

Cloud-Top Temperature in DL-based Precipitation Nowcasting

Fusion Strategies and Rain Regime-Conditioned
Uncertainty in NowcastNet over the Nether-
lands

Ruilin An

Delft University of Technology

Cloud-Top Temperature in DL-based Precipitation Nowcasting

Fusion Strategies and Rain
Regime-Conditioned Uncertainty in
NowcastNet over the Netherlands

by

Ruilin An

to obtain the degree of Master of Science

at the Delft University of Technology,

to be defended publicly on Wednesday August 19, 2026 at 10:00 AM.

Student number: 6127754
Project duration: Jan 6, 2026 – August 19, 2026
Thesis committee: Dr. E. Abraham, TU Delft, supervisor
Dr. M. A. Schleiss, TU Delft, supervisor
Ir. T. Stolp, HKV Lijn in Water

Cover: Cover photograph by the author.

Preface

This thesis was completed as part of the MSc programme in Environmental Engineering at Delft University of Technology. I would like to express my sincere gratitude to my supervisors, Dr. Edo Abraham and Dr. Marc Schleiss, for their guidance, constructive feedback, and continuous support throughout this project. I am also deeply grateful to Ir. Thomas Stolp for the valuable discussions, technical support, and the opportunity to carry out this research. In addition, I would like to thank Dorien Lugt for making this opportunity possible. This project has truly been a useful and enjoyable part of my master's studies. I would like to thank everyone who contributed ideas, discussions, technical support, and feedback throughout the process.

Working with NowcastNet has been an interesting and valuable experience. The project also showed me that atmospheric science and deep learning can be combined in many different ways. Although it was not possible to investigate every question within the available time, I would have liked to spend more time exploring some of these topics. If I had encountered them earlier in my master's programme, I might have chosen to study them in greater depth.

My heartfelt thanks also go to my housemates and friends. I am especially grateful to my high school friends for the many good meals we shared and for their persistent attempts to persuade me to return home and hang out. I would also like to thank my friends from my bachelor's programme for sharing their experiences of further study, professional life in China, and work in the civil service. Their different perspectives provided me with valuable insights into both professional development and adult life.

My heartfelt thanks also go to the alumni of my bachelor's university who, like me, came to the Netherlands as international students and later navigated the challenges of finding employment here. I am deeply grateful for the valuable experiences and insights they shared with me, as well as for their generous encouragement and support throughout this journey.

Looking back, I am somehow also grateful for all the internship rejections I received during the first year of my studies. They were frustrating at the time, but they pushed me to improve my CV, gain more experience, and become more resilient. More importantly, they taught me that, especially when you have little or no work experience, simply sending applications is often not enough. Talking to people and building connections can be much more helpful than I had expected. Sometimes, reaching out to someone, asking a question, or inviting them for a short conversation can already create an opportunity. I learned that there is no need to be afraid of asking. Even when someone can only help a little, that little bit of advice may still lead you somewhere unexpected.

I am equally thankful for the many bowls of ramen and plates of pizza, spaghetti, and tiramisu that I shared with my friends on many many weekends in The Hague, as well as for the memorable mango smoothies from a wonderful Colombian restaurant in Amsterdam-Zuid. Moreover, I would like to thank the TU Delft X classes, all ping-pong sessions and all badminton sessions for helping me stay physically active and mentally healthy, and for giving me the energy and positive mindset needed to complete my studies.

I would also like to thank my bicycle. Although well worn and already third-hand, it proved extraordinarily resilient and never abandoned me in moments of urgency. My deepest appreciation also goes to my second handed Illy Y3.3 coffee machine and to volle melk from Albert Heijn. Completing this thesis without either of you would have been almost unimaginable.

My deepest gratitude goes to my family—my mom, dad, grandma, and grandpa—for their unwavering love, encouragement, and both emotional and financial support throughout this journey.

Looking back, I realise that this journey has changed me not only academically, but also personally. I have

become more self-aware and more mature, both in my outlook on life and in the way I approach professional relationships and responsibilities. The setbacks, uncertainties, and occasional moments of frustration made me increasingly aware of my own limitations, but they also encouraged me to confront them and continue improving. Living, studying, and searching for opportunities in a new environment also taught me that becoming an adult is not a single achievement, but a continuous process of learning how to take responsibility, communicate with others, and find one's place in society. I am still far from becoming the person I hope to be, but I have learned to approach this process with greater humility, patience, and determination.

*Ruilin An
Delft, August 2026*

Abstract

Short-term precipitation nowcasting is important for operational decision-making and the timely response to high-impact rainfall. Radar observations alone provide limited information on cloud development and convective evolution, which can restrict the prediction of newly developing precipitation. This study investigates the use of cloud-top temperature (CTT) in deep-learning-based precipitation nowcasting, focusing on CTT–radar fusion strategies and rain-regime-conditioned uncertainty in NowcastNet over the Netherlands.

The relationship between CTT and rainfall intensity was first examined to determine whether distinct precipitation regimes could be identified. Based on their temporal relationship, precipitation events were grouped into two clusters representing different CTT–rainfall characteristics. The identified relationship provided the basis for investigating the incorporation of CTT into NowcastNet. Three levels of CTT conditioning were considered: input-level fusion using concatenated and stacked radar–CTT inputs, half fusion introducing a separate CTT encoder into the evolution network, and full fusion additionally conditioning the generative network through multi-scale feature modulation. Model performance was evaluated using aggregated verification metrics and event-based spatial analyses. Finally, the two CTT–rainfall clusters were used to define a regime-dependent affine post-processing method for adjusting the latent noise employed in ensemble generation.

Our results show that incorporating CTT generally enables NowcastNet to detect a larger proportion of observed precipitation than the radar-only baseline. However, the effect depends strongly on the fusion strategy: the input-level fusion models perform poorly, the half-fusion model achieves the best overall CSI, and the full-fusion model produces the highest POD but also more false alarms. Event-based spatial analyses further show that the additional precipitation produced by the CTT-conditioned models cannot be consistently associated with either weak or intense rainfall, but varies with rainfall structure, forecast lead time, and the individual event. For the probabilistic forecasts, the two CTT–rainfall clusters respond differently to the cluster-dependent affine adjustment. The adjustment reduces CRPS mainly for low-intensity rainfall and selected events and lead times, whereas at higher intensities and in several events it increases CRPS. These findings indicate that CTT can improve precipitation detection and support rainfall-regime-based uncertainty characterization, but its effectiveness depends on the fusion architecture, rainfall regime, and event characteristics.

Contents

Preface	i
Abstract	iii
Nomenclature	vi
1 Introduction	1
1.1 The Importance of Precipitation Nowcasting	1
1.2 Challenges in Nowcasting Short-Duration Intense Rainfall	1
1.3 Beyond Radar-Based Nowcasting	2
1.4 Ensemble Nowcasting for Uncertainty Representation	2
1.5 Research Questions	3
2 Data sources and Background	4
2.1 Radar Data	4
2.2 Pre-processing of cloud top temperature	5
2.2.1 Spatial alignment and regridding	5
2.2.2 Temporal regridding	6
2.2.3 CTT Value Range Limiting and Missing Value Handling	6
2.3 Event selection	7
2.4 Dataset Split	8
2.5 CTT-Rainfall Regime Identification	8
2.5.1 Construction of Event-Level CTT-Rainfall Profiles	8
2.5.2 Clustering and Selection of the Number of Regimes	9
2.5.3 Characteristics of the Identified Regimes	10
3 Methodology	12
3.1 NowcastNet Framework	12
3.1.1 Model Overview	12
3.1.2 Evolution Network	12
3.1.3 Generative Network	13
3.2 Incorporating of CTT into NowcastNet	14
3.2.1 Input-level fusion	16
3.2.2 Feature-Level Fusion	16
3.2.3 CTT-Conditioned Generator	17
3.2.4 Model Summary	19
3.3 Cluster-Informed Ensemble Post-Processing	20
3.4 Evaluation Metrics	21
3.4.1 Deterministic results evaluation	21
3.4.2 Ensemble results evaluation	22
3.5 Training setup	23
4 Results	25
4.1 Preliminary Experiment	25
4.1.1 Checkpoint Initialization	25
4.1.2 Input-Level CTT Fusion	26
4.2 Model Results	27
4.2.1 Forecast Event on 2025-07-21 17:35	27

4.2.2	Forecast Event on 2025-09-04 17:00	28
4.2.3	Forecast Event on 2025-09-05 13:50	29
4.2.4	Overall Categorical Performance	31
4.3	Clustering Results	32
4.3.1	Ensemble results	34
4.3.2	Event-Based Spatial Analysis of the Cluster-Conditioned Ensembles	35
5	Conclusion	39
6	Recommendation	41
	References	43
A	Events summary	47
B	NowcastNet full structure	50
C	Additional events	52
D	CTT-Conditioned Model Architecture	56
D.1	CTT-conditioned evolution network	56
D.2	Generative encoder and CTT encoder	57
D.3	Noise projector and generative decoder	57
D.4	Temporal discriminator	58
D.4.1	Parameter summary	59

Nomenclature

Abbreviations

Abbreviation	Definition
CRPS	Continuous Ranked Probability Score
CRS	Coordinate Reference System
CSI	Critical Success Index
CTT	Cloud-Top Temperature
DTW	Dynamic Time Warping
FAR	False Alarm Ratio
GAN	Generative Adversarial Network
KNMI	Royal Netherlands Meteorological Institute
MAE	Mean Absolute Error
MSG	Meteosat Second Generation
PDF	Probability Density Function
POD	Probability of Detection
RHX	Maximum hourly rainfall intensity
RMSE	Root Mean Squared Error
SEVIRI	Spinning Enhanced Visible and Infrared Imager

