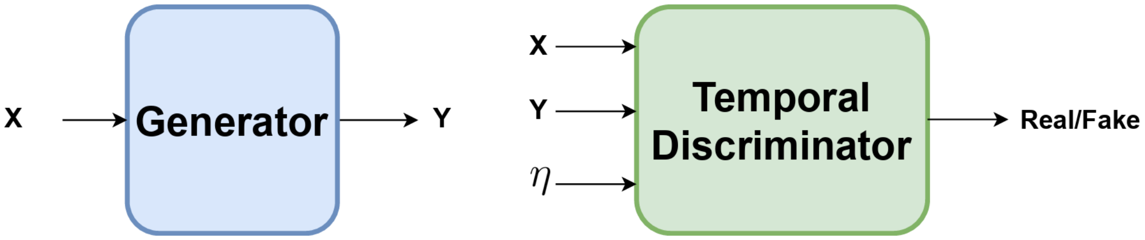


Figure 4.11: PID-GAN model structure

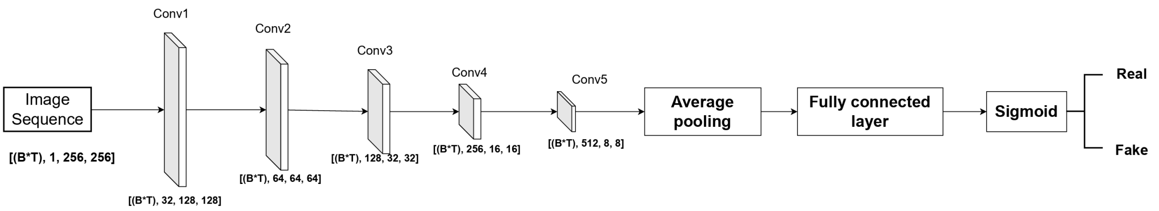


Figure 4.12: Temporal Discriminator structure

To frame the model within the GAN architecture, a specialized discriminator known as the temporal discriminator is incorporated, as proposed by [17] and depicted in Figure 4.12. This temporal discriminator is tailored for sequences, distinguishing between real and generated sequences by analyzing their temporal properties. Its architecture comprises a series of convolutional layers that lead to a single fully-connected layer. The output of this fully-connected layer is then converted into a probability score through the application of a sigmoid activation function, indicating the likelihood that the input sequence is genuine.

Since the Equation (3.12) described in Chapter 3 is the imperfect physics equation. Therefore, follow the case study in [27], the objective for the generator and discriminator are follows:

$$\mathcal{L}_G(\theta) = -\frac{1}{N}\sum_{i=1}^{N} D(x_i, \eta_i) \tag{4.7}$$

The Equation (4.7) denotes the evaluations made by the Physics-Informed Discriminator on the predicted data samples. The generator aims to reduce these scores, thereby deceiving the discriminator into classifying the synthetically generated predictions $\hat{x}_i$ as authentic.

$$\mathcal{L}_D(\phi) = -\frac{1}{N}\sum_{i=1}^{N} \log(D(x_i, \eta_i)) - \frac{1}{N}\sum_{i=1}^{N} \log(1 - D(\hat{x_i}, \eta_i')) \tag{4.8}$$

The symbol n_i' represents the physics consistency score for ground truth data, which ensures it aligns with physical laws. This approach blocks the generator from simply generating samples that match physical constraints without true understanding. Instead, the generator is guided to reproduce the physics consistency scores observed in the real samples.

4.2 Managing extreme precipitation events

The latent representation of the data corresponds to various levels of precipitation intensity, ranging from none to extreme rain. However, tokens indicating no or light rain are far more prevalent than those representing heavy or extreme rain. Consequently, the proposed model is trained on a dataset exhibiting significant imbalance. Due to the lack of training samples for extreme precipitation events, the model's accuracy in predicting these severe weather occurrences is diminished. Consequently, the model's output distribution favours more common tokens, such as those indicating no or light rain.

To address this problem, two post-processing methods are used in this project.The first post-processing method aims to highlight high precipitation pixels [65]. It involves manipulating processed (TP) and unprocessed (RP) predictions. Parameters a and b are optimized to maximize the Gilbert Skill Score (GSS) on the validation set, with specific values of 0.66 for a and 0.81 for b in this project. While this method effectively enhances the detection rate of high precipitation pixels, it also increases false alarm cases.

$$TP_j[i] = \left[1 + a\left(\frac{RP_j[i]}{\max(RP_j)}\right)^b\right] \cdot RP_j[i] \tag{4.9}$$

where i is coordinate matrix referencing geographic positions, j is the time sample.

The second post-processing method outlined in Pysteps [6] is designed to non-parametrically adjust a prediction array so that its empirical cumulative distribution function (CDF) aligns with that of a ground truth array. This method ensures the preservation of the relative ordering of values within the prediction array while adjusting its distribution to match that of the ground truth array.

$$R'(x,y) = F_{\text{obs}}^{-1}(F(R(x,y))), \tag{4.10}$$

where F_{obs} represents the cumulative distribution function (CDF) of the observed data, and F signifies the CDF of the input forecast field R.

4.3 Evaluation metrics

This section introduces the evaluation metrics employed in the experiments for nowcasting the performance of the entire study area and the extreme event detection within 12 specific Dutch catchments.

4.3.1 Pearson's Correlation Coefficient (PCC)

Pearson's Correlation Coefficient (PCC) is a statistical measure that calculates the linear relationship between two variables. It ranges from -1 to 1, where 1 indicates a perfect positive linear relationship, -1 indicates a perfect negative linear relationship, and 0 suggests no linear correlation between the variables.

$$PCC = \frac{1}{N_f} \sum_{i=1}^{N_f} \left(\frac{F_i - \mu_F}{\sigma_F} \right) \left(\frac{O_i - \mu_O}{\sigma_O} \right) \tag{4.11}$$

In Equation (4.11) , F_i and O_i represent the measured rainfall amounts at a specific point on the predicted and observed radar maps. The terms μ_F and μ_O denote the average rainfall amounts across the entire predicted and observed radar maps, while σ_F and σ_O are the corresponding standard deviations of this rainfall. N_f is the total count of pixels in the radar map projection for a given forecast interval.

4.3.2 Mean absolute error (MAE)

$$MAE = \frac{\sum_{i=1}^{N_f} |F_i - O_i|}{N_f} \tag{4.12}$$

The MAE is a widely utilized and simple metric for evaluating nowcasting tasks. Lower values of MAE are indicative of better outcomes.

4.3.3 Critical success index (CSI)

To determine categorical metrics, each pixel in the prediction and observation maps is initially categorized as positive (greater than or equal to) or negative (less than) according to a specified threshold as shown in Table 4.1.

Critical Success Index (CSI) is a well-regarded measure used in nowcasting to evaluate the performance of binary classifications, considering both the accuracy of positive predictions and the cost of false alarms. A higher CSI value is indicative of improved performance.

	Predicted Positive	Predicted Negative
Actual Positive	Hits (True Positive)	Misses (False Negative)
Actual Negative	False Alarms (False Positive)	Correct Negatives (True Negative)

Table 4.1: Confusion Matrix

$$CSI = \frac{H}{H + F + M} \tag{4.13}$$

4.3.4 False alarm ratio (FAR)

The False Alarm Ratio (FAR) is another crucial metric for assessing binary classification performance and is frequently utilized in meteorological forecasting. It measures the precision of predictive alarms. A reduced FAR indicates superior performance.

$$FAR = \frac{F}{F + H} \tag{4.14}$$

4.3.5 Fractions skill score (FSS)

The Fraction Skill Score (FSS) is a spatial verification metric to evaluate the ability of a model to predict precipitation at various scales, given a specific threshold. In the FSS calculation, a window of size n is slid over the image to determine the fraction of pixels exceeding the predetermined threshold. MSE(n) represents the mean square error between the ground truth and predicted values at the scale n. The reference MSE is defined as the maximum MSE that can be observed and predicted at the same scale n.

$$FSS = 1 - \frac{MSE(n)}{MSE_{ref}(n)} \tag{4.15}$$

$$MSE_{ref}(n) = \frac{1}{N_x N_y}\left[\sum_{i=1}^{N_x}\sum_{j=1}^{N_y} O_{i,j}^2(n) + \sum_{i=1}^{N_x}\sum_{j=1}^{N_y} F_{i,j}^2(n)\right] \tag{4.16}$$

Predictions are typically deemed skillful when the Fraction Skill Score (FSS) exceeds the value of $0.5 + \frac{1}{f0}$, where f0 represents the domain-averaged fraction of rainfall as observed across the area of interest.

4.3.6 Hit Rate(HR)

The hit rate is also referred to as sensitivity and recall, a metric indicating a model's capability to identify events correctly. It solely assesses the probability of successfully detecting events.

$$HR = \frac{H}{H + M} \tag{4.17}$$

4.3.7 False Alarm Rate (FA)

The False Alarm Rate (FA) indicates the frequency of false alarms surpassing a specified threshold. The False Alarm Rate (FA) varies between 0 and 1, where 0 represents an ideal scenario with no false alarms and 1 signifies the least favourable situation with all alarms being false.

$$FA = \frac{F}{F + R} \tag{4.18}$$

4.3.8 Receiver Operating Characteristic (ROC) Curve

The ROC curve is a visual representation used to assess the capability of a binary classifier to identify outcomes across various thresholds correctly. The x-axis represents the False Alarm Rate (FA), and the y-axis corresponds to the Hit Rate(HR). The Area beneath the ROC curve is known as the Area Under the Curve (AUC), which is frequently employed to evaluate and contrast the classification proficiency of different machine learning models. The AUC value spans from 0 to 1, with a value of 0.5 indicating a classifier performing at the chance level. A higher AUC value is preferred, indicating a better classification performance.

4.3.9 Precision-Recall Curves

Precision-Recall curve is a visual representation used to evaluate the performance of a binary classifier, especially when the classes are imbalanced. Unlike ROC curves, which plot the HR against the FA, Precision-Recall curves focus on the relationship between Precision and Recall for different threshold settings.

$$\text{Precision} = \frac{H}{H + F} \tag{4.19}$$

Precision, as shown in Equation (4.19), indicates the accuracy of the classifier when it predicts a positive class, whereas Recall, also known as the hit rate, represents the classifier's ability to detect all relevant instances.

On a Precision-Recall curve, Recall is represented on the x-axis and Precision on the y-axis. A higher area under the Precision-Recall curve (AUC) signifies the better performance of the classifier in accurately predicting positive cases while minimizing false positives. A model with perfect Precision and recall would be represented as a point in the top-right corner of the graph, indicating a Precision and Recall of 1.

In situations where the positive class is infrequent or where the cost of false positives outweighs that of false negatives, Precision-Recall curves offer a more detailed assessment of a classifier's performance compared to ROC curves. The AUC for Precision-Recall curves ranges from 0 to 1, with higher values denoting greater classification accuracy and reliability.

4.3.10 Physical Consistency (MSE)

$$\begin{aligned}\text{Physical Consistency} &= \text{Mean}\left((R_q(\text{pre}) - R_q(\text{Ground}))^2\right), \\ R_q(\text{pre}) &= -\frac{\partial q}{\partial t} - u_{10}\frac{\partial q}{\partial x} - v_{10}\frac{\partial q}{\partial y} - u_{100}\frac{\partial q}{\partial x} - v_{100}\frac{\partial q}{\partial y} + ET - P_{\text{Trans}}, \\ R_q(\text{Ground}) &= -\frac{\partial q}{\partial t} - u_{10}\frac{\partial q}{\partial x} - v_{10}\frac{\partial q}{\partial y} - u_{100}\frac{\partial q}{\partial x} - v_{100}\frac{\partial q}{\partial y} + ET - P_{\text{radar}}.\end{aligned} \tag{4.20}$$

Physical Consistency is quantified as the mean squared error (MSE) between two quantities: R_q(Ground) and R_q(pre). In this context, R_q(Ground) is derived from the ground truth conditions as stated in Equation (3.12), while R_q(pre) originates from the simulated conditions as per the same equation. Physical Consistency thus measures the degree of deviation of the simulated condition from the ground condition, on average, with the squaring of differences emphasizing larger discrepancies. Furthermore, as per Equation (3.12), this metric is transformed into the standard form of MSE for analytical purposes.

5 Results

This chapter is dedicated to exploring two key experiments: the first centres on evaluating nowcasting performance across the comprehensive study area, while the second delves into assessing extreme event detection within 12 specific Dutch catchments.

The first section meticulously evaluates the nowcasting performance throughout the entire study area. This evaluation employs a comprehensive suite of established metrics for a multifaceted analysis. The metrics include the Mean Absolute Error (MAE), which quantifies the average magnitude of the errors in a set of forecasts; the Pearson Correlation Coefficient (PCC), assessing the linear correlation between observed and predicted values; the Critical Success Index (CSI) measures the proportion of correct positive forecasts out of the total number of actual and predicted events, the False Alarm Ratio (FAR) reflects the fraction of forecasted events that were incorrectly predicted as occurring out of all the forecasts of that event occurring, and the Fraction Skill Score (FSS), providing a spatial forecast verification method. Each metric offers a unique lens through which the nowcasting results are scrutinized, ensuring a robust and comprehensive performance analysis.

The second section focuses on detecting extreme events within individual catchment areas. This analysis commences by calculating the mean precipitation over three hours for each catchment, subsequently comparing these results against a predefined extreme event threshold. Following this comparison, events are systematically categorized into one of four distinct classifications: true positive, false positive, true negative, or false negative. The effectiveness and precision of the model are rigorously evaluated using a suite of binary classification metrics, each providing a unique insight into the model's predictive performance. The Hit Rate (HR) metric gauges the model's ability to predict events correctly. The False Alarm (FA) ratio measures the incidence of incorrect positive predictions against all non-event instances, indicating the model's tendency to forecast events erroneously. The Critical Success Index (CSI) offers a holistic performance indicator. The False Alarm Ratio (FAR) provides a specific lens on forecast accuracy by quantifying the proportion of predicted events that were falsely predicted. Lastly, the areas under the Receiver Operating Characteristic Curve and Precision-Recall Curve offer insights into the trade-off between sensitivity and specificity and the balance between precision and recall, respectively. Each metric provides critical insights, contributing to a nuanced assessment of the model's capability to identify and characterize extreme weather events.

5.1 Nowcasting performance on the Whole Netherlands

In this analytical segment, two principal experiments are conducted to examine the predictive modelling. The first experiment explores the effects of different module configurations on the predictive model's performance. Concurrently, the second experiment evaluates the effects of implementing averaging methods and post-processing techniques on the model's output.

The modular configurations are delineated into four distinct categories:

- PID-GAN(-PTS): Constituting the foundational setup, this model amalgamates the Vector Quantization Variational Autoencoder (VQ-VAE) with an auto-regressive transformer, serving as the initial standard for comparison.
- PID-GAN(-PT): Building upon the core structure, this model integrates a Vector Quantized Generative Adversarial Network (VQ-GAN) with an auto-regressive transformer and enhances it further by adding a spatial discriminator. This addition aims to refine the model's spatial understanding, setting a new baseline for comparison.
- PID-GAN(-P): This model enhances the baseline by incorporating a temporal discriminator alongside the VQ-GAN and the auto-regressive transformer, introducing a temporal dimension to the generative capabilities.
- PID-GAN: This model stands out for its intricate structure, merging the VQ-GAN and auto-regressive transformer with a temporal discriminator, all operating within a Generative Adversarial Network framework that a Physics-informed Discriminator augments for enhanced analytical fidelity.

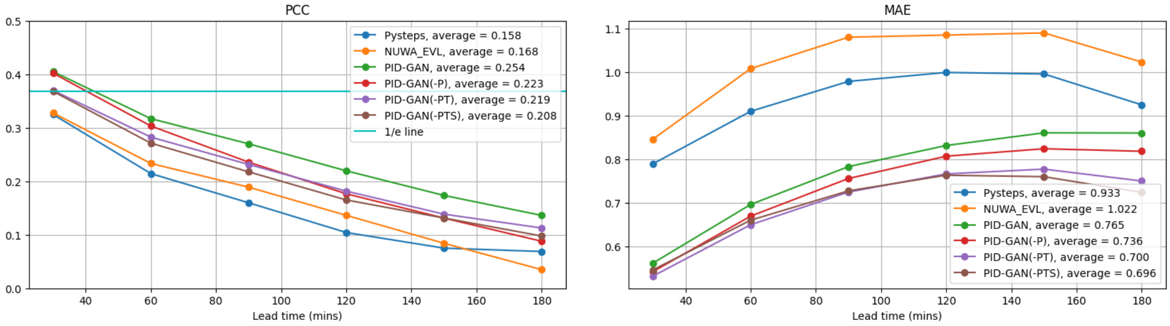
In the first experiment, the models described are assessed in comparison to two benchmarks: PySteps and NUWA-EVL[28]. The evaluation involves a range of selected metrics. Furthermore, these metrics are determined pixel-by-pixel throughout the entire study region. The test dataset spans 2019 to 2021 and includes data on 357 extreme weather events for in-depth analysis.

5.1.1 Evaluation on the different lead-time

The relationship between metric scores and lead time is depicted in the figures referred to as Figures 5.1, 5.2, and 5.3. Generally, as the lead time increases, the accuracy of nowcasting tends to decrease.

Figure 5.1 illustrates the model performance concerning continuous metrics, including Pearson's Correlation Coefficient (PCC) and Mean Absolute Error (MAE).

According to Germann [66] and Berenguer [67], understanding the lead time threshold for accurate forecasting is essential from an end-user perspective. This threshold, determined by the Pearson Correlation Coefficient (PCC), is set at the $1/e$ line. A



(a) PCC

(b) MAE

Figure 5.1: Evaluation of the 3-hour prediction: (continuous metrics) Subfigure (a) presents the Pearson Correlation Coefficient (PCC), and Subfigure (b) displays the Mean Absolute Error (MAE). This figure illustrates how the metrics vary with the forecasting lead time and includes a legend showing average scores over 3 hours.

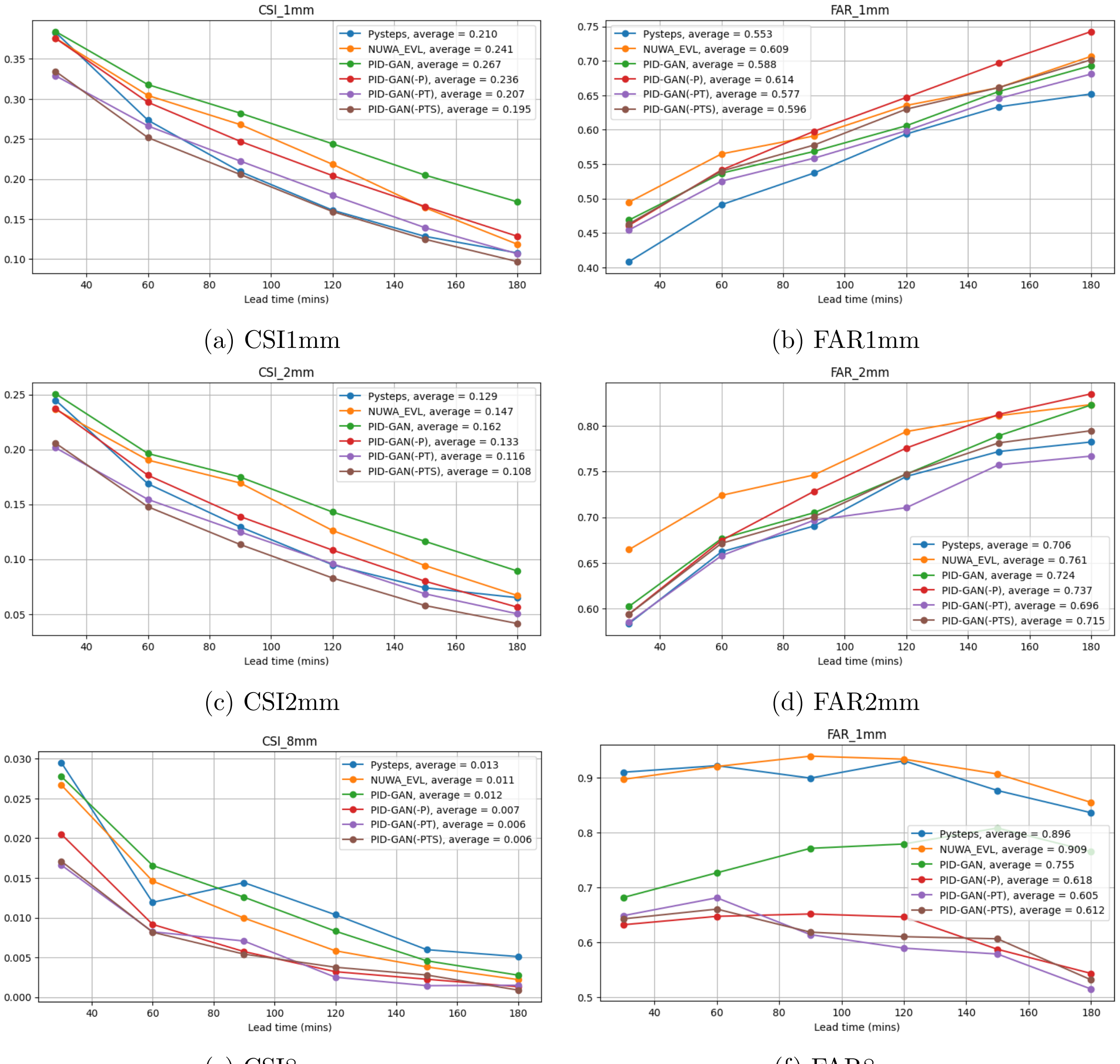
forecast is considered adequate only if its correlation falls below this threshold. Consequently, the specific lead time at which a forecast's correlation diminishes to this level is identified as the forecast's decorrelation time.

For the Pearson Correlation Coefficient (PCC) outcomes, a consistent decline in the performance of all models is observed as the lead time increases. The PID-GAN model outperforms other models for all lead times. Furthermore, only the PID-GAN, PID-GAN(-P), PID-GAN(-PT) and PID-GAN(-PTS) models achieve skilful nowcasting based on the threshold mentioned above, with the PID-GAN having a decorrelation time of around 45 minutes, the PID-GAN(-P) model around 40 minutes, and the PID-GAN(-PT) and PID-GAN(-PTS) model around 30 minutes.

In terms of Mean Absolute Error (MAE), values consistently increase across all models as the lead time extends. Furthermore, instances of more intense rainfall are associated with higher MAE values. High MAE in precipitation nowcasting is typically due to overestimation rather than underestimation. As a result, the PID-GAN, PID-GAN(-P), PID-GAN(-PT) and PID-GAN(-PTS) models exhibit much lower MAE values than Pysteps, indicating more accurate predictions. Conversely, the NUWA-EVL model displays the highest MAE, suggesting a tendency to overestimate exceptionally high rainfall intensities, resulting in more overestimated pixels.

Figure 5.2 depicts the performance of various models in terms of categorical scores across different rainfall thresholds: 1mm, 2mm, and 8mm. These evaluations focus on the Critical Success Index (CSI) and the False Alarm Ratio (FAR).

The Critical Success Index (CSI) is a crucial meteorological metric for assessing the binary classification accuracy of nowcasting predictions, especially in identifying whether rainfall exceeds specific thresholds. Alongside the CSI, the False Alarm Ratio (FAR) serves as another crucial metric, offering a different perspective on the detection



(a) CSI1mm

(b) FAR1mm

(c) CSI2mm

(d) FAR2mm

(e) CSI8mm

(f) FAR8mm

Figure 5.2: Evaluation of the 3-hour prediction: (categorical scores) (CSI and FAR with different thresholds: a,b for 1mm; c, d for 2mm and e, f for 8mm). This figure illustrates how the metrics vary with the forecasting lead time and includes a legend showing average scores over 3 hours.

capabilities of nowcasting systems. The primary objective in nowcasting is to achieve a balance by keeping the FAR at an acceptable level while simultaneously enhancing the CSI. This approach ensures that the model predicts rainfall accurately and minimizes false alarms.

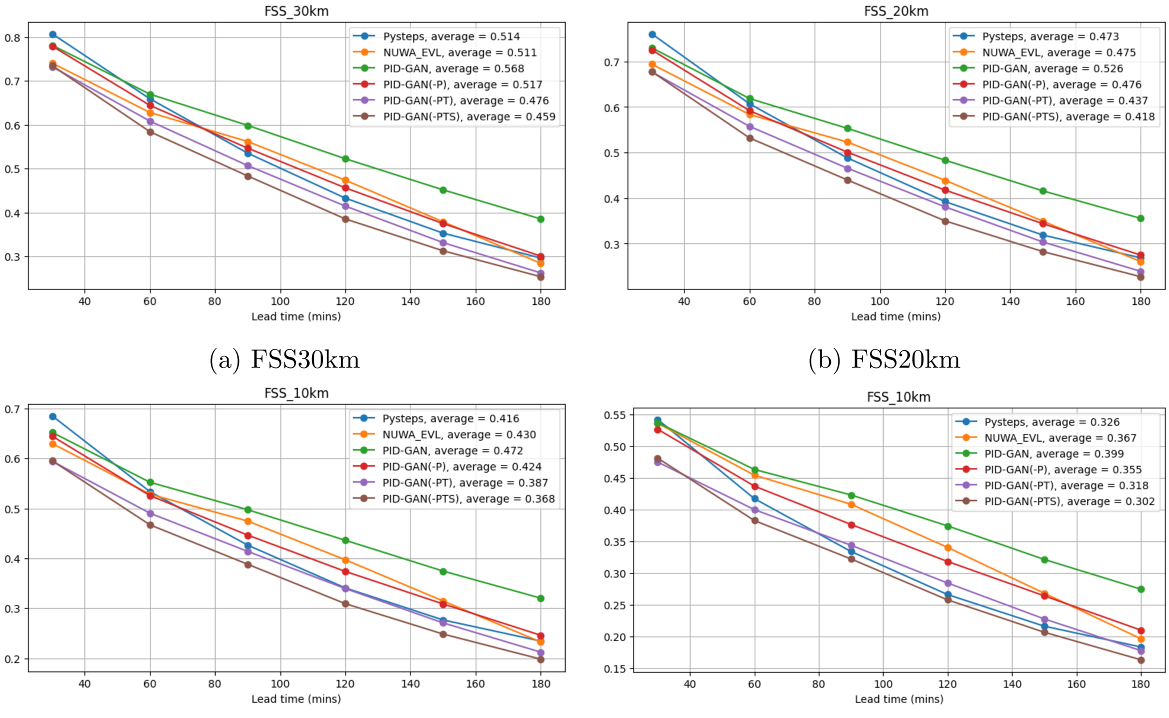
When evaluating the CSI and FAR at the 1mm and 2mm thresholds, the PID-GAN model shows the highest CSI scores among all models, maintaining the same FAR level as Pysteps. At the more severe rainfall threshold of 8mm, the PID-GAN model's performance is comparable to that of Pysteps but with a FAR that is 15% lower than

Pysteps.

Figure 5.3 illustrates the model's performance in spatial verification, using the Fraction Skill Score (FSS) at various length scales. Typically, a larger length scale corresponds to a higher FSS, indicating that errors in forecasting the location of precipitation fields are reduced when predictions are made at a coarser resolution. Additionally, the model is deemed skilful if the FSS exceeds $0.5 + \frac{1}{f0}$, where $f0$ is the skill score of a random forecast.

The results demonstrate that all models experience a decline in the Fraction Skill Score (FSS) as the lead time increases, signifying that the precision of these models in predicting the location of precipitation wanes as the forecast extends further into the future. These models are more accurate at shorter lead times, but their accuracy diminishes over more extended forecasting periods.

For instance, the PID-GAN model exhibits the highest FSS for lead times beyond 60 minutes. However, as depicted in Figure 5.3c, the PID-GAN model achieves skilful forecasting only for a lead time of 60 minutes and at a scale larger than 10 km.



(a) FSS30km

(b) FSS20km

(c) FSS10km

(d) FSS1km

Figure 5.3: Evaluation of the 3-hour prediction: (spatial scores) (FSS analysis at varying spatial resolutions: Subfigures (a), (b), (c), and (d) correspond to length scales of 30 km, 20 km, 10 km, and 1 km, respectively.) This figure illustrates how the metrics vary with the forecasting lead time and includes a legend showing average scores over 3 hours.

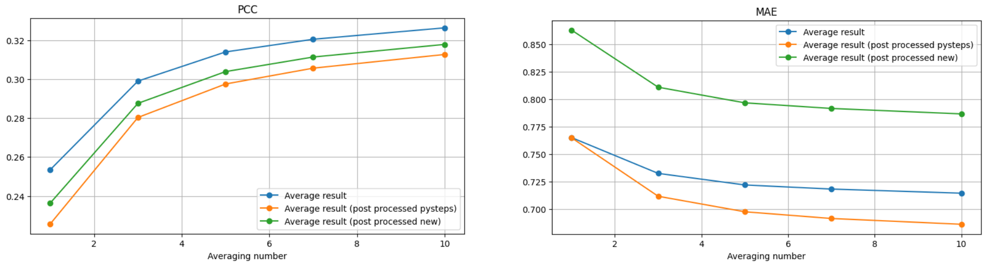
Across different lead times, the PID-GAN model consistently outperforms its counterparts in nowcasting performance. The PID-GAN(-P) model's performance is on par with the benchmark NUWA-EVL model, while the PID-GAN(-PT) and PID-GAN(-PTS) model shows comparable performance to Pysteps. Notably, all PID-GAN(-P), PID-GAN(-PT) and PID-GAN(-PTS) models benefit from a lower Mean Absolute Error (MAE) and exhibit significantly better Pearson Correlation Coefficient (PCC) and comparable FSS.

Furthermore, in light rain scenarios (at 1mm and 2mm thresholds), the PID-GAN(-P) model secures higher Critical Success Index (CSI) values, suggesting improved detection of such events, albeit with an increase in the False Alarm Ratio (FAR). In contrast, for heavy rain conditions (at the 8mm threshold), the model's CSI is slightly reduced, which indicates a marginal decrease in detection accuracy. However, it also shows a lower FAR, implying fewer false alarms.