Symbols

Symbol	Definition	Unit
b_c	Cluster-dependent noise shift factor	–
c	Cluster label or rainfall–CTT regime index	–
h_1, h_2, h_4, h_8	Decoder feature levels at different spatial resolutions	–
N	Number of ensemble members	–
q_τ	τ -quantile of the ensemble distribution	[mm h ^{−1}]
$q_{0.05}$	Lower ensemble quantile	[mm h ^{−1}]
$q_{0.95}$	Upper ensemble quantile	[mm h ^{−1}]
R	Rainfall intensity	[mm h ^{−1}]
r	Neighbourhood radius used in spatial verification	[pixel]
s_c	Cluster-dependent noise scale factor	–
T_{CTT}	Cloud-top temperature	[K]
t	Forecast lead time	[min]
x_t	Input radar field at time t	[mm h ^{−1}]
y_t	Observed rainfall field at time t	[mm h ^{−1}]
$\hat{y}_t$	Predicted rainfall field at time t	[mm h ^{−1}]
$\hat{y}_t^{(m)}$	The m -th ensemble member prediction	[mm h ^{−1}]
z	Standard Gaussian noise input	–
z'	Affine-transformed Gaussian noise input	–
α	Scaling coefficient for noise modification or CTT modulation strength	–

Symbol	Definition	Unit
β	Shift coefficient for affine noise modification	–
θ	Rainfall threshold used for event-based verification metrics	[mm h ⁻¹]

Introduction

1.1. The Importance of Precipitation Nowcasting

Reliable precipitation forecasts support decision-making in many weather-sensitive sectors, but their importance is particularly pronounced for water-related risk management at short lead times [1, 2]. Precipitation nowcasting aims to provide high-resolution forecasts of rainfall at short lead times, typically from a few minutes up to several hours ahead. Such forecasts are particularly important in the Netherlands, where a low-lying delta landscape, dense urban areas, artificial drainage networks, polders, and flood-protection infrastructure make water management highly sensitive to short-duration rainfall extremes.

The importance of this task is expected to increase under climate change. The Royal Netherlands Meteorological Institute (KNMI) reports that the intensity of extreme precipitation over one hour or less is likely to increase by approximately 14% for each degree of temperature rise [3]. Such short-duration extremes are particularly challenging for urban and regional water systems because they can produce rapid runoff responses and local drainage overloads, especially given that 59% of the Dutch land surface is prone to flooding according to the Netherlands Environmental Assessment Agency [4]. Therefore, accurate and reliable precipitation nowcasting is essential for operational flood forecasting and decision-support tool for managing rainfall-related risks over the Netherlands.

1.2. Challenges in Nowcasting Short-Duration Intense Rainfall

Precipitation nowcasting has gradually evolved from traditional approaches, including Numerical Weather Prediction (NWP) models [5, 6] and extrapolation-based methods using radar echoes and satellite observations [7, 8], toward deep learning-based models. Deep learning models are capable of learning complex nonlinear relationships from large datasets, which demonstrate considerable promise for nowcasting [9]. Recurrent and convolutional recurrent models such as ConvLSTM [10] and its successors improved the ability to learn motion and structural evolution directly from radar sequences. However, these models mainly focus on structural improvements, and while they demonstrate enhanced predictive capabilities, they still face some inherent limitations like strong blurriness at longer lead times. This limitation motivated the use of generative models, which aim not only to predict a single most likely future rainfall field, but also to generate sharper and more realistic precipitation structures. Within this development, NowcastNet [11] represents an important recent step because it combines a physical-evolution component with a conditional generative learning framework, aiming to preserve both the dynamical plausibility and the fine-scale structure of extreme precipitation nowcasts.

Nevertheless, even advanced deep generative nowcasting models remain challenged by short-duration intense rainfall. These events are often spatially localized, rapidly evolving, and strongly affected by convective processes, which makes them difficult to predict. These kind of short-duration rainfall extremes

are frequently associated with convective processes[12, 13], although not every heavy-rainfall event is necessarily convective. Convective rainfall is generated when warm, moist air rises rapidly, cools, and condenses into vertically developed clouds[14]. This process can produce localized rainfall cells with high intensities over short time periods. Typically, convective rainfalls are more spatially concentrated, shorter-lived, and more variable in intensity. These characteristics make short-duration extreme rainfall particularly difficult to nowcast. Many nowcasting methods rely on the recent radar-observed rainfall field and extrapolate it forward in time[15]. This approach works best when rainfall systems move smoothly and maintain their structure. However, convective cells may initiate, intensify, merge, split, or dissipate within tens of minutes. Small errors in the timing, position, or intensity of these cells can therefore lead to large forecast errors, especially for high-impact rainfall events.

1.3. Beyond Radar-Based Nowcasting

Current precipitation nowcasting studies mainly focus on improvements at the model-structure level, while the influence of data-level factors has received less attention[16, 17, 18]. However, precipitation is a complex atmospheric and microphysical process affected by multiple environmental factors, such as moisture, temperature, wind, and topography. Therefore, relying only on a single radar modality may be insufficient, especially for rainfall systems whose development is strongly controlled by changes in the surrounding atmospheric environment.

Recent studies have therefore started to explore the use of additional data sources together with radar observations. MM-RNN uses radar observations together with ground observation data as inputs to guide precipitation prediction, rather than relying only on radar sequences [19]. Similarly, MetNet uses radar and satellite observations as inputs for probabilistic precipitation forecasting over the continental United States [20]. Kim et al. further incorporated atmospheric factors such as low-level divergence and water-vapor-related variables into a deep-learning-based nowcasting model, showing that these additional predictors can help forecast the development of heavy rainfall systems [21].

The role of satellite-derived cloud properties that are closely related to the rapid development of convective rainfall has received less attention. From a physical perspective, lower cloud-top temperature (CTT) values usually indicate that the cloud top has developed into higher and colder atmospheric layers. For convective clouds, higher cloud tops often imply deeper vertical development, which is typically driven by stronger updrafts that continuously lift warm and moist air to upper levels. Such deep cloud columns provide favourable conditions and a longer vertical pathway for the growth of precipitation particles. As these particles grow and fall, they can contribute to heavy rainfall at the surface [22]. In addition, previous studies have shown that CTT can serve as an effective predictor for different precipitation types and can support the clustering of precipitation regimes [23].

1.4. Ensemble Nowcasting for Uncertainty Representation

A single deterministic nowcast provides only one possible future rainfall field and therefore cannot describe the full range of plausible rainfall developments. Probabilistic and ensemble nowcasting methods address this limitation by generating multiple possible future rainfall scenarios, allowing forecast uncertainty to be represented through the ensemble distribution[24, 25, 26].

In many radar-based and deep-learning nowcasting systems, ensemble members are generated through random perturbations, stochastic latent variables, or sampled noise fields[27, 28]. For example, PySTEPS generate ensemble members by perturbing the precipitation and motion fields, while deep generative models such as DGMR[29] and latent diffusion models sample future rainfall scenarios from latent random variables or Gaussian latent noise conditioned mainly on past observations. These ensemble members form an empirical predictive distribution, which makes it possible to estimate the likelihood of different rainfall developments and the probability of threshold exceedance at different lead times. The forementioned methods of perturbations can increase the diversity of ensemble members, but they are not necessarily well connected to the actual meteorological situation[30, 31]. More importantly, they may lack an explicit link to the physical processes that control rainfall uncertainty, such as storm initiation, displacement, intensi-

fication, and decay. Recent work such as ClusterCast [32] has shown that precipitation-type-dependent latent representations can be beneficial for generative nowcasting. Different rainfall regimes are represented in heterogeneous latent spaces rather than a single universal latent distribution. Therefore, linking physically meaningful precipitation regimes to forecast uncertainty shows potential for improving probabilistic forecasting and supporting operational decision-making.

1.5. Research Questions

This thesis aims to implement and train NowcastNet, as first described by [11] for precipitation nowcasting over the Netherlands and to investigate whether cloud-top temperature information can improve the forecasting of high-intensity rainfall and its associated uncertainty.

Following from the above introduction, the research questions of this MSc thesis are:

Sub-RQ1: How reliable and sharp are the baseline probabilistic forecasts of NowcastNet over the Netherlands when only radar information is used?

Approach: The baseline NowcastNet model will be evaluated without introducing CTT information. Its performance for selected high-intensity rainfall events will be assessed qualitatively through analysis of the forecast fields and quantitatively using the Critical Success Index (CSI), Probability of Detection (POD), and False Alarm Ratio (FAR).

Sub-RQ2: How can CTT-rainfall regimes provide physically meaningful information for conditioning ensemble uncertainty?

Approach: The CTT-rainfall relationships will be analyzed for the selected precipitation events. Events will be grouped into different regimes based on their event-level CTT-rainfall profiles. The resulting regimes will be compared in terms of cloud-top temperature, rainfall intensity, and their statistical characteristics to determine whether they represent distinguishable precipitation conditions.

Sub-RQ3: How does CTT-regime-dependent adjustment of Gaussian noise affect the probabilistic forecast skill and ensemble uncertainty of NowcastNet?

Approach: The latent Gaussian noise sampled by NowcastNet will be adjusted after model training using regime-dependent scale and shift parameters. Ensemble forecasts generated with the adjusted noise will be compared with ensembles generated using standard Gaussian noise. The effects of the adjustment on probabilistic forecast skill and ensemble uncertainty will be evaluated using CRPS, prediction intervals, and ensemble quantile bounds.

2

Data sources and Background

This chapter first introduces the radar rainfall and CTT datasets used in this study, together with the main data preparation procedures. These procedures include high-intensity rainfall event selection, radar–CTT pairing and preprocessing, and the division of the processed data into training and validation datasets. Based on the introduced datasets, the chapter subsequently examines the relationship between CTT and rainfall. Finally, the relevant background of NowcastNet and the role of CTT in precipitation nowcasting are presented to provide the foundation for the methodology developed in the following chapter.