When considering all evaluation metrics, the PID-GAN model stands out with superior performance for all lead times, especially showing significant improvements in PCC, lower MAE, and better CSI for 1mm and 2mm thresholds, as well as a lower FAR for the 8mm threshold compared to the two benchmarks. Therefore, the PID-GAN model is deemed the most effective for these nowcasting applications.

5.1.2 Results for post-pressing and average

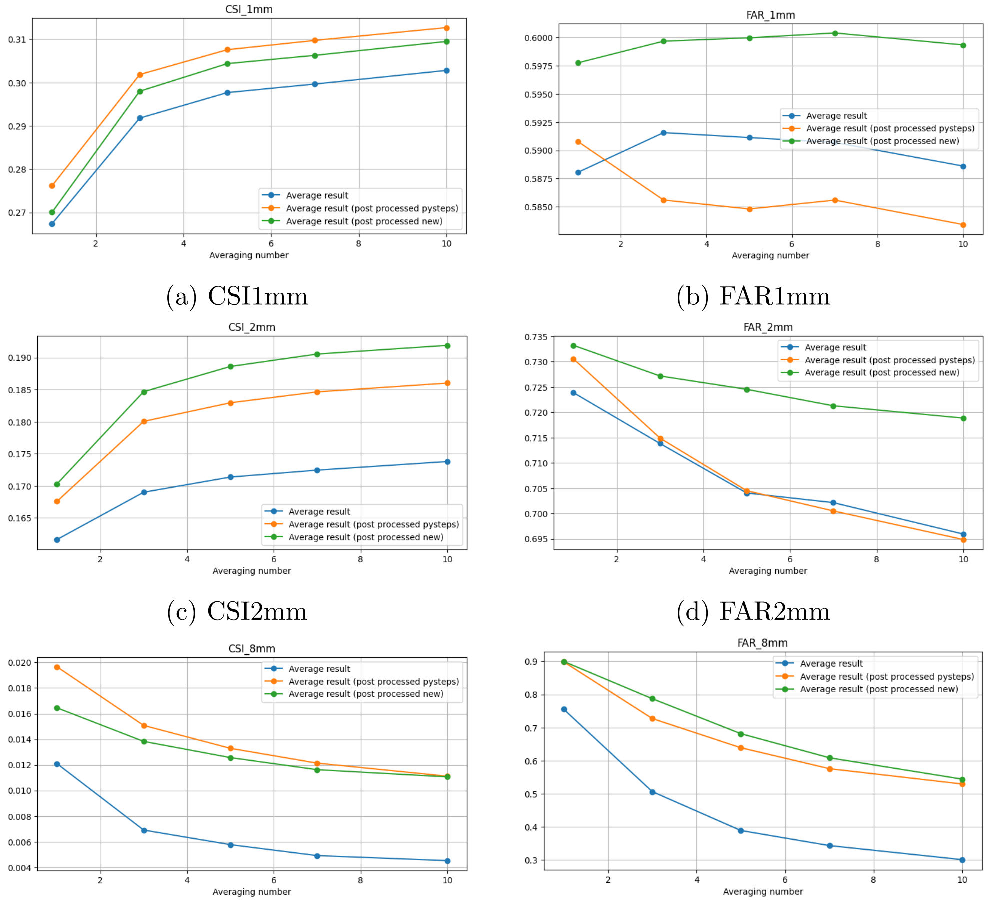
This section elucidates the impact of varying averaging numbers on nowcasting performance across the entire study area, as represented in Figures 5.4, 5.5, and 5.6. The analysis delves into the correlation between averaging numbers ranging from 1 to 9 and the 3-hour average score, particularly examining the PID-GAN model as a case study (depicted by the blue curves). Moreover, this exploration integrates two distinct post-processing techniques: the first, depicted by the orange curves, is adapted from the Pysteps approach(orange curves), while the second, represented by the green curves, is based on the methodology delineated in Chapter 4(green curves).



(a) PCC

(b) MAE

Figure 5.4: The connection between the number of averages and the pixel-level evaluation, specifically the 3-hour averaged Pearson Correlation Coefficient (PCC) in sub-figure a, and the Mean Absolute Error (MAE) in sub-figure b.



(a) CSI1mm

(b) FAR1mm

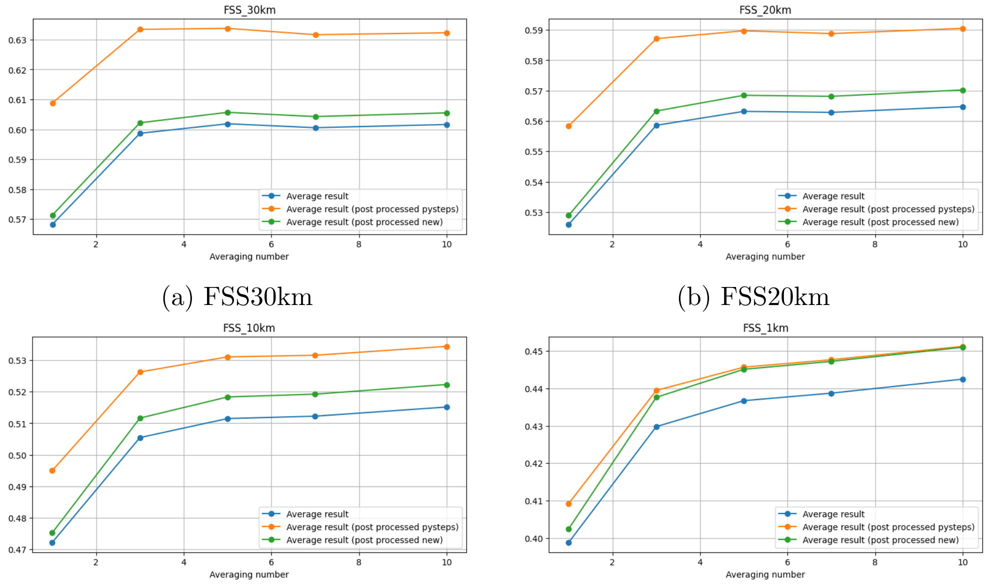
(c) CSI2mm

(d) FAR2mm

(e) CSI8mm

(f) FAR8mm

Figure 5.5: The connection between the number of averages and the pixel-level evaluation, specifically the 3-hour averaged Critical Success Index (CSI) in sub-figure (a,c,e), and the False Alarm Ratio (FAR) in sub-figure (b,d,f).



(a) FSS30km

(b) FSS20km

(c) FSS10km

(d) FSS1km

Figure 5.6: The connection between the number of averages and the pixel-level evaluation, specifically the 3-hour averaged Fraction Skill Score (FSS) at varying spatial resolutions: Subfigures (a), (b), (c), and (d) correspond to length scales of 30 km, 20 km, 10 km, and 1 km, respectively.

The data presented in the graphs indicate that applying an averaging method improves critical metrics such as the Pearson Correlation Coefficient (PCC), Mean Absolute Error (MAE), Critical Success Index at 1mm and 2mm thresholds (CSI1, CSI2), False Alarm Ratio at the 1mm threshold (FAR1), and Fraction Skill Score (FSS). The enhancement becomes more pronounced with an increasing number of averages. However, the benefits of averaging tend to plateau beyond an average number of 5, possibly due to the diminishing representation of overall rainfall intensity captured by the models post-averaging.

To counteract this effect, two post-processing techniques were applied. Generally, both methods improve the PCC, CSI, and FSS metrics compared to the non-averaged model. However, concerning MAE and FAR at the 1mm threshold, the new post-processing method results in a significant increase, suggesting a potential issue with overestimation. Conversely, the Pysteps method manages to reduce the MAE. Notably, the Pysteps method significantly improves the performance of the 2mm threshold CSI, FAR, and FSS when compared to the new post-processing method.

Given these findings and considering that a higher number of averages leads to increased generation times, an average number of 5 combined with the Pysteps post-processing technique is identified as the optimal approach for subsequent sections focusing on extreme event detection.

5.1.3 Summary of the nowcasting performance

Metrics/Models	**PySTEPs**	**NUWA-EVL**	**PID-GAN**	**PID-GAN(-P)**	**PID-GAN(-PT)**	**PID-GAN(-PTS)**
PCC ↑	0.158	0.202	**0.313**	0.288	0.250	0.241
MAE ↓	0.933	0.938	**0.686**	0.706	0.692	0.725
CSI(1mm) ↑	0.210	0.262	**0.313**	0.296	0.234	0.210
CSI(8mm) ↑	0.008	0.006	**0.011**	0.008	0.004	0.005
FAR(1mm) ↓	0.553	0.623	0.583	0.601	**0.549**	0.579
FAR(8mm) ↓	0.896	**0.399**	0.529	0.499	0.435	0.513
FSS(1km) ↑	0.326	0.394	**0.451**	0.430	0.428	0.414
FSS(10km) ↑	0.416	0.456	**0.534**	0.51	0.481	0.463
FSS(20km) ↑	0.473	0.498	**0.591**	0.565	0.521	0.508

Table 5.1: Summary of the 3-hour averaged precipitation nowcasting skill of different models (Pixel-level evaluation).

Model	**Physical Consistency(MSE)**↓
PID-GAN	**3.117**
PID-GAN(-P)	3.1618
PID-GAN(-PT)	3.2711
PID-GAN(-PTS)	3.2934
NUWA-EVL	3.5921
Pysteps	4.2102

Table 5.2: Pixel-level evaluation of Physical Consistency

Table 5.1 summarizes different models' three-hour averaged precipitation nowcasting skills at a pixel-level evaluation. Each deep learning model, including the PID-GAN(-P), PID-GAN(-PT) and PID-GAN(-PTS), employs an average number of 5, unlike PySTEPS, which utilizes an ensemble of 20. The highest metric values are highlighted in bold to denote the top performance, while the second-highest values are underlined to facilitate comparative analysis.

The review of the results indicates that the PID-GAN(-P) model generally outperforms the PySTEPS, NUWA-EVL, PID-GAN(-PT) and PID-GAN(-PTS) models, suggesting that GAN models are particularly effective for precipitation nowcasting tasks. Furthermore, integrating a Physics-Informed Discriminator (PID) has markedly improved the PID-GAN model's performance, emphasizing the advantages of incorporating physics-based principles into the adversarial learning process.

Table 5.2 presents a pixel-level evaluation of Physical Consistency, measured by Mean Squared Error (MSE). It reveals that the PID-GAN model achieves better physics consistency than the PID-GAN(-P), PID-GAN(-PT), PID-GAN(-PTS), and both benchmark models, PySTEPS and NUWA-EVL. This highlights the PID-GAN model's effective use of physical laws to inform the learning processes of its generator and discriminator, thus enhancing the reliability of its predictions.

5.2 Extreme events detection

To evaluate the detection of extreme precipitation events within 12 Dutch catchments, it is essential to identify these specific regions in the predictive precipitation maps generated by the models. Extreme events are characterized by the average precipitation over three hours within each catchment area, with critical thresholds set at the top 1% (5mm/3h) and top 5% (2mm/3h) of the highest average precipitation accumulations, as outlined in Table 5.4 and Table 5.3.

The assessment of model performance in identifying these events proceeds in two ways: Firstly, thresholds for extreme events are established at fixed values specific to each catchment area. The accuracy of detection at these thresholds is measured using four metrics: Hit Rate (HR), False Alarm Rate (FA), False Alarm Ratio (FAR), and Critical Success Index (CSI). Secondly, to gauge the models' overall effectiveness in detecting extreme events, various uniform thresholds are applied across all catchments, effectively treating each catchment as having the same criteria for an extreme event. This analysis incorporates an averaging number of 5 and employs the Pysteps post-processing technique. Considering the testing dataset encompasses 357 events for the entire study area, this results in a total of 3927 events for the 12 Dutch catchments being analyzed.

5.2.1 Evaluation of fixed threshold of extreme events

Table 5.3 presents a comparative model performance analysis across 12 Dutch catchments, using a 2mm/3-hour precipitation threshold. The PID-GAN(-PT), and PID-GAN(-PTS) models show enhanced performance metrics, notably in Hit Rate (HR) and Critical Success Index (CSI), indicating superior performance compared to benchmark models like PySTEPS. While the PID-GAN(-P) model achieves a slightly higher HR and a lower False Alarm Rate (FA) than the NUWA-EVL model, the PID-GAN model improves upon both HR and CSI compared to AR-GAN. However, it shows slight FA and False Alarm Ratio (FAR) increases. Overall, the PID-GAN model stands out with the best HR and CSI figures, maintaining FAR and FA within acceptable limits.

Table 5.3: Summary of the 5% extreme event detection performance of different models (Catchment-level evaluation, RT dataset) 2mm/3h

Models/Metrics	**HR = H/(H+M)↑**	**FA = F/(R+F)↓**	**FAR = F/(H+F)↓**	**CSI = H/(H+M+F)↑**
PySTEPS	0.5256	**0.0972**	**0.1943**	0.4665
NUWA-EVL	0.7155	0.2213	0.2875	0.5552
PID-GAN(-PTS)	0.7865	0.1654	0.2034	0.5832
PID-GAN(-PT)	0.8039	0.1745	0.2202	0.6052
PID-GAN(-P)	0.8190	0.2190	0.2586	0.6370
PID-GAN	**0.8381**	0.2245	0.2589	**0.6483**

Table 5.4 details the assessment of various models' effectiveness in detecting 1% extreme precipitation events, defined by a threshold of 5mm/3h, across 12 Dutch catchments. Among these models, the PID-GAN stands out by achieving the highest Hit Rate (HR) and the lowest False Alarm (FA), emphasizing its reliability in predicting extreme weather events. Notably, its False Alarm Ratio (FAR) is significantly lower than those of the PySTEPS and NUWA-EVL models, highlighting its precision. Furthermore, the PID-GAN model registers the highest Critical Success Index (CSI), confirming its superior performance in accurately detecting events while minimizing false alarms. In comparison, while the PID-GAN(-P), PID-GAN(-PT) and PID-GAN(-PTS) models show improvements in FAR and CSI metrics, they fall short of the PID-GAN model's exemplary performance.

Table 5.4: Summary of the 1% extreme event detection performance of different models (Catchment-level evaluation, RT dataset) 5mm/3h

Models/Metrics	**HR = H/(H+M)↑**	**FA = F/(R+F)↓**	**FAR = F/(H+F)↓**	**CSI = H/(H+M+F)↑**
PySTEPS	0.3838	**0.0965**	0.4812	0.2830
NUWA-EVL	0.3959	0.1205	0.5288	0.2941
PID-GAN(-PTS)	0.3532	0.0654	0.4021	0.2786
PID-GAN(-PT)	0.3709	0.0711	0.4138	0.2939
PID-GAN(-P)	0.4031	0.0824	0.4298	0.3092
PID-GAN	**0.4334**	0.0742	**0.3870**	**0.3403**

The PID-GAN model demonstrates superior performance over the benchmarks, PID-GAN(-PTS), PID-GAN(-PT), and PID-GAN(-P) models in detecting extreme precipitation events at various thresholds. It achieves significant enhancements in Hit Rate (HR) and Critical Success Index (CSI) while maintaining False Alarm (FA) and False Alarm Ratio (FAR) within acceptable limits. This performance highlights the model's effectiveness in integrating physics-supervised learning into the Generative Adversarial Network (GAN) architecture. Moreover, it underscores the importance of managing unbalanced datasets in precipitation nowcasting fields adeptly.

5.2.2 Comprehensive Assessment of Extreme Events Identification Capability

In order to evaluate the model's effectiveness in detecting extreme precipitation events across different thresholds, a range of thresholds from 0.5 to 10 mm (totalling 20 thresholds) is utilized for the predictive data. The extreme threshold for the ground truth data is defined according to the previously discussed definition of extreme events. This section includes three experiments:

1. The extreme threshold for the ground truth data is set at 5mm/3h. By adjusting the threshold for the predictive data, the Hit Rate (HR) is fixed at 0.8 for all models to facilitate a comparison of other evaluation metrics.

2. The full range of thresholds from 10 to 0.5 mm is employed to assess the detection performance of all models, with the extreme threshold for the ground truth data remaining at 5mm/3h. A Receiver Operating Characteristic (ROC) curve is generated in this scenario.

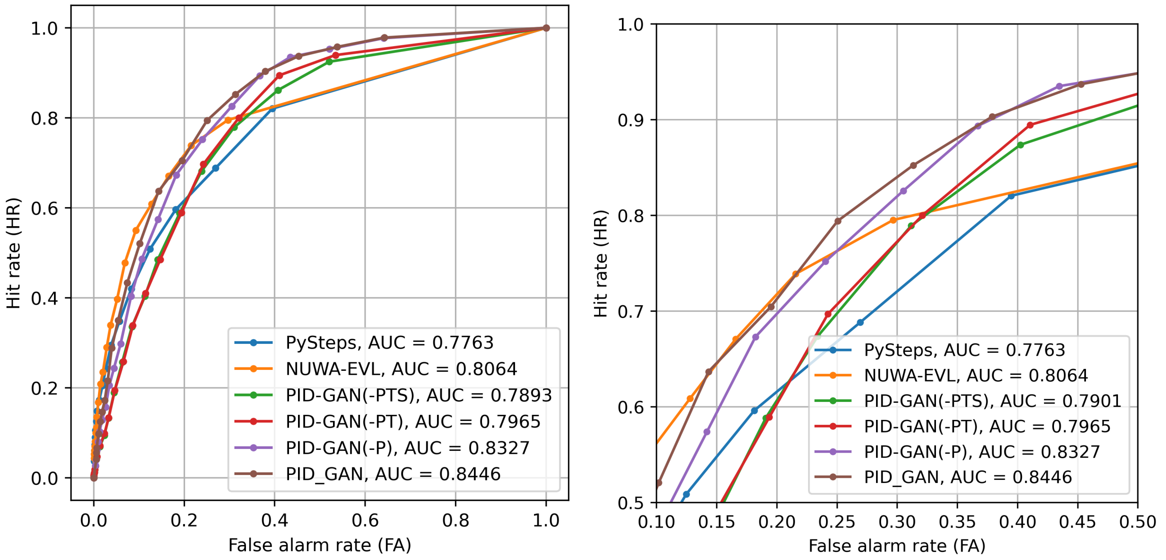
3. Two extreme thresholds for the ground truth data, 5mm and 2mm are used while the full range of thresholds from 10 to 0.5 mm (totalling 20 thresholds) is employed to evaluate the models' precision. Precision-recall curves are generated for summarization.

Table 5.5 presents the results when the HR is fixed at 0.8 for all models, enabling a comparison based on other evaluation metrics: False Alarm (FA), False Alarm Ratio (FAR), and Critical Success Index (CSI). With the HR set at 0.8, the PID-GAN(-PT), PID-GAN(-PTS) and benchmark models, such as PySTEPS and NUWA-EVL, exhibit much lower CSI values and higher FA and FAR values compared to the GAN model, underscoring the superior effectiveness of the GAN model structure. Furthermore, the PID-GAN model outperforms the baseline model across all evaluation metrics.

Models/Metrics	HR (↑)	FA (↓)	FAR (↓)	CSI (↑)
PySTEPS	0.8	0.3627	0.6089	0.3532
NUWA-EVL	0.8	0.3159	0.5819	0.3782
PID-GAN(-PTS)	0.8	0.3356	0.6278	0.3503
PID-GAN(-PT)	0.8	0.3208	0.5961	0.3669
PID-GAN(-P)	0.8	0.2902	0.5696	0.3906
PID-GAN	0.8	0.2515	0.5383	0.4127

Table 5.5: Catchment-level evaluation results for extreme-event forecasting, averaged over 3927 catchment-level events.

The ROC curve, illustrated in Figure 5.7, supports the conclusion that the PID-GAN model outperforms the other models, particularly the benchmarks, Pysteps and NUWA-EVL, as indicated by a larger area under the curve (AUC). Moreover, the PID-GAN(-PT) and PID-GAN(-PTS) model exhibits detection performance comparable to these benchmarks. While the differences between models may not be substantial across the entire curve, they become more pronounced when examining the False Alarm Rate (FA) and Hit Rate (HR) within practical ranges. This is seen in Figure 5.7b, where the PID-GAN model's enhanced performance is distinctly highlighted by restricting the HR to between 0.5 and 1 and the FA to between 0.1 and 0.5.

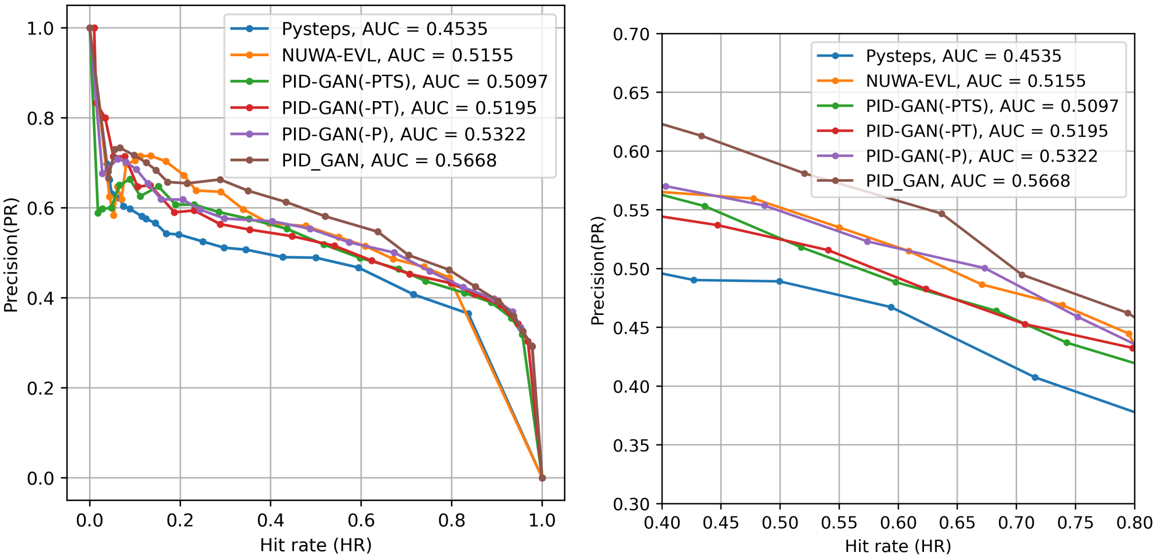


(a) Complete ROC

(b) Close-up of ROC Curve

Figure 5.7: Sub-figure a. The comprehensive ROC curves present the detection of extreme events over 3 hours. The points on the curve, arranged from left to right, correspond to precipitation thresholds ranging from 10mm to 0.5mm at the catchment level, where the reference threshold for the ground truth is set as 5mm/3h. Sub-figure b. The ROC curve is modified by constraining the hit rate to exceed 0.5 and the false alarm rate to be between 0.1 and 0.5.

The precision-recall curves offer a comprehensive model performance analysis in



(a) Complete Precision-recall Curve

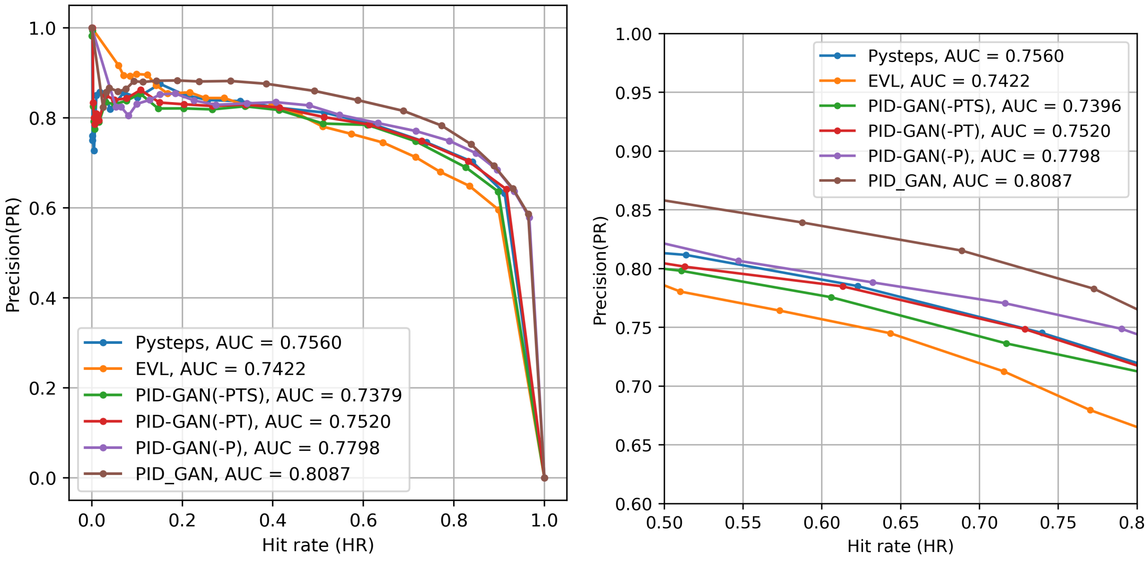
(b) Close-up of Precision-recall Curve

Figure 5.8: Sub-figure a presents comprehensive precision-recall curves for detecting extreme events over 3 hours. The points on the curve, arranged from left to right, correspond to precipitation thresholds ranging from 10mm to 0.5mm at the catchment level, where the reference threshold for the ground truth is set as 5mm/3h. Sub-figure b. The precision-recall curve is modified by constraining the precision between 0.2 and 0.8 and the hit rate between 0.5 and 0.8.

detecting extreme precipitation events within a 3-hour forecast period. These curves are assessed based on precision and hit rate across various precipitation thresholds, which span from 10mm to 0.5mm at the catchment level. A reference threshold for the ground truth is also established at 5mm/3h.

Figure 5.8a presents the full precision-recall curves for all the models in comparison. Notably, the PID-GAN model exhibits the highest Area Under the Curve (AUC), signifying its superior capability to balance precision and recall relative to other models, including Pysteps, NUWA-EVL, PID-GAN(-PT), and the PID-GAN(-PTS). Although the PID-GAN(-PT) and PID-GAN(-PTS) model does not reach the AUC levels of the PID-GAN, they still show competitive performance, with an AUC surpassing that of Pysteps and approaching that of NUWA-EVL.

In Figure 5.8b, a specific segment of the precision-recall curve is examined, focusing on a more stringent range for precision and hit rate. By honing in on these narrower ranges, the distinctions in model performance become more apparent. Within this focused perspective, the PID-GAN model demonstrates superior precision at any given hit rate, further confirming its effectiveness in detecting extreme events compared to its counterparts, particularly under these more demanding conditions.



(a) Complete Precision-recall Curve

(b) Close-up of Precision-recall Curve

Figure 5.9: Sub-figure a presents comprehensive precision-recall curves for detecting extreme events over 3 hours. The points on the curve, arranged from left to right, correspond to precipitation thresholds ranging from 10mm to 0.5mm at the catchment level, where the reference threshold for the ground truth is set as 2mm/3h. Sub-figure b. The precision-recall curve is modified by constraining the precision between 0.2 and 0.8 and the hit rate between 0.5 and 0.8.

In Figure 5.9, the precision-recall curve analysis assesses the models' ability to detect extreme precipitation events within a 3-hour forecast period using a modified ground truth reference threshold of 2mm/3h, compared to the 5mm/3h threshold used in the prior analyses. Figure 5.9a displays the precision-recall curves across the entire range.

The PID-GAN model again exhibits the highest Area Under the Curve (AUC), underlining its consistent leading performance in precisely identifying extreme events. This outcome aligns with earlier results, suggesting that the robustness of the PID-GAN model is not significantly affected by changes in the ground truth threshold.

Figure 5.9b provides a zoomed-in view of a specific curve segment where precision exceeds 0.5, and the hit rate lies between 0.5 and 0.8. This focused perspective allows for a detailed comparison of model performances within a more confined operational range. The PID-GAN model maintains a notable lead in this segment, underscoring its effective identification of extreme events under various ground truth thresholds. The consistency of the PID-GAN model's superior performance, even after threshold adjustments, reinforces its robustness and reliability in detecting extreme events under diverse conditions. Additionally, the ROC and precision-recall curves for all 12 Dutch catchments, as presented in Appendix E, affirm the same conclusion: the PID-GAN model outperforms the others.

Conclusion and Further Research 6

6.1 Conclusion

This thesis presents a pioneering approach to nowcasting and extreme precipitation event detection using a hybrid "VQGAN + Transformer" model integrated with physics-informed machine learning (PIML) principles. The developed model addresses the critical challenge of accurately forecasting precipitation intensities over short periods and identifying extreme precipitation events, leveraging the advanced capabilities of deep generative models while incorporating fundamental meteorological principles to ensure physical realism and accuracy.

The core contributions of this work include the formulation of a novel generative model tailored for the complexities of precipitation nowcasting tasks. By innovatively combining VQGAN for high-resolution image synthesis with the Transformer model for modelling sequential data, the research achieves notable advancements in generating accurate precipitation forecasts. The model's architecture is further refined with the introduction of a Physics-Informed Discriminator (PID-GAN), enhancing its ability to produce forecasts that are not only visually accurate but also adhere to meteorological constraints, thus improving the reliability of extreme event detection.

Empirical results demonstrate that the proposed model achieves comparable performance in overall nowcasting accuracy and extreme event detection when compared to existing methods such as PySTEPS. This marks a significant advancement in the application of deep learning and physics-informed machine learning to meteorological forecasting. The integration of the "VQGAN + Transformer" model with PIML principles not only enhances the model's ability to forecast precipitation with high accuracy but also improves its capability to identify extreme precipitation events effectively.

Furthermore, the thesis highlights the importance of integrating physics-based constraints into deep learning models. This integration not only bolsters the models' predictive accuracy but also ensures that the forecasts remain grounded in the fundamental laws governing atmospheric processes. Such an approach is pivotal for advancing nowcasting technologies and enhancing their application in real-world scenarios, where the timely and accurate prediction of precipitation events is crucial for effective disaster management and mitigation strategies.

In conclusion, this work not only contributes a significant methodological advancement to the field of precipitation nowcasting and extreme event detection but also sets a new precedent for the integration of deep learning with physical constraints. The findings underscore the potential of physics-informed machine learning in revolutionizing weather forecasting, offering a promising avenue for future research and development in this domain.

6.2 Further Research

6.2.1 Data

The meteorological data from Automatic Weather Stations used in this thesis primarily consists of hourly measurements. However, the Royal Netherlands Meteorological Institute (KNMI) provides a dataset with a higher temporal resolution, reporting essential meteorological parameters such as temperature, relative humidity, wind speed and direction, and air pressure every 10 minutes. This higher-resolution dataset offers several advantages for meteorological analysis and forecasting, particularly by capturing the dynamic nature of weather patterns more accurately.