2.1. Radar Data

The radar-based precipitation data used in this study are provided by the Royal Netherlands Meteorological Institute (KNMI) and can be downloaded from the KNMI Data Platform (KDP)[33]. This radar dataset covers Netherland and its surrounding area and is corrected by available data from automatic rain gauges. The gridded data have a 5-minute temporal resolution and a spatial resolution of $1 \text{ km} \times 1 \text{ km}$. Each pixel represents the accumulated precipitation over a 5-minute interval in millimetres (mm)[34]. These accumulations were converted to equivalent rainfall intensities in mm/h by multiplying them by 12.

The radar field was cropped to 512×512 pixels, as shown in figure 2.1, matching the input dimensions of NowcastNet while also retaining a sufficiently large surrounding area to provide spatial context for precipitation systems affecting the Netherlands.

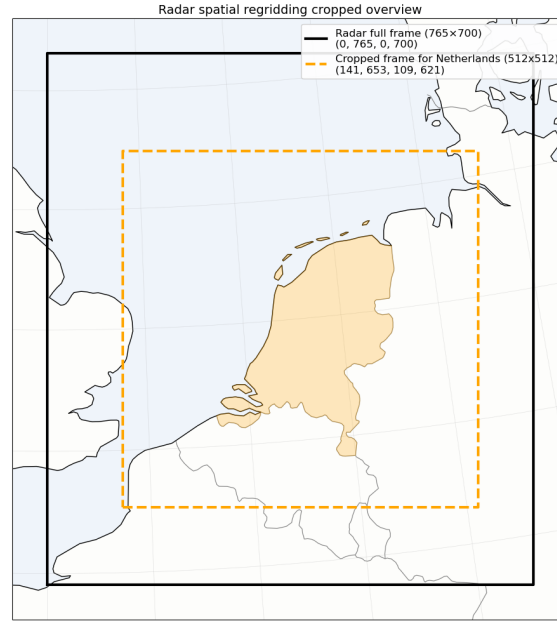


Figure 2.1: Spatial overview of the radar domain. The black rectangle represents the original radar coverage, and the orange dashed rectangle indicates the cropped 512×512 computational domain centred over the Netherlands.

2.2. Pre-processing of cloud top temperature

Cloud-top temperature data were retrieved from the KNMI MSG-SEVIRI product [35]. The dataset has a nominal spatial resolution of $3 \text{ km} \times 3 \text{ km}$ at the sub-satellite point and a temporal resolution of 15 minutes. Due to the oblique viewing geometry, the effective pixel size over the Netherlands is approximately $4 \text{ km} \times 7 \text{ km}$ [36].

The large differences between CTT and radar data in terms of spatial resolution, temporal resolution, and value range may lead to unbalanced gradient updates, causing variables with larger numerical ranges to dominate the learning process. Therefore, in order to ensure data consistency, the following sections will introduce the CTT preprocessing processes.

2.2.1. Spatial alignment and regridding

The CTT data are provided in the default MSG-SEVIRI geostationary projection. To spatially align the CTT fields with the radar data, the full-disk CTT fields shown (in Figure 3.2) were first cropped to the spatial extent of the radar domain. The cropped CTT data were subsequently transformed from the MSG-SEVIRI coordinate reference system (CRS) to the radar CRS. The target CRS and grid geometry were obtained from the KNMI radar dataset, and the radar-grid coordinates were reconstructed from the geographic corner coordinates and the number of rows and columns. The CTT data were then resampled onto the finer radar grid using `rioxarray` [37] with bilinear resampling.

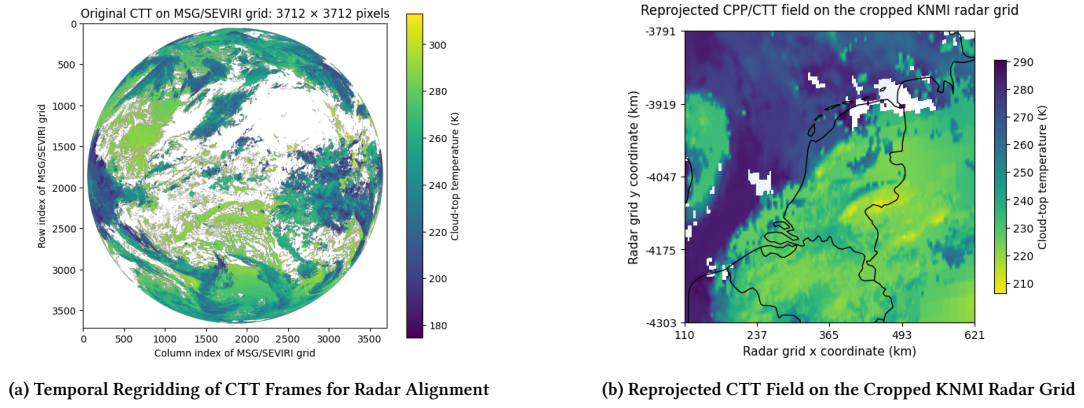


Figure 2.2: Temporal and spatial preprocessing of CTT data for alignment with the KNMI radar grid. Fig (a) shows the original CTT field on the MSG/SEVIRI grid, where the axes represent pixel indices. Fig (b) shows the reprojected CTT field on the cropped KNMI radar grid, where the x- and y-axes represent radar-grid coordinates in kilometres. White pixels indicate missing CTT values, corresponding to locations where no cloud cover is detected.

2.2.2. Temporal regridding

In contrast to radar precipitation data, which represent precipitation accumulated over a fixed time interval, CTT data are derived from satellite observations and represent the cloud state at approximately the time of acquisition. The MSG-SEVIRI instrument scans the Earth disk sequentially, such that different locations are observed at slightly different times within a single scan cycle. Consequently, the nominal timestamp assigned to a CTT product does not correspond exactly to the observation time over the Netherlands.

Because the CTT products have a temporal resolution of 15 minutes, whereas the radar data are available at 5-minute intervals, temporal alignment is required before the two datasets can be used together. Several temporal regridding approaches are available, including linear interpolation [38] and motion-aware interpolation [39]. However, interpolation produces intermediate CTT values that were not directly observed and may therefore introduce artificial temporal variability. More sophisticated interpolation approaches were considered unnecessary for the present application. A temporal replication strategy was therefore adopted to align the CTT data with the radar time steps. The detailed implementation of this procedure is described in Section ??.

2.2.3. CTT Value Range Limiting and Missing Value Handling

To align the numerical ranges of the CTT and radar inputs and to prevent gradient explosion or vanishing caused by magnitude disparities [40], inverse min-max scaling was applied to the valid CTT values as normalization process:

$$\tilde{T}_{CTT}(i, j) = \begin{cases} R_{\max} \cdot \frac{T_{\max}^K - T_{CTT}(i, j)}{T_{\max}^K - T_{\min}^K}, & \text{if } M_{CTT}(i, j) = 1, \\ 0, & \text{if } M_{CTT}(i, j) = 0, \end{cases} \quad R_{\max} = 128. \quad (2.1)$$

where $T_{CTT}(i, j)$ is the original CTT value in Kelvin, and $M_{CTT}(i, j)$ is a binary validity mask, with value 1 indicating a valid CTT observation and 0 indicating a missing or invalid value. $T_{\max}^K$ and $T_{\min}^K$ denote the maximum and minimum valid CTT values used for scaling, respectively. The parameter $R_{\max} = 128$ was selected to match the clipped radar rainfall range of $[0, 128]$ mm/h. The inverse transformation assigns larger scaled values to colder cloud tops, while preventing either input source from dominating the learning process solely because of differences in numerical magnitude.

Invalid or missing CTT pixels were assigned a value of zero. An additional binary mask channel was included to explicitly indicate valid and invalid CTT regions, allowing the model to learn how missing

CTT information should be treated[41].

2.3. Event selection

Training a deep-learning model on all rainfall events may lead to increasingly blurred predictions at longer lead times, as the model tends to smooth fine-scale spatial structures and regress toward the global mean of the training data [29]. This would result in a severe underestimation of high-intensity precipitation events. This section therefore introduces the method used to identify only the high-intensity rainfall events included in this study.

The event selection was based on validated daily station observations from the KNMI Data Platform - [42]. This dataset contains essential climate variables measured at KNMI weather stations from the 1990s onwards. It is an ongoing dataset, and the observations undergo a validation process before being released.

The radar dataset[33], produced using improved algorithms was only available from January 31, 2023 onward and it was further improved in methodologies from November 18, 2024. The dataset's quality was proven to be higher since the improvement from 2023[34], and all other datasets were therefore aligned with its period of availability. Consequently, this study used observations from 2023 to 2025.

Two parameters from the dataset were used to identify high-intensity rainfall events: the daily maximum hourly precipitation accumulation (RHX) and the hourly interval during which this maximum was measured (RHXH). RHX was used to construct an empirical exceedance probability plot, see in Fig 2.3, from which the threshold for high-intensity rainfall was selected. RHXH was then used to determine the relevant time windows around the selected rainfall events. These time windows define the radar and CTT data included in the dataset for this study.