Measurements reported at ten-minute intervals allow for a more detailed understanding of atmospheric conditions. This is crucial for nowcasting tasks, where detecting rapid meteorological changes can significantly enhance forecast accuracy and timeliness for precipitation and extreme weather events.

6.2.2 Model

There have been limitations observed in the VQ-GAN reconstruction process, particularly regarding the insufficient detailing of precipitation maps and the less-than-optimal reconstruction of high rainfall pixels. To address these issues, future research should aim to enhance the model's ability to capture and accurately reconstruct these critical aspects. A promising approach involves integrating advanced encoding techniques or developing specialized loss functions designed to prioritize areas of high detail within the precipitation maps. Implementing attention mechanisms could also prove beneficial by allowing the model to focus more on regions with high rainfall intensity, ensuring these areas are reconstructed with greater precision. Further exploration into adopting multi-scale approaches or hierarchical models could yield significant advancements, enabling the VQ-GAN to more effectively capture and reconstruct the full range of precipitation intensities at different scales. Additionally, VideoGPT, as introduced by

Yan et al., [63] extends the VQGAN+autoregressive transformer paradigm to video analysis, adapting it from its original application on images. In this advanced model, the incorporation of 3D convolutions within the VQ-GAN encoder plays a pivotal role, enabling the extraction of not only spatial features from individual frames but also temporal or depth features across the sequence. This modification is particularly advantageous for handling radar map data, which naturally forms a time series of images akin to a video sequence. Capturing the dynamics between successive frames, this approach can facilitate a more nuanced understanding of precipitation patterns over time. For future research, enhancing VideoGPT to further optimize the processing of temporal relationships in weather data could offer significant improvements by refining the model to more accurately represent the progression of weather events or developing new techniques to handle the inherent variability and complexity of meteorological data. By focusing on these areas, the model's ability to learn detailed discrete latent representations of sequential weather patterns could be substantially improved, potentially leading to breakthroughs in the accuracy and reliability of precipitation forecasting and extreme weather event detection.

Incorporating the moisture conservation equation as a physical constraint within the PID-GAN model has laid a solid foundation for integrating physics-informed principles into deep learning frameworks for meteorological forecasting. Looking ahead, there's a promising avenue for expanding this approach by embedding additional physical constraints that govern atmospheric dynamics, particularly the relationship between extreme precipitation events and air temperature [68]. This further integration aims to deepen the model's comprehension of the complex interplay between various meteorological factors, enhancing its predictive accuracy and reliability.

By incorporating the correlation between air temperature and extreme precipitation into the PID-GAN model, future research can address the nuances of climate variability and its impact on precipitation patterns. Such advancements would enable the model to adjust its predictions based on temperature variations, providing a more nuanced analysis of potential extreme weather events. This approach aligns with the broader objective of developing models that not only forecast with high precision but also encapsulate the multifaceted nature of weather systems.

The current approach estimates vertical wind speed by using the wind speed at different heights. However, it's important to note that atmospheric interactions, which significantly influence weather patterns, extend up to the end of the troposphere, approximately 10 km in altitude. Therefore, by focusing on wind speed within the lower 100 meters, this approach inevitably entails a degree of uncertainty, given the comprehensive atmospheric interactions occurring beyond this range. A promising future direction for enhancing the physical realism and accuracy of precipitation forecasting models involves leveraging Numerical Weather Prediction (NWP) models to estimate wind speed at higher atmospheric heights, beyond the 100 meters currently focused on. NWP models, renowned for their comprehensive simulation of atmospheric dynamics, offer valuable insights into wind patterns and behaviors at various altitudes, including those at the upper levels of the troposphere, which can significantly impact weather systems and precipitation processes.

Incorporating wind speed data from NWP models into precipitation forecasting could provide a more nuanced understanding of the atmospheric conditions contributing to precipitation events. By extending the analysis to higher altitudes, researchers can capture the full vertical profile of wind speeds, playing a crucial role in the formation and movement of weather systems. This approach could greatly enhance the model's ability to predict precipitation with higher accuracy, particularly for extreme weather events where upper atmospheric conditions are critical.

Furthermore, integrating wind speed data from NWP models introduces the opportunity to explore the interactions between different atmospheric layers and their collective influence on precipitation patterns. It enables the development of forecasting models that consider the three-dimensional complexity of the atmosphere, offering a significant leap forward in our ability to simulate and predict weather phenomena.

Bibliography

[1] N. A. of Sciences, D. on Earth, L. Studies, B. on Atmospheric Sciences, C. on Extreme Weather Events, and C. C. Attribution, *Attribution of extreme weather events in the context of climate change*. National Academies Press, 2016.

[2] H. Tabari, "Climate change impact on flood and extreme precipitation increases with water availability," *Scientific reports*, vol. 10, no. 1, p. 13 768, 2020.

[3] R. Prudden, S. Adams, D. Kangin, *et al.*, "A review of radar-based nowcasting of precipitation and applicable machine learning techniques," *arXiv preprint arXiv:2005.04988*, 2020.

[4] W. M. O. (WMO), "Guidelines for nowcasting techniques," in *Guidelines for Nowcasting Techniques*, WMO, 2017.

[5] J. Sun, M. Xue, J. W. Wilson, *et al.*, "Use of nwp for nowcasting convective precipitation: Recent progress and challenges," *Bulletin of the American Meteorological Society*, vol. 95, no. 3, pp. 409–426, 2014.

[6] S. Pulkkinen, D. Nerini, A. A. Pérez Hortal, *et al.*, "Pysteps: An open-source python library for probabilistic precipitation nowcasting (v1. 0)," *Geoscientific Model Development*, vol. 12, no. 10, pp. 4185–4219, 2019.

[7] N. E. Bowler, C. E. Pierce, and A. W. Seed, "Steps: A probabilistic precipitation forecasting scheme which merges an extrapolation nowcast with downscaled nwp," *Quarterly Journal of the Royal Meteorological Society: A journal of the atmospheric sciences, applied meteorology and physical oceanography*, vol. 132, no. 620, pp. 2127–2155, 2006.

[8] J. Bech and J. L. Chau, *Doppler radar observations: Weather radar, wind profiler, ionospheric radar, and other advanced applications*. BoD–Books on Demand, 2012.

[9] R. O. Imhoff, C. C. Brauer, K.-J. van Heeringen, R. Uijlenhoet, and A. H. Weerts, "Large-sample evaluation of radar rainfall nowcasting for flood early warning," *Water Resources Research*, vol. 58, no. 3, e2021WR031591, 2022.

[10] R. Imhoff, C. Brauer, A. Overeem, A. Weerts, and R. Uijlenhoet, "Spatial and temporal evaluation of radar rainfall nowcasting techniques on 1,533 events," *Water Resources Research*, vol. 56, no. 8, e2019WR026723, 2020.

[11] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo, "Convolutional lstm network: A machine learning approach for precipitation nowcasting," *Advances in neural information processing systems*, vol. 28, 2015.

[12] X. Shi, Z. Gao, L. Lausen, *et al.*, "Deep learning for precipitation nowcasting: A benchmark and a new model," *Advances in neural information processing systems*, vol. 30, 2017.

[13] M. Qiu, P. Zhao, K. Zhang, *et al.*, "A short-term rainfall prediction model using multi-task convolutional neural networks," in *2017 IEEE international conference on data mining (ICDM)*, IEEE, 2017, pp. 395–404.

[14] G. Ayzel, T. Scheffer, and M. Heistermann, "Rainnet v1. 0: A convolutional neural network for radar-based precipitation nowcasting," *Geoscientific Model Development*, vol. 13, no. 6, pp. 2631–2644, 2020.

[15] I. Goodfellow, J. Pouget-Abadie, M. Mirza, *et al.*, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014.

[16] L. Tian, X. Li, Y. Ye, P. Xie, and Y. Li, "A generative adversarial gated recurrent unit model for precipitation nowcasting," *IEEE Geoscience and Remote Sensing Letters*, vol. 17, no. 4, pp. 601–605, 2019.

[17] J. Jing, Q. Li, X. Ding, N. Sun, R. Tang, and Y. Cai, "Aenn: A generative adversarial neural network for weather radar echo extrapolation," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 42, pp. 89–94, 2019.

[18] K. Zheng, Y. Liu, J. Zhang, *et al.*, "A generative adversarial model for radar echo extrapolation based on convolutional recurrent units," *Geoscientific Model Development Discussions*, vol. 2021, pp. 1–11, 2021.

[19] J. Willard, X. Jia, S. Xu, M. Steinbach, and V. Kumar, "Integrating scientific knowledge with machine learning for engineering and environmental systems," *ACM Computing Surveys*, vol. 55, no. 4, pp. 1–37, 2022.

[20] M. Reichstein, G. Camps-Valls, B. Stevens, *et al.*, "Deep learning and process understanding for data-driven earth system science," *Nature*, vol. 566, no. 7743, pp. 195–204, 2019.

[21] K. Kashinath, M. Mustafa, A. Albert, *et al.*, "Physics-informed machine learning: Case studies for weather and climate modelling," *Philosophical Transactions of the Royal Society A*, vol. 379, no. 2194, p. 20 200 093, 2021.

[22] D. Cho, C. Yoo, B. Son, J. Im, D. Yoon, and D.-H. Cha, "A novel ensemble learning for post-processing of nwp model's next-day maximum air temperature forecast in summer using deep learning and statistical approaches," *Weather and Climate Extremes*, vol. 35, p. 100 410, 2022.

[23] M. Choma, P. Šimánek, and J. Bartel, "Improving deep learning precipitation nowcasting by using prior knowledge," *arXiv preprint arXiv:2301.11707*, 2023.

[24] R. Wang, K. Kashinath, M. Mustafa, A. Albert, and R. Yu, "Towards physics-informed deep learning for turbulent flow prediction," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020, pp. 1457–1466.

[25] A. Chattopadhyay, M. Mustafa, P. Hassanzadeh, and K. Kashinath, "Deep spatial transformers for autoregressive data-driven forecasting of geophysical turbulence," in *Proceedings of the 10th international conference on climate informatics*, 2020, pp. 106–112.

[26] P. Esser, R. Rombach, and B. Ommer, "Taming transformers for high-resolution image synthesis," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 12 873–12 883.

[27] A. Daw, M. Maruf, and A. Karpatne, "Pid-gan: A gan framework based on a physics-informed discriminator for uncertainty quantification with physics," in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 2021, pp. 237–247.

[28] H. Bi, M. Kyryliuk, Z. Wang, *et al.*, "Nowcasting of extreme precipitation using deep generative models," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2023, pp. 1–5.

[29] *KNMI - KNMI opent neerslagradar in Herwijnen — knmi.nl*, https://www.knmi.nl/over-het-knmi/nieuws/knmi-opent-neerslagradar-in-herwijnen, [Accessed 08-02-2024].

[30] A. Best, "The size distribution of raindrops," *Quarterly journal of the royal meteorological society*, vol. 76, no. 327, pp. 16–36, 1950.

[31] A. Overeem, I. Holleman, and A. Buishand, "Derivation of a 10-year radar-based climatology of rainfall," *Journal of Applied Meteorology and Climatology*, vol. 48, no. 7, pp. 1448–1463, 2009.

[32] *Uurwaarden van weerstations — daggegevens.knmi.nl*, https://www.daggegevens.knmi.nl/klimatologie/uurgegevens, [Accessed 09-02-2024].

[33] H. Hersbach, B. Bell, P. Berrisford, *et al.*, "The era5 global reanalysis," *Quarterly Journal of the Royal Meteorological Society*, vol. 146, no. 730, pp. 1999–2049, 2020.

[34] B. S. Murphy, "Pykrige: Development of a kriging toolkit for python," in *AGU fall meeting abstracts*, vol. 2014, 2014, H51K–0753.

[35] M. A. Oliver and R. Webster, "Kriging: A method of interpolation for geographical information systems," *International Journal of Geographical Information System*, vol. 4, no. 3, pp. 313–332, 1990.

[36] Z. K. Bargaoui and A. Chebbi, "Comparison of two kriging interpolation methods applied to spatiotemporal rainfall," *Journal of Hydrology*, vol. 365, no. 1-2, pp. 56–73, 2009.

[37] G. D. Knott, *Interpolating cubic splines*. Springer Science & Business Media, 1999, vol. 18.

[38] E. Maeland, "On the comparison of interpolation methods," *IEEE transactions on medical imaging*, vol. 7, no. 3, pp. 213–217, 1988.

[39] R. V. Rohli and C. Li, *Meteorology for Coastal Scientists*. Springer Nature, 2021.

[40] H. De Bruin, "From penman to makkink," in *Evaporation and Weather: Technical Meeting 44, Ede, The Netherlands 25 March 1987. The Hague, Netherlands. 1987. p 5-31. 1 fig, 4 tab, 34 ref.*, 1987.

[41] R. G. Allen, L. S. Pereira, D. Raes, M. Smith, *et al.*, "Crop evapotranspiration-guidelines for computing crop water requirements-fao irrigation and drainage paper 56," *Fao, Rome*, vol. 300, no. 9, p. D05109, 1998.

[42] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," *arXiv preprint arXiv:1710.10196*, 2017.

[43] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4401–4410.

[44] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *International conference on machine learning*, PMLR, 2017, pp. 214–223.

[45] Q. Guo, W. Feng, R. Gao, Y. Liu, and S. Wang, "Exploring the effects of blur and deblurring to visual object tracking," *IEEE Transactions on Image Processing*, vol. 30, pp. 1812–1824, 2021.

[46] Y. Zhang, Z. Gan, and L. Carin, "Generating text via adversarial training," in *NIPS workshop on Adversarial Training*, academia. edu, vol. 21, 2016, pp. 21–32.

[47] N.-T. Tran, V.-H. Tran, N.-B. Nguyen, T.-K. Nguyen, and N.-M. Cheung, "On data augmentation for gan training," *IEEE Transactions on Image Processing*, vol. 30, pp. 1882–1897, 2021.

[48] S. Ravuri, K. Lenc, M. Willson, *et al.*, "Skilful precipitation nowcasting using deep generative models of radar," *Nature*, vol. 597, no. 7878, pp. 672–677, 2021.

[49] K. Schreurs, Y. Shapovalova, M. Schmeits, K. Whan, and T. Heskes, "Precipitation nowcasting using generative adversarial networks," 2021.

[50] C. Wu, J. Liang, L. Ji, *et al.*, "Nüwa: Visual synthesis pre-training for neural visual world creation," in *European conference on computer vision*, Springer, 2022, pp. 720–736.

[51] A. Karpatne, G. Atluri, J. H. Faghmous, *et al.*, "Theory-guided data science: A new paradigm for scientific discovery from data," *IEEE Transactions on knowledge and data engineering*, vol. 29, no. 10, pp. 2318–2331, 2017.

[52] J. Willard, X. Jia, S. Xu, M. Steinbach, and V. Kumar, "Integrating physics-based modeling with machine learning: A survey," *arXiv preprint arXiv:2003.04919*, vol. 1, no. 1, pp. 1–34, 2020.

[53] B. Dai and D. Wipf, "Diagnosing and enhancing vae models," *arXiv preprint arXiv:1903.05789*, 2019.

[54] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[55] D. J. Rezende, S. Mohamed, and D. Wierstra, "Stochastic backpropagation and approximate inference in deep generative models," in *International conference on machine learning*, PMLR, 2014, pp. 1278–1286.

[56] P. Esser, R. Rombach, and B. Ommer, "A disentangling invertible interpretation network for explaining latent representations," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9223–9232.

[57] Z. Xiao, Q. Yan, Y.-a. Chen, and Y. Amit, "Generative latent flow: A framework for non-adversarial image generation," *arXiv preprint arXiv:1905.10485*, 2019.

[58] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2223–2232.

[59] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[60] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, *et al.*, "Improving language understanding by generative pre-training," 2018.

[61] N. Li, S. Liu, Y. Liu, S. Zhao, and M. Liu, "Neural speech synthesis with transformer network," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, 2019, pp. 6706–6713.

[62] M. Chen, A. Radford, R. Child, *et al.*, "Generative pretraining from pixels," in *International conference on machine learning*, PMLR, 2020, pp. 1691–1703.

[63] W. Yan, Y. Zhang, P. Abbeel, and A. Srinivas, "Videogpt: Video generation using vq-vae and transformers," *arXiv preprint arXiv:2104.10157*, 2021.

[64] D. Waddington, J. Colmenares, J. Kuang, and F. Song, "Kv-cache: A scalable high-performance web-object cache for manycore," in *2013 IEEE/ACM 6th In-*

ternational Conference on Utility and Cloud Computing, IEEE, 2013, pp. 123–130.

[65] G. Chen and W.-C. Wang, "Short-term precipitation prediction for contiguous united states using deep learning," *Geophysical Research Letters*, vol. 49, no. 8, e2022GL097904, 2022.

[66] U. Germann and I. Zawadzki, "Scale-dependence of the predictability of precipitation from continental radar images. part i: Description of the methodology," *Monthly Weather Review*, vol. 130, no. 12, pp. 2859–2873, 2002.

[67] M. Berenguer, D. Sempere-Torres, and G. G. Pegram, "Sbmcast–an ensemble nowcasting technique to assess the uncertainty in rainfall forecasts by lagrangian extrapolation," *Journal of Hydrology*, vol. 404, no. 3-4, pp. 226–240, 2011.

[68] G. Lenderink and E. Van Meijgaard, "Linking increases in hourly precipitation extremes to atmospheric temperature and moisture changes," *Environmental Research Letters*, vol. 5, no. 2, p. 025 208, 2010.

Appendix: Interpolated Map for the meteorological data A

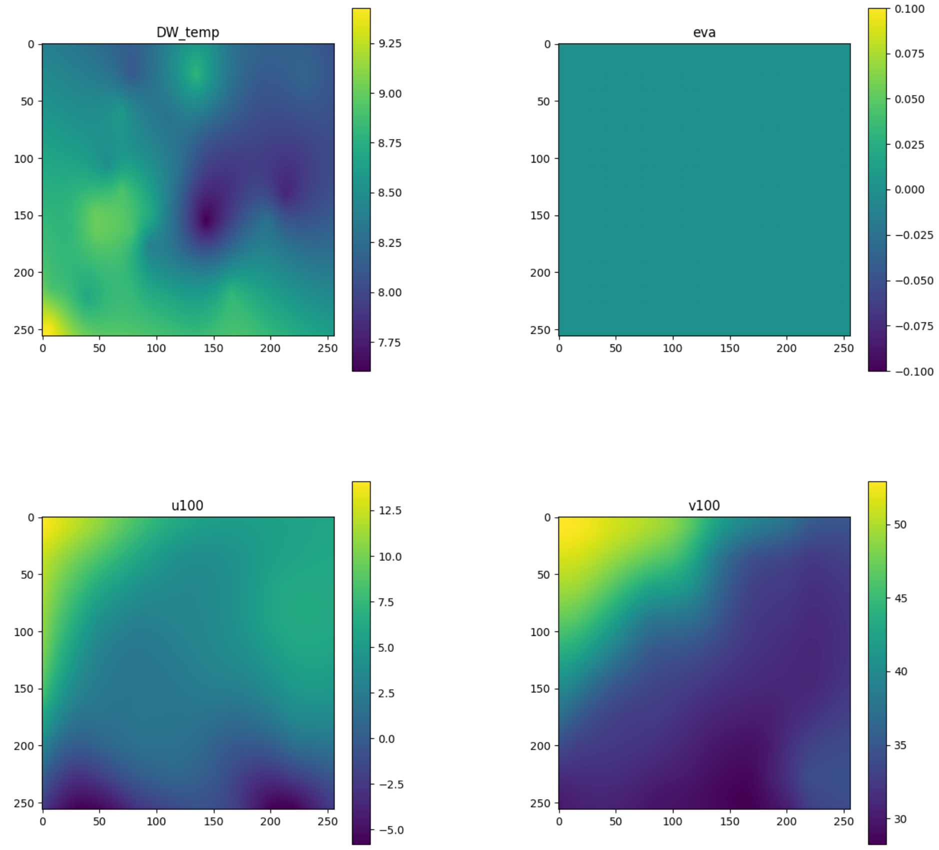


Figure A.1: Example of Interpolated Map for the meteorological data: evapotranspiration rate (EVA), Dew Point Temperature (DW-temp), East-west wind component at 100m (u100), South-north wind component at 100m (v100)

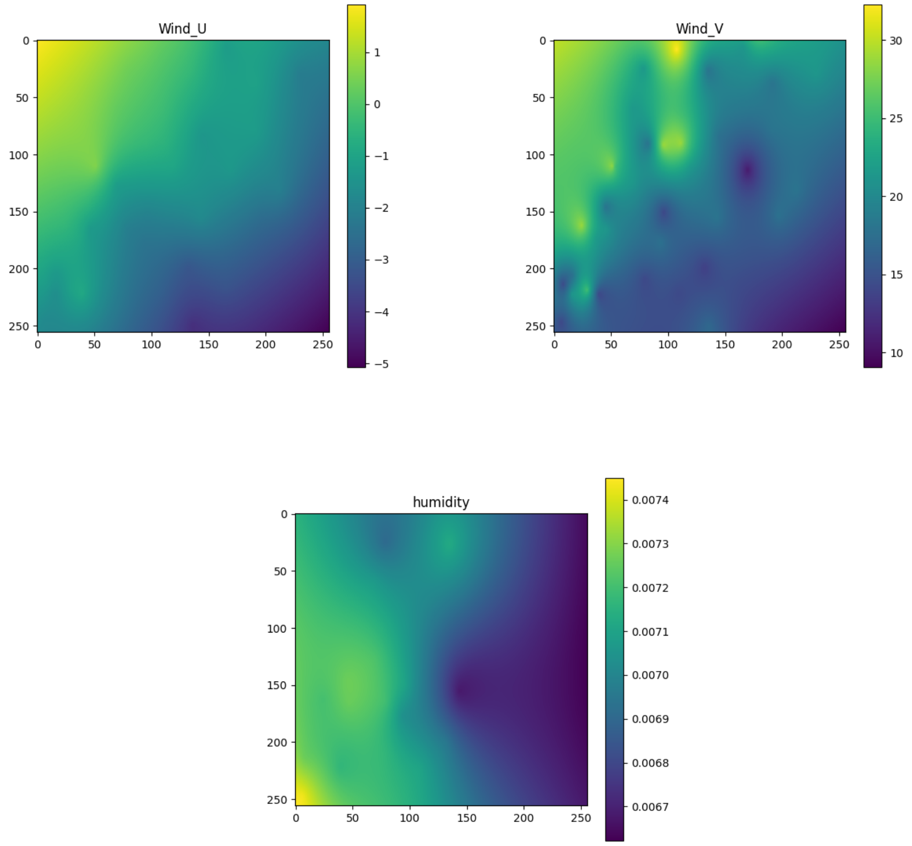


Figure A.2: Example of Interpolated Map for the meteorological data: East-west wind component at 10m (Wind-U), South-north wind component at 10m (Wind-V), specific humidity (humidity)

Appendix:AWS data B

Table B.1: Description of available data parameters from AWS

Variable	**Symbol**	**Resolution**	**Description**
Wind direction	DD	1 degree	Averaged over the last 10 minutes of the past hour. 360=north, 90=east, 180=south, 270=west, 0=calm 990=variable
Hourly mean wind speed	FH	0.1 m/s	
Wind speed	FF	0.1 m/s	Averaged over the last 10 minutes of the last hour
Highest wind gust	FX	0.1 m/s	Over the past hour
Temperature	T	0.1 degrees Celsius	At a height of 1.50 m during the observation
Minimum temperature	T10N	0.1 degrees Celsius	At 10 cm height in the last 6 hours
Dew point temperature	TD	0.1 degrees Celsius	At 1.50 m altitude during the observation
Duration of sunshine	SQ	0.1 hours	Per hour box, calculated from global radiation (-1 for <0.05 hours)
Global radiation	Q	J/cm2	Per hourly period
Duration of precipitation	DR	0.1 hours	Per hour box
Hourly sum of precipitation	RH	0.1 mm	(-1 for < 0.05 hours)
Air pressure	P	0.1 hPa	Reduced to sea level, during the observation
Horizontal view during sighting	VV	0-89 coding scheme	0=less than 100m, 1=100-200m, 2=200-300m,..., 49=4900-5000m, 50=5-6km, 56=6-7km, 57= 7-8km, ..., 79=29-30km, 80=30-35km, 81=35-40km,..., 89=more than 70km)
Cloud cover	N	0-9 coding scheme	Coverage of the upper air in eighths during the observation
Relative humidity	U	0.01	At 1.50 m altitude during the observation
Weather code	WW	00-99 coding scheme	Detected visually (WW) or automatically (WaWa), for the current weather or the weather in the past hour
Weather code indicator for the mode of observation	IX	1-7 coding scheme	1=manned using code from visual observations, 2,3=manned and omitted no major weather event, no data), 4=automatic and recorded using code from visual observations), 5.6=automatic and omitted no major weather phenomenon, no data), 7=automatic using code from automatic observations)
Fog	M	Boolean	0=not occurred, 1= occurred in the previous hour and/or during the observation
Rain	R	Boolean	0=not occurred, 1= occurred in the previous hour and/or during the observation
Snow	S	Boolean	0=not occurred, 1= occurred in the previous hour and/or during the observation
Thunderstorm	O	Boolean	0=not occurred, 1= occurred in the previous hour and/or during the observation
ICE	Y	Boolean	0=not occurred, 1=occurred in the previous hour and/or during observation

Table B.2: Geographic and Altitudinal Data of AWS

STN	NAME	LON	LAT	ALT
209	IJmond	4.518	52.465	0.0
210	Valkenburg Zh	4.43	52.171	-0.2
215	Voorschoten	4.437	52.141	-1.1
225	IJmuiden	4.555	52.463	4.4
235	De Kooy	4.781	52.928	1.2
240	Schiphol	4.79	52.318	-3.3
242	Vlieland	4.921	53.241	10.8
248	Wijdenes	5.174	52.634	0.8
249	Berkhout	4.979	52.644	-2.4
251	Hoorn Terschelling	5.346	53.392	0.7
257	Wijk aan Zee	4.603	52.506	8.5
258	Houtribdijk	5.401	52.649	7.3
260	De Bilt	5.18	52.1	1.9
265	Soesterberg	5.274	52.13	13.9
267	Stavoren	5.384	52.898	-1.3
269	Lelystad	5.52	52.458	-3.7
270	Leeuwarden	5.752	53.224	1.2
273	Marknesse	5.888	52.703	-3.3
275	Deelen	5.873	52.056	48.2
277	Lauwersoog	6.2	53.413	2.9
278	Heino	6.259	52.435	3.6
279	Hoogeveen	6.574	52.75	15.8
280	Eelde	6.585	53.125	5.2
283	Hupsel	6.657	52.069	29.1
286	Nieuw Beerta	7.15	53.196	-0.2
290	Twenthe	6.891	52.274	34.8
315	Hansweert	3.998	51.447	0.0
319	Westdorpe	3.861	51.226	1.7
323	Wilhelminadorp	3.884	51.527	1.4
324	Stavenisse	4.006	51.596	0.0
330	Hoek van Holland	4.122	51.992	11.9
331	Tholen	4.193	51.48	0.0
340	Woensdrecht	4.342	51.449	19.2
343	Rotterdam Geulhaven	4.313	51.893	3.5
344	Rotterdam	4.447	51.962	-4.3
348	Cabauw Mast	4.926	51.97	-0.7
350	Gilze-Rijen	4.936	51.566	14.9
356	Herwijnen	5.146	51.859	0.7
370	Eindhoven	5.377	51.451	22.6
375	Volkel	5.707	51.659	22.0
377	Ell	5.763	51.198	30.0
391	Arcen	6.197	51.498	19.5

C

Reconstruction Examples

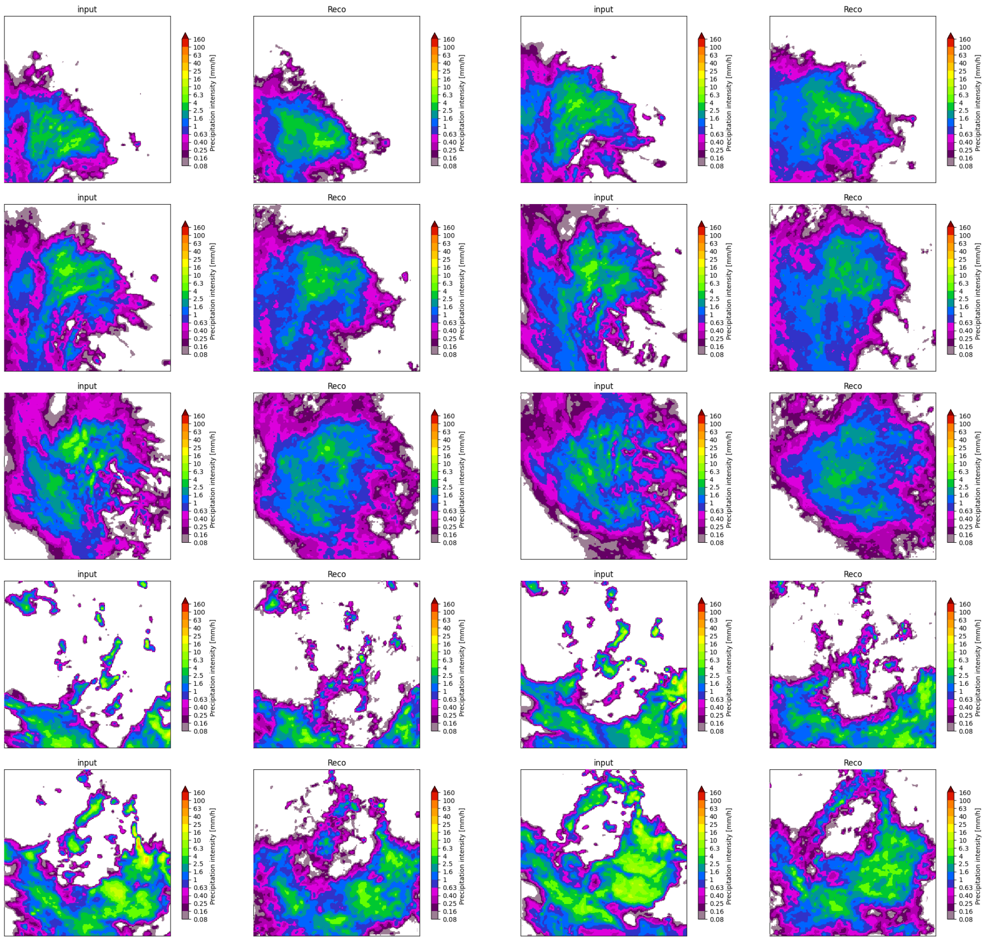


Figure C.1: Example of reconstruction of Precipitation fields by the VQ-GAN, input is the original image, reco is the reconstruction.