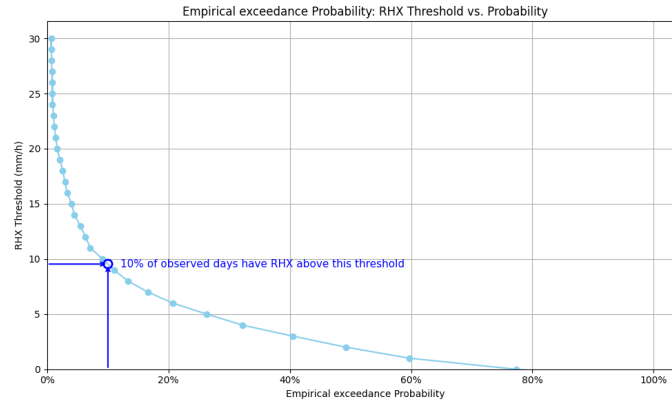
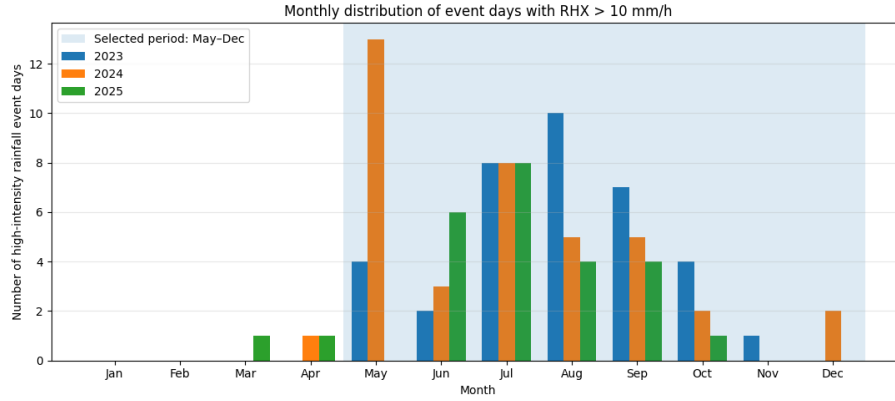


Figure 2.3: Empirical exceedance probability of daily maximum hourly precipitation (RHX) thresholds based on KNMI station observations from 2023 to 2025.

The threshold of 10 mm in one hour is determined from an exceedance analysis using the RHX observations from 2023 to 2025. The empirical exceedance probability represents the observed proportion of daily RHX values exceeding a given threshold. Therefore, it was used as a practical threshold for selecting relatively rare and intense rainfall events rather than for estimating an exact return period, which would require a substantially longer observational record. A distribution of the monthly occurrence counts of selected high-intensity rainfall event days is presented in Fig. 2.4. Based on this distribution, the study period was restricted to May–December, as the majority of selected events occurred within these months. This selection retains most of the relevant heavy-precipitation cases while excluding months with few or no high-intensity events.

Table 2.1: Monthly number of high-intensity rainfall event days for the RHX threshold of 10 mm in one hour from 2023 to 2025.

Year	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
2023	0	0	0	0	4	2	8	10	7	4	1	0
2024	0	0	0	1	13	3	8	5	5	2	0	2
2025	0	0	1	1	0	6	8	4	4	1	0	0

**Figure 2.4:** Monthly distribution of selected high-intensity rainfall event days.

2.4. Dataset Split

For machine-learning model, the dataset was divided into training and validation subsets. Here, the dataset was split using consecutive years to prevent data leakage[1].

As described in Section 2.3, the dataset consists of high-intensity rainfall events occurring between May and December from 2023 to 2025. After the radar-CTT pairing procedure described in Section 2.2.2, the paired samples were assigned to the training and validation sets according to their occurrence time. The resulting dataset division is summarized in Table 2.2.

Table 2.2: Overview of training and validation datasets used for different model groups.

Model group	Training period	Validation period	Training samples	Validation samples
Baseline model	May–December, 2023–2024	May–December, 2025	4548	876
CTT-integrated model variants	May–December, 2023–2024	May–December, 2025	3827	777

2.5. CTT--Rainfall Regime Identification

To provide regime-dependent information for the subsequent ensemble analysis, the relationship between cloud-top temperature (CTT) and rainfall intensity was examined at the event level. This analysis was treated as a preprocessing step: its purpose was to identify representative CTT–rainfall regimes and assign each rainfall event to one of these regimes. To avoid information leakage, the clustering structure was determined using only the training events.

2.5.1. Construction of Event-Level CTT--Rainfall Profiles

The preprocessed radar and CTT fields have a spatial resolution of 512×512 pixels and a temporal interval of 5 minutes. Directly clustering the complete spatial fields would result in a very high-dimensional dataset and would make the interpretation of the resulting clusters difficult. Therefore, each rainfall event was

compressed into an event-level statistical profile describing how rainfall intensity varies across different CTT ranges.

For each event, the valid CTT observations were divided into temperature bins with a width of 5 K. Within each bin, rainfall values below a predefined rainfall threshold were excluded to reduce the influence of near-zero and weak rainfall values. The remaining pixels were summarized using the mean rainfall intensity, selected rainfall quantiles, the rainy-pixel fraction, and a validity indicator. The resulting profile for event i is expressed as

$$\mathbf{X}_i = [\mathbf{x}_{i,1}, \mathbf{x}_{i,2}, \dots, \mathbf{x}_{i,L}], \quad (2.2)$$

where L denotes the number of CTT bins and the feature vector for bin l is defined as

$$\mathbf{x}_{i,l} = [\mu_{i,l}, q_{i,l}^{50}, q_{i,l}^{75}, q_{i,l}^{90}, f_{i,l}, v_{i,l}]. \quad (2.3)$$

Here, $\mu_{i,l}$ is the mean rainfall intensity, $q_{i,l}^{50}$, $q_{i,l}^{75}$, and $q_{i,l}^{90}$ are the corresponding rainfall quantiles, $f_{i,l}$ is the fraction of rainy pixels, and $v_{i,l}$ indicates whether the CTT bin contains valid rainfall observations.

2.5.2. Clustering and Selection of the Number of Regimes

Dynamic Time Warping (DTW) K-means was applied to the event-level profiles. Compared with conventional Euclidean K-means, DTW allows profiles with similar shapes to be grouped together even when their main rainfall response occurs within slightly different CTT ranges. This is useful because comparable CTT-rainfall relationships may be shifted along the temperature axis between different events.

The number of clusters was evaluated using the silhouette coefficient. For event i , the silhouette coefficient is defined as

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}, \quad (2.4)$$

where $a(i)$ is the average distance between event i and the other events in the same cluster, while $b(i)$ is the minimum average distance between event i and the events in another cluster. The mean silhouette score over all events was used to compare the candidate values of K .

Table 2.3: Silhouette scores for different numbers of CTT-rainfall clusters.

Number of clusters K	Silhouette score
2	0.2997
3	0.2643
4	0.2815
5	0.2332
6	0.2186
7	0.1888
8	0.1866

As shown in Table 2.3, $K = 2$ produced the highest silhouette score among the evaluated solutions. Although the score indicates that the separation between the regimes is moderate rather than complete, the two-cluster solution provided the clearest combination of statistical separation and physical interpretability. Therefore, two CTT-rainfall regimes were retained for the subsequent analysis.

2.5.3. Characteristics of the Identified Regimes

Following the identification of the CTT–rainfall regimes, the cluster labels were used to group the original rainfall observations. For each event e , valid rainfall pixels were selected using the same criteria adopted in the construction of the clustering profiles:

$$\Omega_e = \{(i, j) \mid 180 \text{ K} \leq \text{CTT}_e(i, j) \leq 310 \text{ K}, \quad R_e(i, j) \geq 1 \text{ mm h}^{-1}\},$$

where $R_e(i, j)$ denotes the observed rainfall intensity at pixel (i, j) in event e .

Figure 2.5 presents the statistical CTT–rainfall profiles of the two identified clusters.

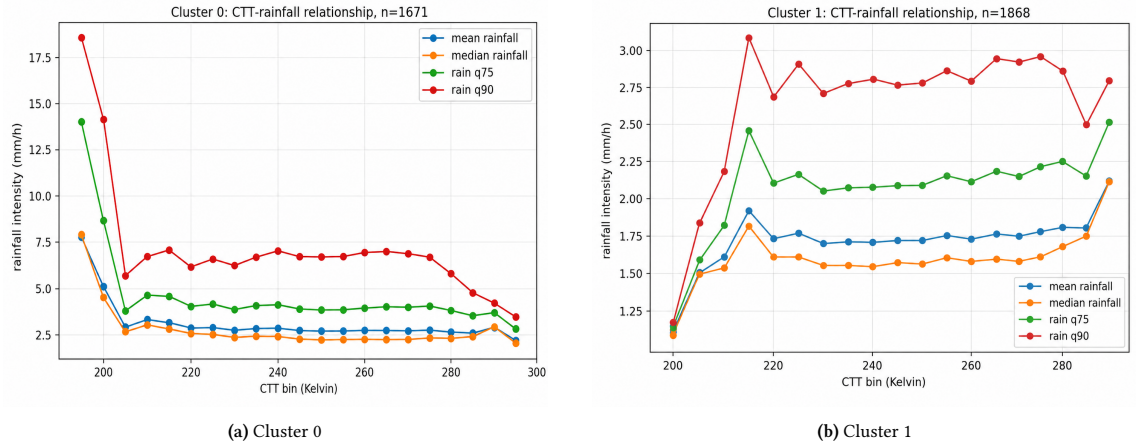


Figure 2.5: CTT–rainfall relationships for the two identified regimes.

The regimes show statistical patterns among the selected events. Cluster 0 is characterized by higher rainfall intensities, particularly within the colder CTT ranges. This regime is therefore interpreted as a cold-cloud-top, high-rainfall-intensity regime, which is consistent with the association between low CTT values, deep convective cloud development, and intense precipitation.

Cluster 1 shows a greater concentration of rainfall within relatively warmer CTT ranges and generally represents lower rainfall intensities and a narrower intensity range than Cluster 0. It is therefore interpreted as a warm-cloud-top, lower-rainfall-intensity regime.