Nowcasting results Examples D

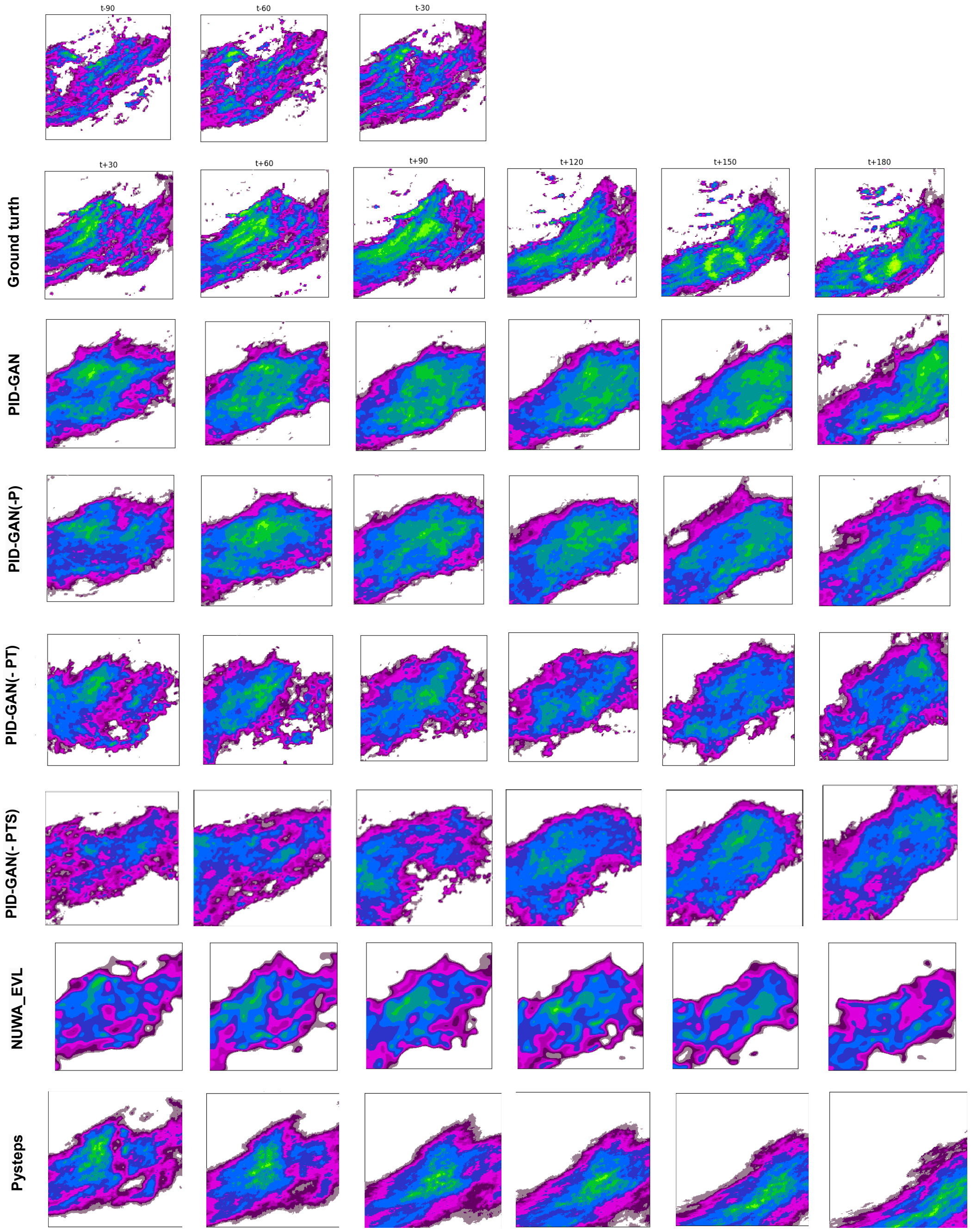
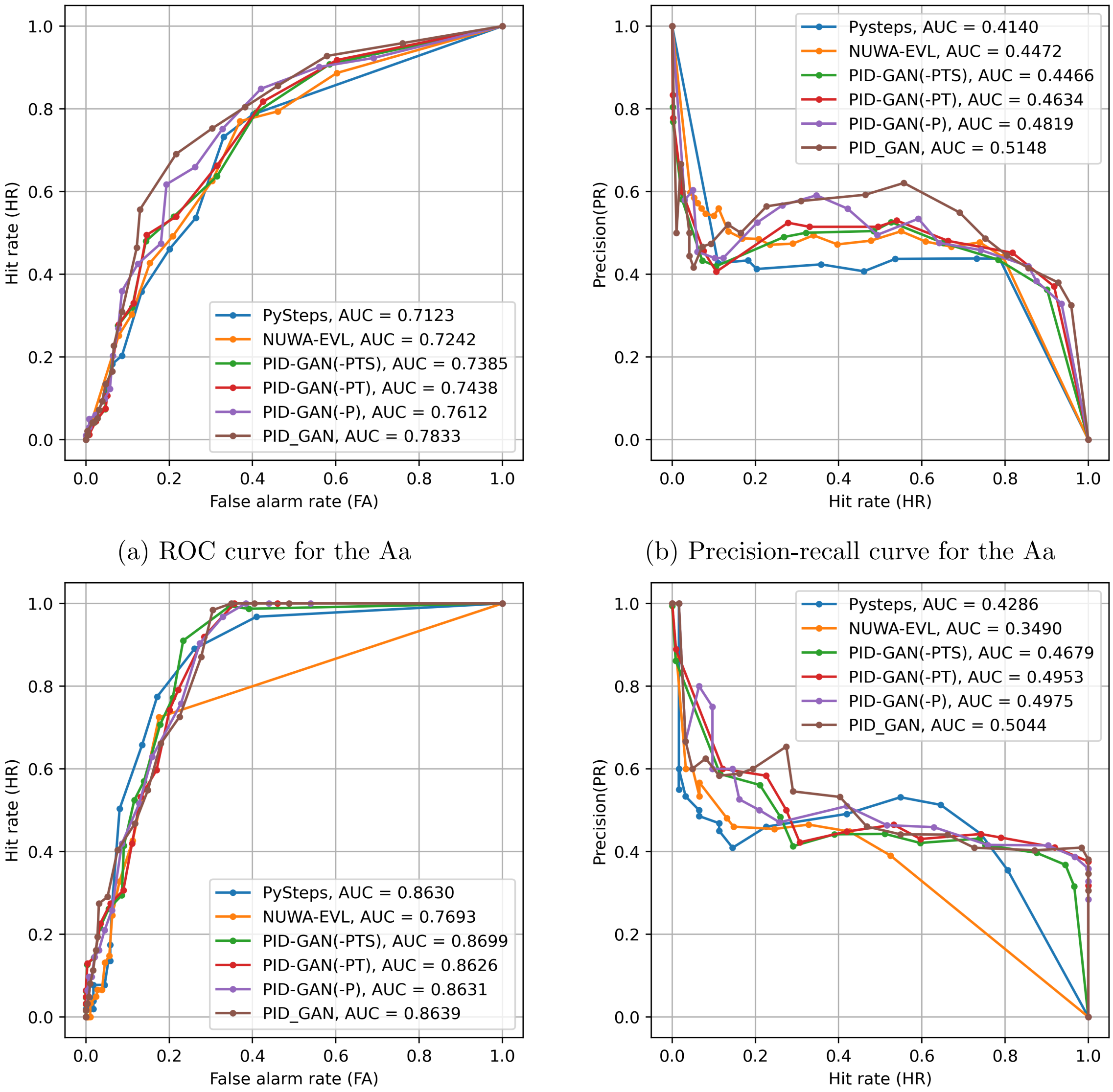