The colder-cloud-top and higher-rainfall characteristics of Cluster 0 are likely associated with deep convective development. In such conditions, warm and moist air rises from the lower atmosphere, expands, and cools adiabatically. Once the air becomes saturated, water vapour condenses into cloud droplets and releases latent heat, which can enhance buoyancy and support continued vertical cloud growth. If the cloud develops sufficiently high, supercooled droplets and ice particles may form in the colder upper part of the cloud. The growth, collision, and melting of these hydrometeors can contribute to intense surface rainfall. Therefore, the low CTT values in Cluster 0 may indicate vertically developed convective clouds with strong updrafts and substantial precipitation production.

By contrast, the relatively warmer-cloud-top rainfall regime represented by Cluster 1 may be associated with shallower clouds or with warm-rain processes. When a cloud contains sufficient moisture and liquid water, rainfall can develop through the collision and coalescence of cloud droplets even without a very cold or highly developed cloud top. Larger droplets fall faster and collect smaller droplets, allowing raindrops to grow progressively. This may explain why measurable rainfall can occur under relatively warmer CTT conditions. Nevertheless, Cluster 1 should not be interpreted as a purely warm-cloud regime, because CTT alone does not provide sufficient information to determine the complete vertical cloud structure or precipitation microphysics.

The differences between the two CTT--rainfall regimes may be related to different atmospheric and cloud-microphysical processes. However, these processes cannot be directly confirmed from the CTT and radar observations alone and are therefore used only as a physical interpretation of the identified statistical patterns.

The identified clusters are subsequently used as fixed regime labels in the cluster-informed ensemble post-processing method introduced in Section 3.3.

3

Methodology

This chapter presents the methodological framework used in this study. It first introduces the baseline NowcastNet architecture and its training objectives. It then describes the temporal alignment of the CTT observations and the proposed strategies for incorporating CTT into the evolution and generative networks. Subsequently, the cluster-informed ensemble post-processing method is introduced. Finally, the evaluation metrics and experiment-specific training settings are presented.

3.1. NowcastNet Framework

This section describes the NowcastNet framework adopted as the baseline model in this study. The model architecture, input–output configuration, evolution network, generative network, and their corresponding training objectives are introduced in the following subsections.

3.1.1. Model Overview

NowcastNet is a physics-conditioned deep generative model developed for extreme-precipitation nowcasting. Traditional advection-based methods, such as PySTEPS and DARTS [43], mainly extrapolate radar echoes using estimated motion fields. In contrast, purely data-driven deep-learning models may produce blurry predictions or physically inconsistent evolution. NowcastNet addresses these limitations by combining a physically informed evolution with a conditional generative framework. It was shown to generate physically plausible and sharp multiscale precipitation nowcasts over large spatial domains and at lead times of up to 3 h [11].

Given a sequence of past radar observations $\mathbf{x}_{-T_0:0}$, NowcastNet predicts a sequence of future precipitation fields $\hat{\mathbf{x}}_{1:T}$, as illustrated in Figure 3.1. Its architecture consists of a deterministic evolution network and a stochastic generative network. The evolution network estimates the large-scale movement, growth, and decay of the precipitation system, while the generative network refines these predictions and introduces fine-scale stochastic structures.

In the configuration used in this study, nine historical radar frames at 5-minute intervals were provided as input, and the subsequent 20 radar frames were predicted. The forecast therefore covers a lead time of 100 minutes. Together, the input and target sequences comprise 29 consecutive radar time steps. This sequence configuration determines the temporal replication procedure applied to the CTT observations, which is described in Section 3.2.

3.1.2. Evolution Network

The evolution network estimates the deterministic, physically informed evolution of the precipitation system. It is based on a modified 2D continuity equation (see in Eq 3.1) in which rainfall evolution is

represented as the combination of advection by a motion field and local growth or decay through an intensity residual.

$$\frac{\partial \mathbf{x}}{\partial t} + (\mathbf{v} \cdot \nabla) \mathbf{x} = \mathbf{s}. \quad (3.1)$$

Here $\mathbf{x}$, $\mathbf{v}$ and $\mathbf{s}$ indicate radar fields of composite reflectivity, motion fields and intensity residual fields, respectively, and ∇ denotes the gradient operator. The tendency term $(\mathbf{v} \cdot \nabla) \mathbf{x}$ reveals the mass leaving the system, which is the first-order approximation of the difference before and after the advection operation.

A shared evolution encoder first extracts spatiotemporal features from the past radar sequence. These features are passed to two separate decoders. The motion decoder predicts future two-dimensional motion fields $\mathbf{v}_{1:T}$, while the intensity decoder predicts the corresponding intensity residual fields $\mathbf{s}_{1:T}$. The motion fields describe the displacement of precipitation patterns, whereas the intensity residuals represent changes in rainfall intensity associated with the growth and decay of precipitation cells (see Appendix B for the full NowcastNet structure).

Starting from the latest observed radar field $\mathbf{x}_0$, the evolution operator iteratively advects the precipitation field using the predicted motion field and subsequently adds the predicted intensity residual. The resulting deterministic sequence $\mathbf{x}_{1:T}''$ represents the large-scale location, movement, growth, and decay of the precipitation system.

The evolution network is trained before the generative network by minimizing a combination of an accumulation loss and a motion-regularization term. Because this stage uses supervised objectives rather than adversarial optimization, it is generally more stable than the subsequent generative training stage.

$$\mathcal{J}_{\text{evo}} = \mathcal{J}_{\text{accum}} + \lambda_{\text{motion}} \mathcal{J}_{\text{motion}} \quad (3.2)$$

where $\mathcal{J}_{\text{accum}}$ measures the discrepancy between the observed future radar fields and both the advected and fully evolved predictions over the forecast horizon. The motion-regularization term $\mathcal{J}_{\text{motion}}$ penalizes spatially irregular motion fields, with greater weight assigned to regions of heavier precipitation. The coefficient λ_{motion} controls the relative contribution of the regularization term. Full definitions of these terms follow the original NowcastNet formulation [11].

3.1.3. Generative Network

After the evolution network has been trained, its deterministic predictions are used to condition the generative network. The generative network refines the evolution-network output and recovers finer convective-scale structures that may not be represented by deterministic advection and intensity residuals alone.

The nowcast encoder processes the past radar observations together with the predictions from the evolution network and converts them into contextual feature representations. In parallel, a latent vector $\mathbf{z}$ is sampled from a standard Gaussian distribution and transformed by a noise projector into a spatial feature tensor. The nowcast decoder then combines the contextual features, the projected latent noise, and the deterministic evolution predictions. The evolution predictions are introduced at multiple spatial scales through spatially adaptive normalization, allowing the decoder to retain the large-scale physical evolution while generating finer-scale rainfall structures.

The generative network was subsequently trained using a combination of an adversarial objective and spatial-pooling regularization:

$$\mathcal{J}_{\text{generative}} = \beta \mathcal{J}_{\text{adv}} + \gamma \mathcal{J}_{\text{pool}} \quad (3.3)$$

Here, $\mathcal{J}_{\text{adv}}$ denotes the adversarial loss, which encourages the generated precipitation sequences to be indistinguishable from the observed sequences, while $\mathcal{J}_{\text{pool}}$ denotes the spatial pooling regularization loss. The coefficients β and γ control the relative contributions of the two loss terms [11].

In this process, Generative adversarial network (GAN) is used. During training, the generator produces future precipitation sequences, while a temporal discriminator attempts to distinguish generated sequences

from observed future radar sequences. The generator is optimized to produce forecasts that the discriminator classifies as realistic, whereas the discriminator is optimized to improve its ability to separate generated and observed sequences. Through this competing optimization process, the generative model is encouraged to produce sharper and more temporally consistent rainfall structures. In addition to the adversarial objective, NowcastNet applies spatial pooling regularization to constrain the generated precipitation at aggregated spatial scales. This regularization discourages the generator from producing fine-scale structures that appear realistic locally but are inconsistent with the broader precipitation distribution [11].

Adversarial training is generally less stable than supervised training because the generator and discriminator must remain sufficiently balanced. If the discriminator becomes too strong, the generator may receive weak or unhelpful gradients. And if the generator dominates, the discriminator may no longer provide an effective learning signal. Generative adversarial networks may also suffer from mode collapse, in which different latent-noise samples produce insufficiently diverse outputs. These difficulties make the generative stage sensitive to model initialization, learning rates, and the relative optimization of the generator and discriminator.

The output of the generative network is the final future precipitation sequence $\hat{\mathbf{x}}_{1:T}$. Because the latent vector is randomly sampled, multiple realizations of $\mathbf{z}$ can be used with the same historical radar sequence to generate different plausible future precipitation sequences. These sequences constitute an ensemble forecast and represent uncertainty in the future development of the precipitation field.

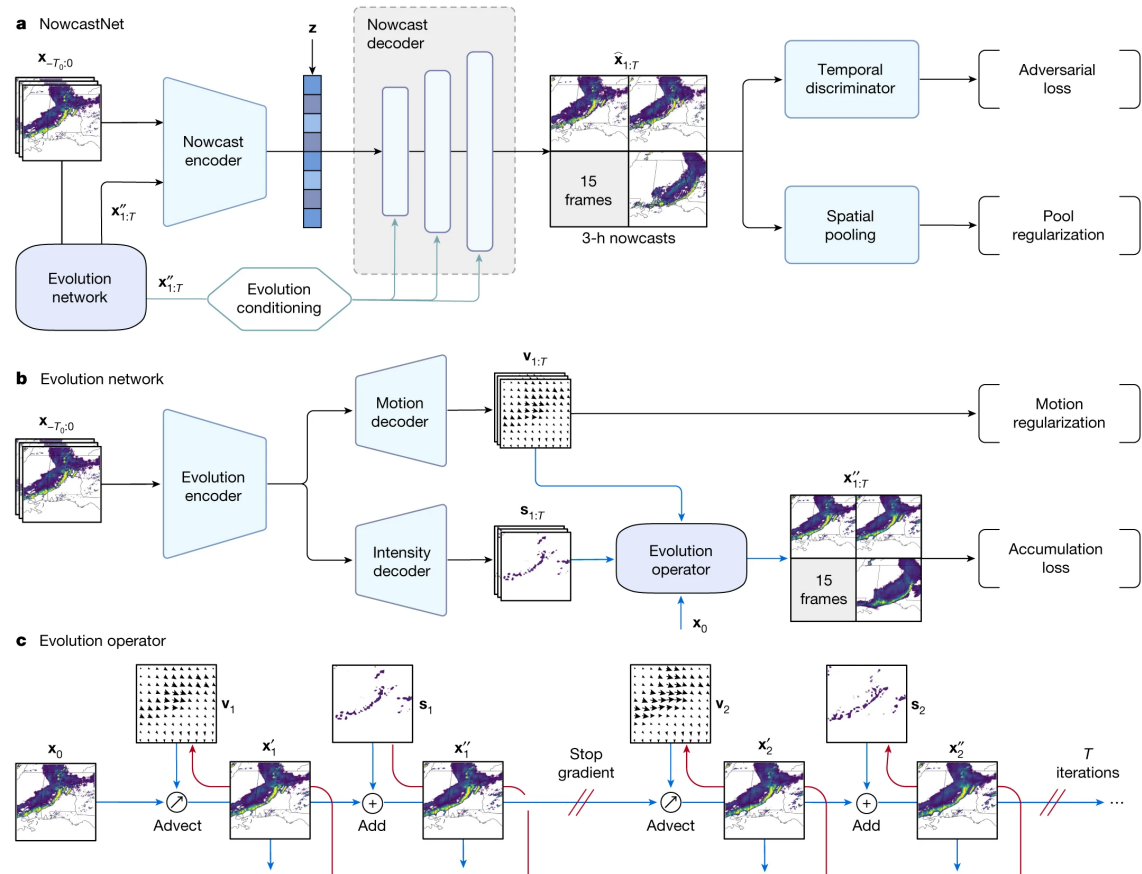


Figure 3.1: Illustration of the NowcastNet model structure, taken from NowcastNet published paper [11]

3.2. Incorporating of CTT into NowcastNet

This section first describes the temporal duplication of CTT data used to align the CTT observations with the 5-min radar sequence, and then investigates three strategies for incorporating cloud-top temperature

(CTT) into NowcastNet. Following the multi-view fusion taxonomy described by Mena et al. [44], the proposed approaches include two variants of input-level fusion, namely merged-input and grouped-input fusion, as well as feature-level fusion. These approaches differ in how rainfall and CTT information are organized and at which stage the two modalities interact.

Temporal Alignment and Replication of CTT

To incorporate CTT into NowcastNet, the CTT observations were temporally aligned with the 5-minute radar time grid. The timestamp of the CTT product corresponds to the nominal satellite product time rather than the actual scan time over the Netherlands. Based on the satellite scanning geometry, the Netherlands is observed approximately 10 minutes after the nominal timestamp[45]. Therefore, the CTT nominal time was shifted by 10 minutes to obtain an approximate effective scan time over the Netherlands. The shifted CTT timestamps were then aligned to the radar 5-minute time grid.

The relative time lag between a radar frame and the nearest effective CTT scan time can only fall into three possible cases: 0 minutes, 5 minutes, or 10 minutes. For each case, a different duplication strategy was used to expand the 10 CTT frames into 29 frames, so that the CTT sequence could be aligned with the radar sequence at 5-minute intervals.

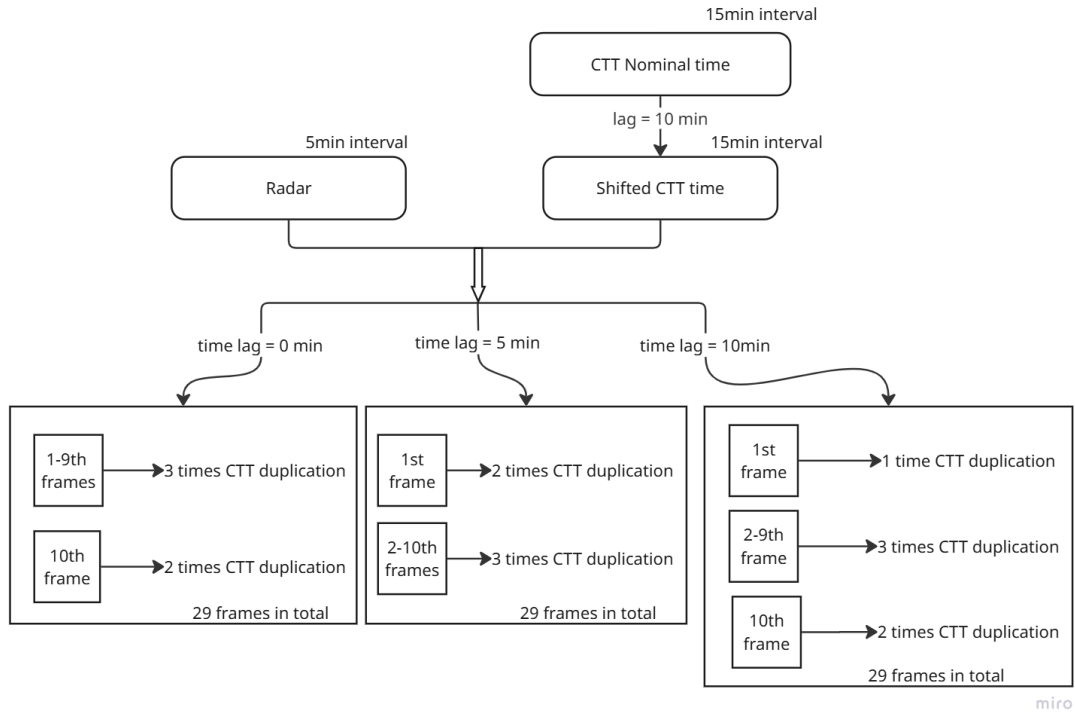


Figure 3.2: Temporal Regridding of CTT Frames for Radar Alignment

The advantage of duplication methods is that it avoids generating artificial intermediate CTT values. However, it assumes that the CTT field remains approximately unchanged between two satellite scans and rapid cloud-top cooling or warming associated with convective development may not be fully captured. This may introduce a small temporal mismatch between the radar and CTT inputs, especially during fast-evolving convective events, and could weaken the apparent relationship between CTT and rainfall intensity.

3.2.1. Input-level fusion

Two input-level fusion strategies were considered, differing only in how the radar and CTT frames were ordered along the channel dimension.

In the merged-input approach, all radar frames were first arranged as input channels, followed by all corresponding CTT frames. For T input time steps, the resulting tensor had the shape $[B, 2T, H, W]$, with the channel ordering

$$[R_{t_0}, R_{t_1}, \dots, R_{t_{T-1}}, C_{t_0}, C_{t_1}, \dots, C_{t_{T-1}}],$$

where R and C denote the radar rainfall and CTT fields, respectively. This arrangement groups the input frames according to their variable type, and the model learns the relationships between radar and CTT directly from the combined channel representation.

The grouped-input approach used the same input dimensions and network architecture, but arranged radar and CTT frames in an alternating temporal order [1]. Its input channels were ordered as

$$[R_{t_0}, C_{t_0}, R_{t_1}, C_{t_1}, \dots, R_{t_{T-1}}, C_{t_{T-1}}].$$

This arrangement places radar and CTT fields from corresponding timestamps next to each other in the channel dimension. However, it does not introduce a separate temporal axis or force the convolutional network to process each radar–CTT pair independently. The first convolution still combines all input channels jointly. This method is to test whether temporally aligned channel ordering influenced how the model learned to use CTT information, compared with grouping the inputs by variable.

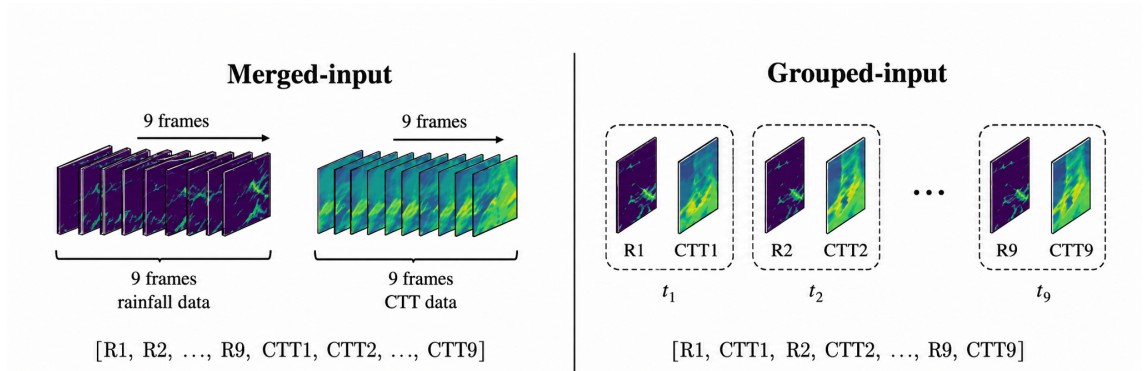


Figure 3.3: Illustration of two input-level CTT integration strategies. The merged-input model groups the channels by variable, with all radar frames followed by all CTT frames. In the grouped-input model, each CTT frame is placed directly after the radar frame from the same time step.

3.2.2. Feature-Level Fusion

The feature-level fusion block approach modifies the network architecture by adding a separate encoder branch for CTT. This design is motivated by multimodal fusion studies, where early interaction between related modalities can be beneficial [46, 47]. Radar and CTT features are extracted independently and fused after the second encoder block at the $H/2 \times W/2$ scale using channel-wise concatenation followed by a 1×1 convolution. This enables the model to learn a feature-level combination of the two modalities. Compatible pretrained weights are retained in the radar branch, while the CTT branch and fusion block are randomly initialised and learned during fine-tuning.