Figure D.1: Nowcasting result of different models, t=2019/11/28/05:45

E Extreme events detection on 12 Dutch catchments



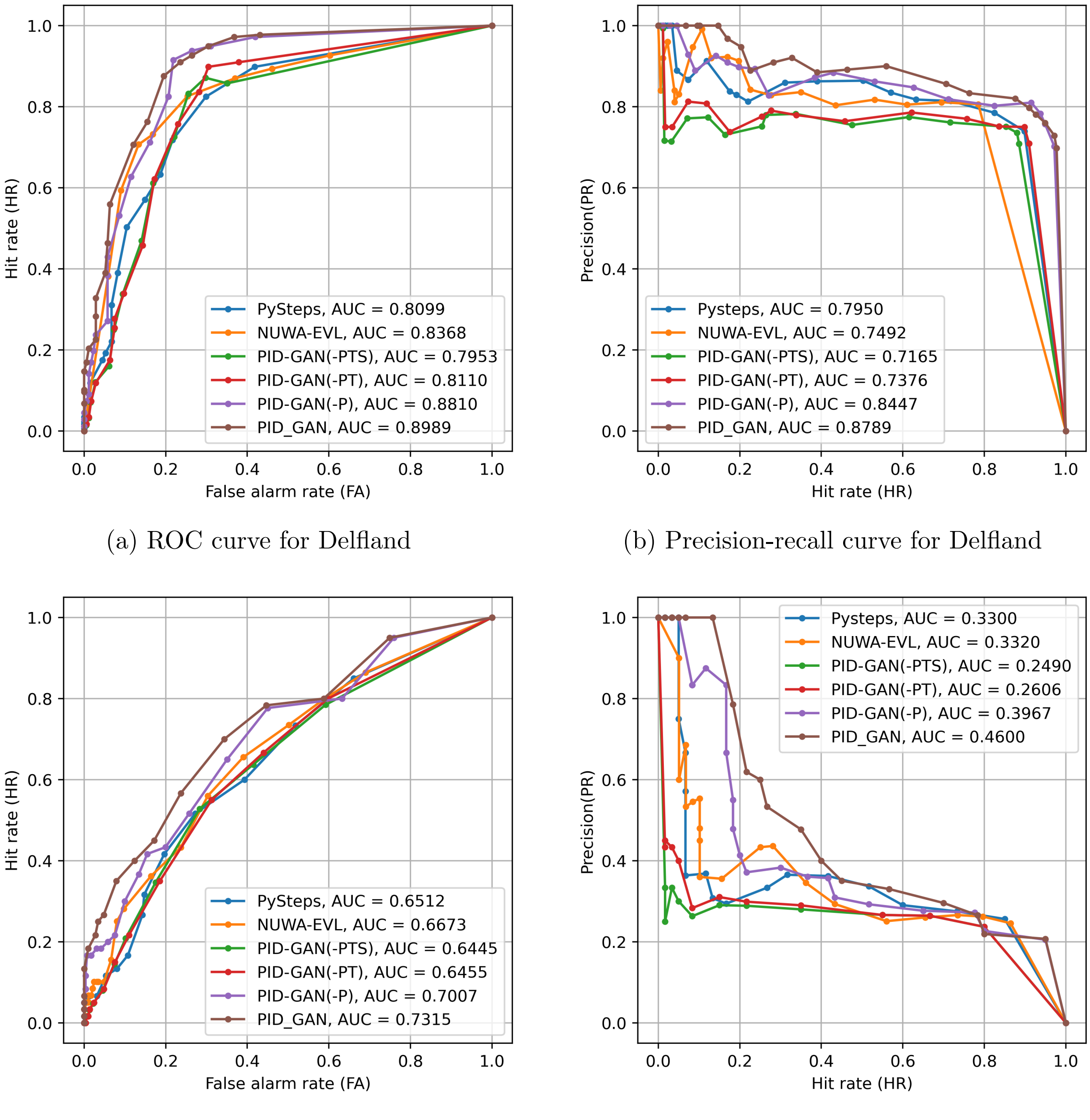
(a) ROC curve for the Aa

(b) Precision-recall curve for the Aa

(c) ROC curve for the Beemster

(d) Precision-recall curve for the Beemster

Figure E.1: ROC and Precision-recall curves for Aa and Beemster



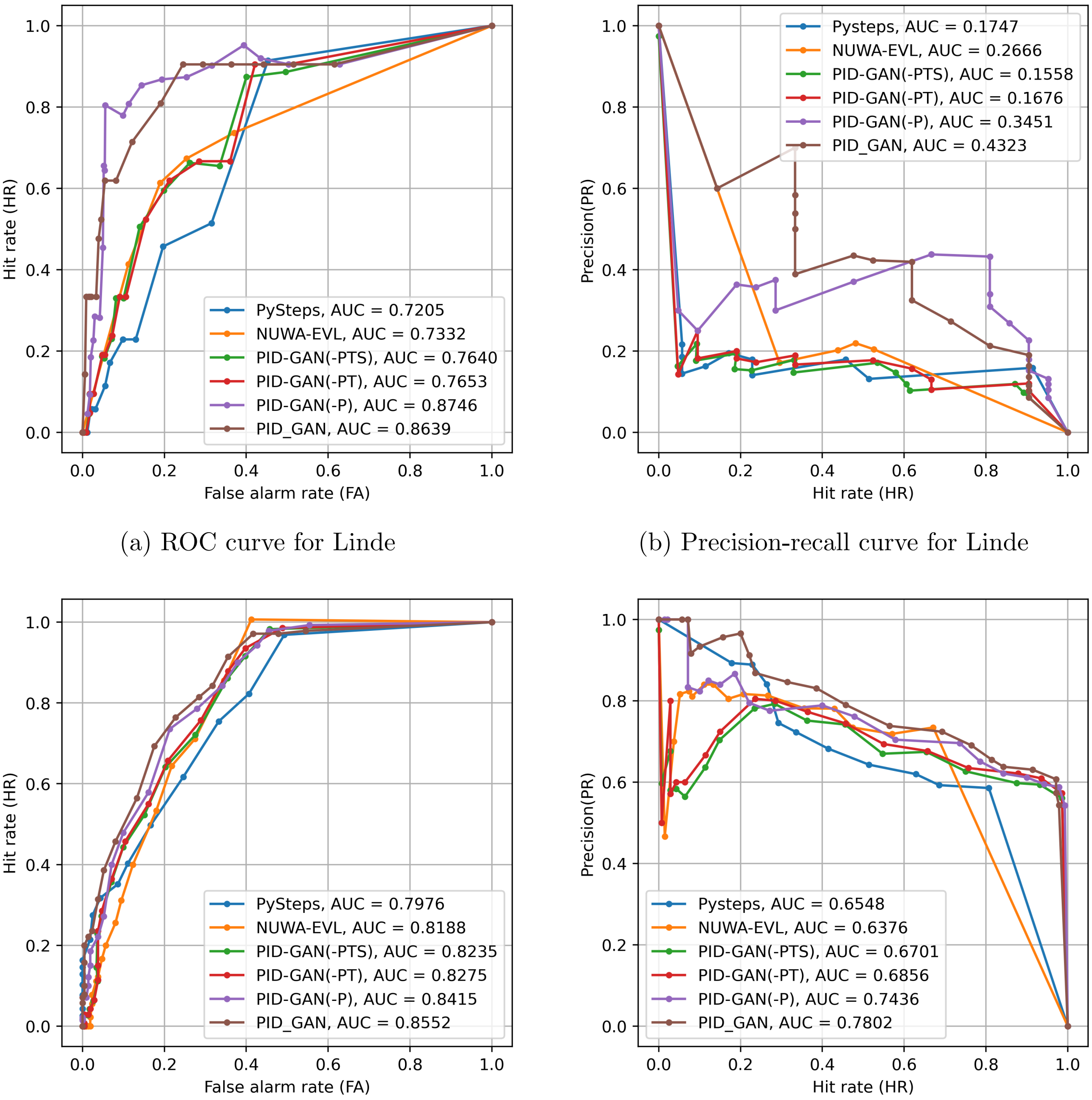
(a) ROC curve for Delfland

(b) Precision-recall curve for Delfland

(c) ROC curve for Reusel

(d) Precision-recall curve for Reusel

Figure E.2: ROC and Precision-recall curves for Delfland and Reusel



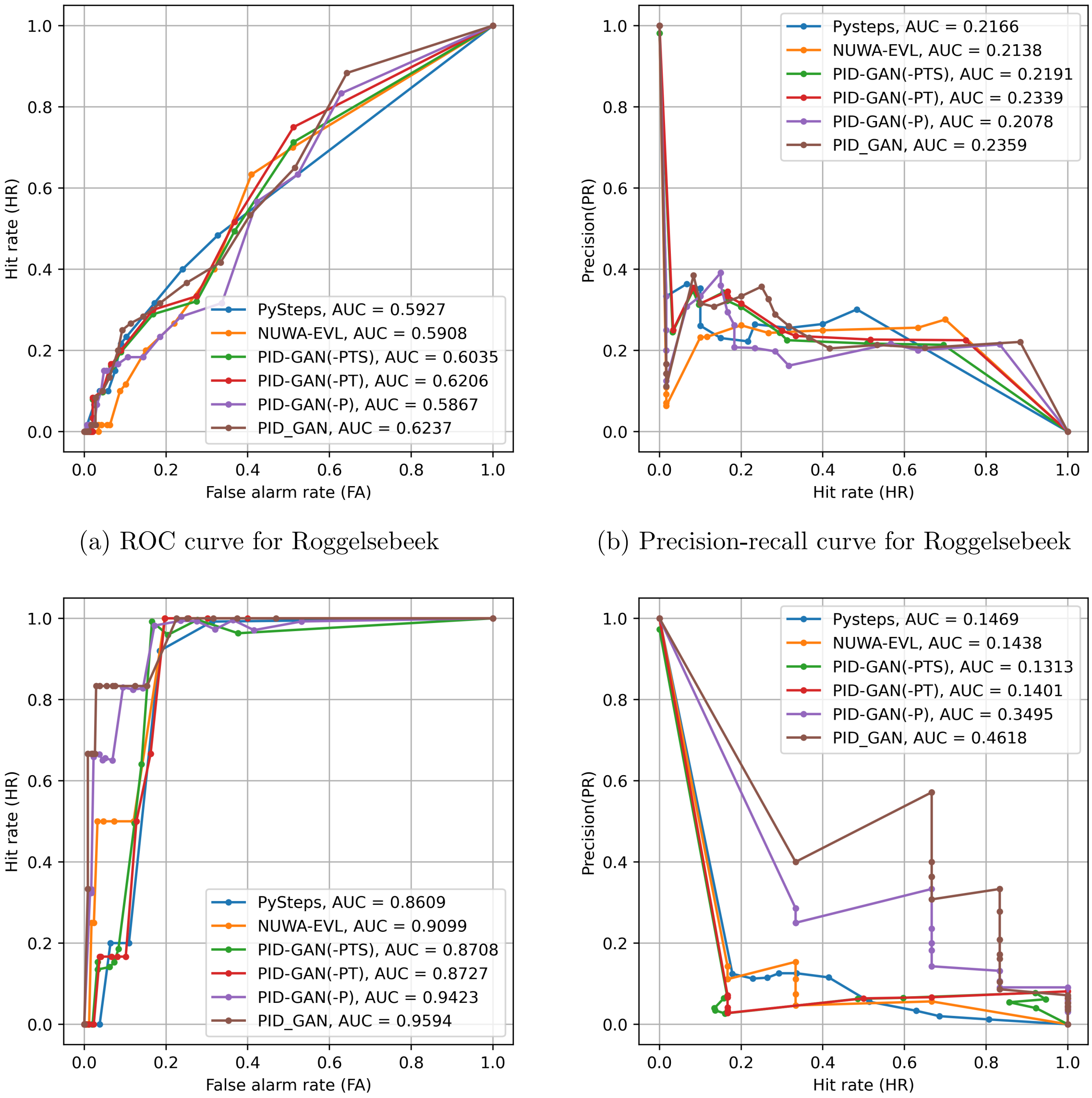
(a) ROC curve for Linde

(b) Precision-recall curve for Linde

(c) ROC curve for Rijnland

(d) Precision-recall curve for Rijnland

Figure E.3: ROC and Precision-recall curves for Linde and Rijnland



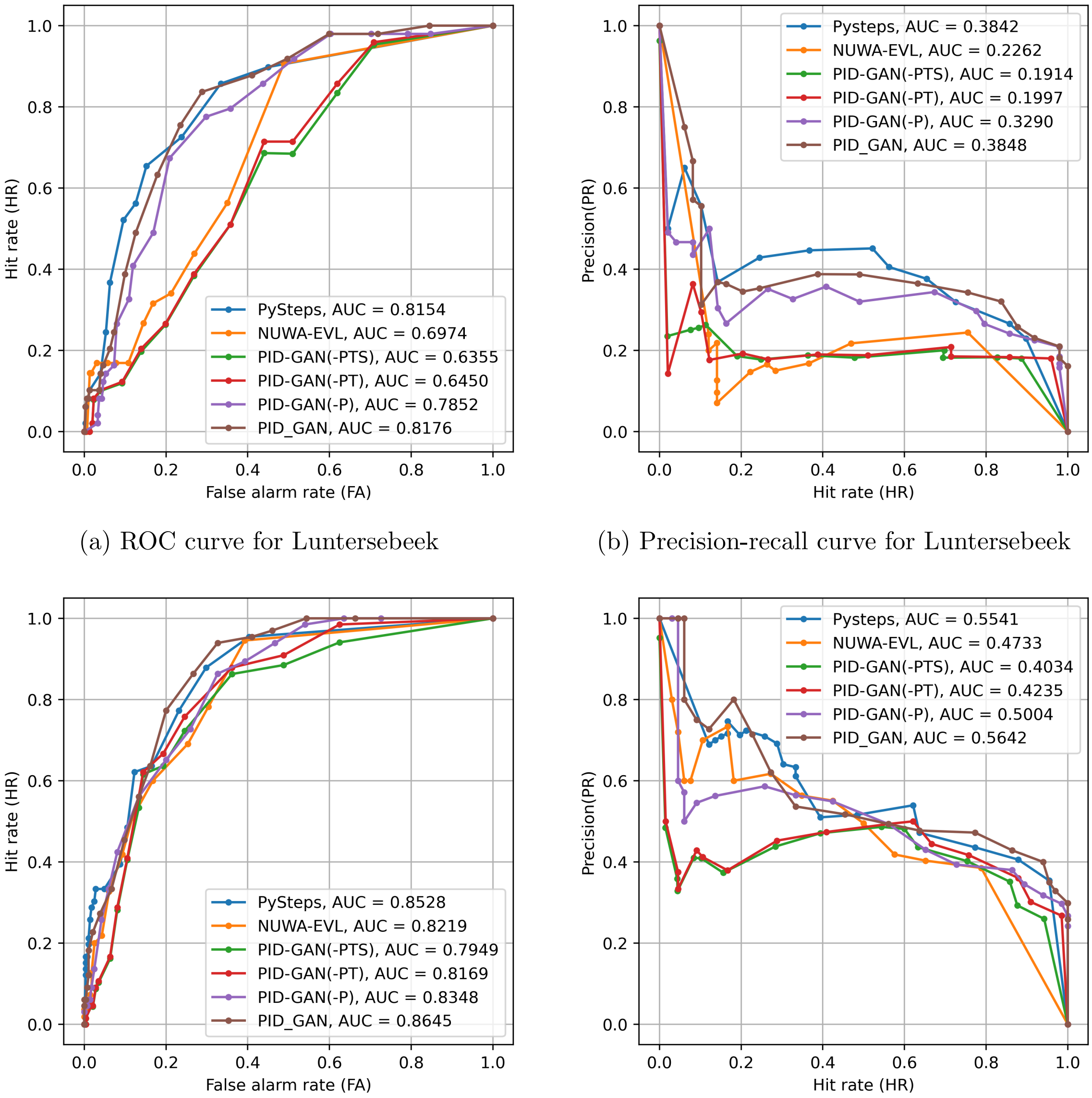
(a) ROC curve for Roggelsebeek

(b) Precision-recall curve for Roggelsebeek

(c) ROC curve for Dwarsdiep

(d) Precision-recall curve for Dwarsdiep

Figure E.4: ROC and Precision-recall curves for Roggelsebeek and Dwarsdiep



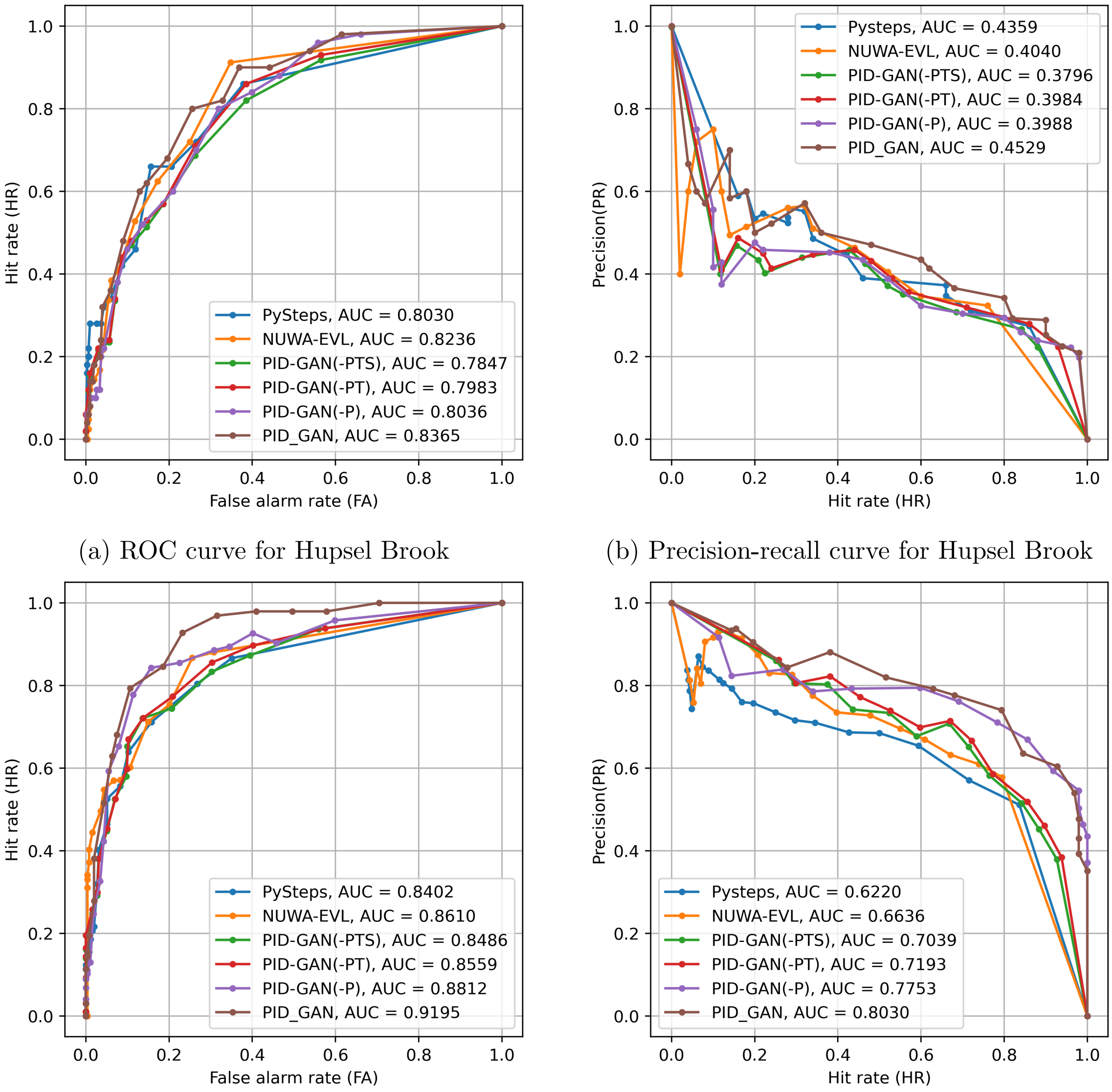
(a) ROC curve for Luntersebeek

(b) Precision-recall curve for Luntersebeek

(c) ROC curve for Grote Waterleiding

(d) Precision-recall curve for Grote Waterleiding

Figure E.5: ROC and Precision-recall curves for Luntersebeek and Grote Waterleiding



(a) ROC curve for Hupsel Brook

(b) Precision-recall curve for Hupsel Brook

(c) ROC curve for Regge

(d) Precision-recall curve for Regge

Figure E.6: ROC and Precision-recall curves for Hupsel Brook and Regge

F Comparison of Generation time of the different models

Model	Number of Parameters	Generation Time
Nuwa-EVL	772,832 M	345.67 s
PySTEPS	-	15.69 s
PID-GAN	433.689 M	43.34 s
PID-GAN(-P):	433.689 M	43.56 s
PID-GAN(-PT):	432.118 M	41.21 s
PID-GAN(-PTS):	421.086 M	40.21 s

Table F.1: Comparison of Generation time of the different models

The table provides a comparative overview of the number of parameters and generation times across several models used for nowcasting. Notably, Nuwa-EVL, with 772.832 million parameters, has the longest generation time of 345.67 seconds. PySTEPS, despite an unspecified number of parameters, showcases the fastest generation time at 15.69 seconds, underlining its efficiency and suitability as a benchmark model for nowcasting.

The PID-GAN models, with slight variations in parameters and generation times, indicate a trade-off between complexity and speed. The full PID-GAN model with 433.689 million parameters has a generation time of 43.34 seconds. Variants of the PID-GAN with certain components removed (-P, -PT, -PTS) show marginal differences in generation times, suggesting that each component's removal does not significantly impact speed. The PID-GAN without the spatial discriminator (PTS) reduces the parameter count to 421.086 million and achieves the shortest generation time of 40.21 seconds among the deep learning models listed.

Overall, the summary suggests that while deep learning models have more parameters and longer generation times compared to PySTEPS, optimizations such as KV-caching, as well as input data conversion to NumPy arrays, can lead to improved efficiency during generation processes.