3.2.3. CTT-Conditioned Generator

This section describes how cloud-top temperature (CTT) information is incorporated into the generative network to modulate the generated precipitation fields. Specifically, the module takes decoder rainfall features and CTT-derived features as inputs, and uses spatial attention, channel attention, and learned modulation parameters to adjust the radar-derived feature representation. The corresponding modulation equations are then introduced.

As discussed in Section 2.5, cloud-top temperature (CTT) provides supplementary information related to cloud development and rainfall formation. In the generative network, CTT is therefore introduced as an auxiliary conditioning variable rather than as an independent predictor of future rainfall. Radar features remain the primary source of precipitation information, while CTT provides complementary cloud-development context that allows the network to selectively modulate spatially relevant regions and feature channels.

This design is inspired by the idea of multi-source feature fusion in precipitation nowcasting. FsrGAN[48] adopts both radar and satellite data as input sources and proposes a spatial-channel attention (SCA) module to adaptively fuse multi-scale spatiotemporal discriminative features from multi-source data. In that work, radar and satellite data are considered to have different characteristics. Instead of treating all features equally, the SCA module applies spatial-wise and channel-wise weighting to strengthen the discriminative learning ability and representational power of the network (see figure 3.4).

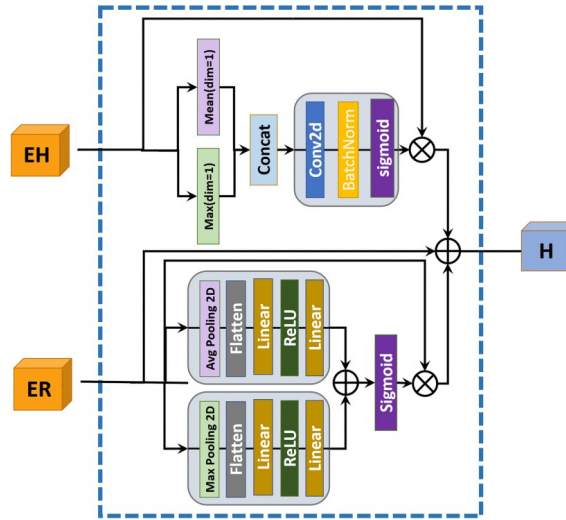


Figure 3.4: A novel spatial channel attention block SCA, figure taken from FsrGAN

Following this general idea, the proposed CTT-guided modulation block also uses spatial and channel-related information to guide feature fusion. The CTT features are used only to modulate the intermediate decoder rain feature maps. Therefore, CTT acts as conditional guidance for rainfall generation.

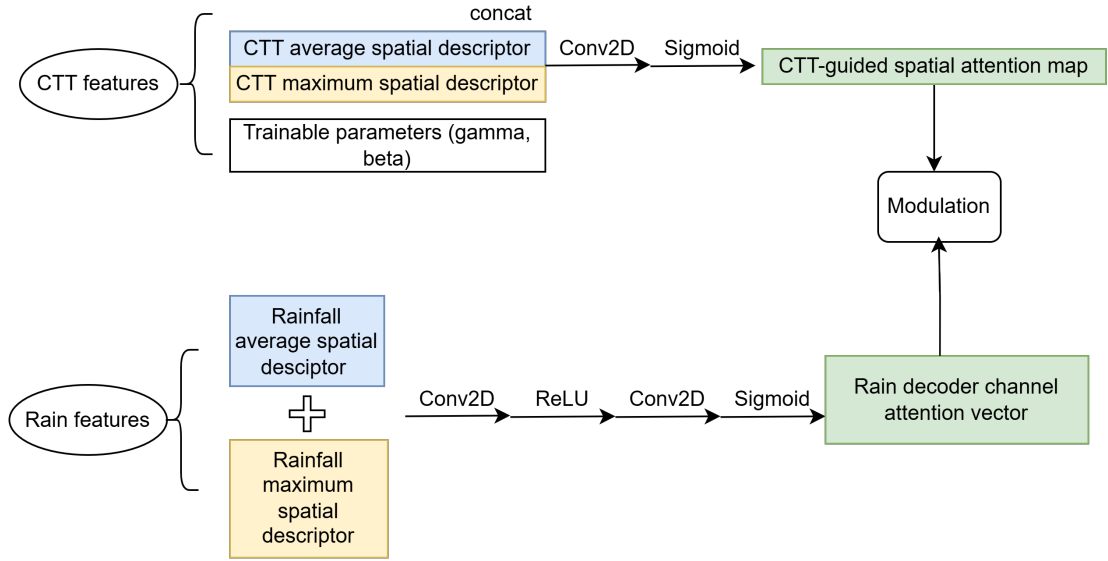


Figure 3.5: CTT-SCA Block

The modulation block in Figure 3.5 receives two feature maps as inputs. The first input is the rainfall-related feature map from the generative decoder. This feature map is an intermediate representation produced during the upsampling process. The second input is the CTT feature map extracted from the CTT encoder. Since both the decoder and the CTT encoder operate at multiple spatial scales, the CTT feature map corresponding to the current decoder resolution is selected. If the CTT feature map and the decoder feature map have different spatial resolutions, the CTT feature map is resized so that both have the same height and width. The selected CTT feature map is then used to generate attention weights and modulation parameters, which conditionally adjust the decoder rainfall feature.

Average pooling and max pooling are used to capture different aspects of the feature maps. Average pooling reflects the overall response, whereas max pooling emphasizes the strongest activations. Although both operations may discard part of the original information, using them together has been shown to provide complementary descriptors for attention estimation [49, 50, 51]. Their combination therefore offers a simple way to improve feature representation with little additional computational cost.

Therefore in the modulation part, the adjusted rain feature map equation is :

$$\Delta = (\gamma \odot F_{rain} + \beta) \odot s_{ctt} \odot a_f \quad (3.4)$$

where F_{rain} is the decoder rain feature map, s_{ctt} is the CTT spatial attention map that controls where the correction is applied, a_f is the decoder channel attention map that controls which rain feature channels are affected, and $\odot$ denotes element-wise multiplication. The final output is computed as:

$$F_{out} = F_{rain} + \alpha \Delta \quad (3.5)$$

where α is a learnable coefficient controlling the overall strength of the residual correction. Therefore, CTT does not replace the original rain feature, but provides an adaptive correction to guide rainfall generation.

Therefore, the proposed block extends the idea of spatial-channel attention to the generative network. Instead of treating all rain features equally, it allows the network to learn the importance of different spatial regions and feature channels according to CTT information. This is expected to help the generative network produce more realistic and physically consistent fine-scale rainfall fields.

The overall structure of generative network with CTT conditioned is presented in Fig 3.6.

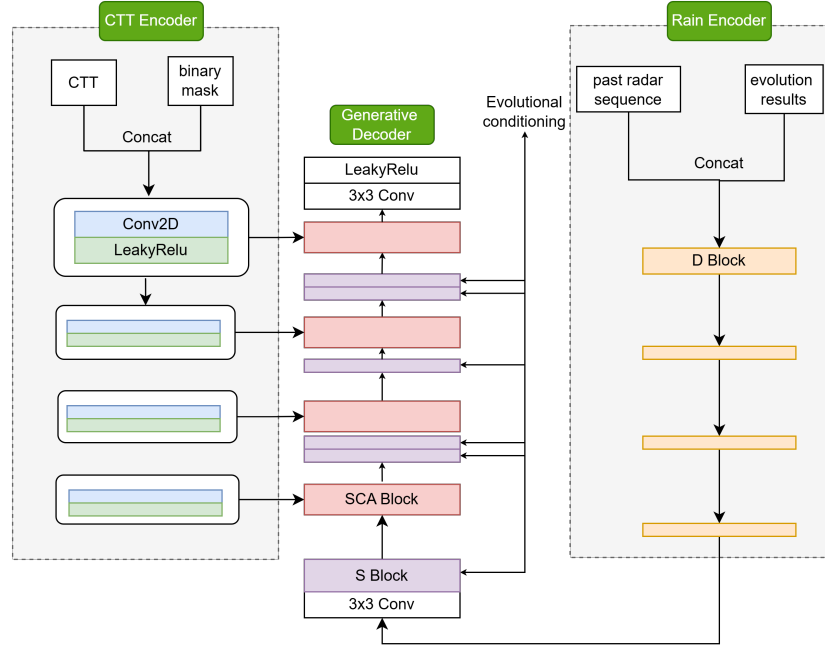


Figure 3.6: Enter Structure of the CTT-conditioned generative network. The past radar sequence and evolution results are first encoded to provide the main rainfall representation, while CTT and its binary mask are encoded separately by the CTT encoder. During the upsampling process of the generative decoder, the evolution result is injected through S-blocks as the main conditional information, and the corresponding multi-scale CTT features are introduced through CTT-guided SCA blocks to modulate the decoder rain features

3.2.4. Model Summary

For clarity, the model architectures described in Section 3.2 are summarized in Table 3.1.

Table 3.1: Comparison of CTT conditioning strategies in the evolution and generative networks.

Model	Conditioned CTT in Evolution Network	Conditioned CTT in Generative Network
Baseline	x	x
Merged-Input CTT	Merged input	x
Grouped-Input CTT	Grouped input	x
Half-Fusion	Fusion block	x
Full-Fusion	Fusion block	✓

The model variants are designed to evaluate how CTT information affects different stages of Nowcast-Net. The radar-only Baseline model is used as the reference configuration and does not include any CTT information.

The Merged-Input CTT and Grouped-Input CTT models introduce CTT only into the evolution network through direct input-level fusion. These variants are used to test whether simply providing CTT together with radar observations can improve the deterministic prediction of rainfall evolution.

The Half-fusion model also introduces CTT only into the evolution network, but uses a dedicated feature-fusion block instead of direct input merging. This comparison examines whether feature-fusion mechanism can make more effective use of CTT than simple input-level integration, particularly within the evolution network, which provides the deterministic rainfall prediction used as the basis for subsequent generative refinement.

The Full CTT-conditioned model extends the Half-fusion configuration by additionally introducing CTT-based conditioning into the generative network. It is therefore used to investigate whether providing CTT information to both the deterministic evolution stage and the stochastic generative refinement stage leads to further changes in forecast performance.

3.3. Cluster-Informed Ensemble Post-Processing

The CTT-rainfall regimes and their physical interpretation were introduced in Section 2.5. Here, the corresponding cluster labels are used as fixed inputs to adjust the latent Gaussian noise during ensemble generation. Specifically, different cluster labels are associated with different scale and shift transformations of the latent noise.

The identified clusters were used as fixed regime labels in the cluster-informed ensemble post-processing method. Within clusters, the event-level 90th percentile of rainfall intensity was then calculated as

$$q_e^{0.90} = Q_{0.90}(\{R_e(i, j) \mid (i, j) \in \Omega_e\}).$$

$Q_{0.90}$ was chosen because it was more responsive to the occurrence of strong rainfall compared with the event-level mean or median, while being less sensitive to isolated extreme pixels than the maximum rainfall intensity. It therefore provided a robust indicator of the upper part of the rainfall-intensity distribution within each event.

For each identified regime c , the event-level values were collected as

$$\mathcal{Q}_c = \{q_e^{0.90} \mid z_e = c\},$$

where z_e denotes the cluster label assigned to event e and $\mathcal{Q}_c$ denotes the empirical distribution profile assigned to cluster c .

These cluster-specific rainfall distributions were subsequently used to derive the scale and shift parameters of the post-processing transformation. Before introducing this transformation, the latent-noise sampling procedure of the original framework is first described.

In the original framework, the latent noise vector is sampled from a standard Gaussian distribution,

$$\mathbf{z} \sim \mathcal{N}(0, I), \tag{3.6}$$

where $\mathbf{z}$ denotes a latent variable sampled from a multivariate Gaussian distribution with zero mean and unit variance in each dimension. The sampled latent noise is then fused with the deterministic prediction feature within the deep generative network. Since the deterministic branch already captures the primary rainfall structure, including the spatial distribution and intensity patterns, the latent noise mainly represents unresolved stochastic variations that may influence future rainfall evolution.

Using the same Gaussian distribution for all events implicitly assumes that uncertainty characteristics are identical across different rainfall regimes. Under this assumption, events associated with strong convection, weak precipitation, cold cloud tops, or relatively flat rainfall responses are all represented by the same latent perturbation distribution. Such an assumption may be overly restrictive for precipitation nowcasting because different CTT-rainfall regimes can exhibit distinct rainfall intensity distributions and uncertainty characteristics.

To address this limitation, the proposed method uses clustered classes to determine the uncertainty of precipitation at the q90 quantile rainfall intensity of each distribution and uses this uncertainty to post-process the Gaussian prior. The modified noise is defined as follows:

$$z_c \mid c \sim \mathcal{N}(\mu_c, \sigma_c^2 I). \quad (3.7)$$

where c denotes the cluster label assigned to the event, $\mu_c = \text{shift}(c)$ and $\sigma_c = \text{scale}(c)$.

$$\text{scale}(c) = \left(\frac{\text{std}_c}{\text{std}_{\text{ref}}} \right)^\alpha, \quad \text{shift}(c) = \beta \frac{\text{mean}_c - \text{mean}_{\text{ref}}}{\text{std}_{\text{ref}}} \quad (3.8)$$

Here, mean_c and std_c denote the mean and standard deviation of the fitted rainfall distribution for cluster c . The reference statistics are computed as the average statistics over all fitted cluster distributions:

$$\text{mean}_{\text{ref}} = \frac{1}{K} \sum_{k=1}^K \text{mean}_k, \quad \text{std}_{\text{ref}} = \frac{1}{K} \sum_{k=1}^K \text{std}_k,$$

where K is the number of clusters.

The scale and shift process is used to control the spread of latent Gaussian noise sampling. The spread size is determined by equation 3.8 and tested to account for the uncertainty of each cluster.

The exponent parameter α was searched within $\alpha \in (0, 4]$, while the mean-shift parameter β was searched within $\beta \in (-5, 5)$. The parameter α controls the strength of the exponential cluster-dependent scaling. Increasing α widens the ensemble spread for one regime while simultaneously narrowing it for the other, thereby introducing a trade-off between the uncertainty representation and forecast performance of the two clusters. The parameter β controls the displacement of the mean of the post-processed latent-noise distribution and was allowed to take both positive and negative values.

The scale and shift transformation was applied as a post-processing step after model training, rather than being incorporated directly into the training architecture. Preliminary experiments in which scaled and shifted Gaussian noise was introduced during training produced ensemble forecasts that were nearly identical to those obtained using the original Gaussian noise. This may be related to the UNet-like structure of the generative network, in which repeated downsampling and upsampling may attenuate or smooth the effect of stochastic perturbations during feature compression and reconstruction. Consequently, applying scale and shift transformations to the latent noise during training may have only a limited influence on the generated ensemble variability.

3.4. Evaluation Metrics

The evaluation consists of two complementary components. First, deterministic forecast performance is assessed using categorical rainfall metrics for selected validation events. Second, the ensemble forecasts are evaluated in terms of prediction-interval width, regional CRPS, and the dependence of CRPS changes on observed rainfall intensity. Together, these analyses assess both forecast accuracy and the effect of the cluster-dependent post-transformation on probabilistic forecast behaviour.

3.4.1. Deterministic results evaluation

For the quantitative assessment of the models, test events are selected from the validation dataset where both CTT and radar observations are available. The candidate events are ranked from highest to lowest observed rainfall intensity, and overlapping events with no rainfall in the Netherlands are excluded from the test set. Only one event is retained from each calendar day for testing. After an event is selected, all remaining events from the same day are excluded. This reduces temporal redundancy because events initiated only 5 minutes apart share substantial overlap in their input and forecast periods.

To evaluate the short-term precipitation nowcasting accuracy, three neighborhood-based categorical metrics were used: Neighborhood Critical Success Index (Neighborhood CSI), Neighborhood Probability of Detection (Neighborhood POD), and Neighborhood False Alarm Ratio (Neighborhood FAR). These metrics are threshold-based. For a given rainfall intensity threshold, the predicted and observed precipitation

fields were first converted into binary event maps, where rainfall values equal to or above the threshold were classified as rainfall events.

Based on the binary event maps, hits (H), misses (M), and false alarms (F) were calculated. A hit indicates that a rainfall event was both predicted and observed. A miss indicates that an observed rainfall event was not predicted. A false alarm indicates that a rainfall event was predicted but not observed.

In this study, a neighborhood-based verification method was adopted for all three metrics. This is because precipitation nowcasting models may predict the correct rainfall structure while showing small spatial displacement errors relative to the observation. In such cases, a strict grid-cell-by-grid-cell comparison may over-penalize the model. Therefore, a neighborhood radius of 2 km was used to allow limited spatial tolerance between the predicted and observed rainfall events.

The Neighborhood CSI, POD, and FAR are defined as:

$$CSI = \frac{H}{H + M + F},$$

$$POD = \frac{H}{H + M},$$

$$FAR = \frac{F}{H + F}.$$

CSI measures the overall event-based prediction skill by considering both missed events and false alarms. POD measures the ability of the model to detect observed rainfall events. FAR measures the false-alarm tendency of the model. CSI and POD range from 0 to 1, where higher values indicate better performance. FAR also ranges from 0 to 1, but lower values indicate better performance.

3.4.2. Ensemble results evaluation

The ensemble evaluation was designed to determine not only whether the proposed cluster-dependent affine transformation changes probabilistic forecast performance, but also how and under which conditions these changes occur. The evaluation therefore consists of three complementary components.

All experiments were conducted using ensembles of $N = 50$ members at lead time 90 minutes (the level of uncertainty grows with longer lead time). Two ensemble-generation strategies were compared: a standard Gaussian noise ensemble and a cluster-dependent affine-transformed noise ensemble. For a fair paired comparison, both ensembles were generated from the same Gaussian noise samples, with the cluster-dependent scale and shift applied only to the transformed ensemble.

First, the lower and upper parts of the ensemble distribution are examined spatially using the 5th- and 95th-percentile forecast maps for one representative event from each CTT-rainfall cluster.

Second, an exploratory CTT-based regional analysis on all validation events evaluates whether the CRPS response differs between cold and warm cloud-top conditions. Third, the CRPS change is quantified as a function of radar-observed rainfall intensity and is compared between the two CTT-rainfall clusters. This is also done on all validation events. Together, these experiments describe how the post transformation changes ensemble uncertainty, under which cloud-top conditions these changes improve or degrade probabilistic skill, and at which rainfall intensities the corresponding CRPS response occurs.

Ensemble quantile maps

To visually compare the spatial characteristics of the ensemble distribution, the 5th and 95th percentiles are calculated from the 50 ensemble members at each pixel and lead time. The two quantiles are used to represent the lower and upper tails of the forecast distribution, showing how the predicted rainfall extent and intensity vary across the ensemble. For each selected event, the observation, the 5th-percentile forecast, and the 95th-percentile forecast are shown from left to right.

Regional CRPS analysis

For an ensemble prediction $\{x_1, x_2, \dots, x_N\}$ and radar observation y , the CRPS is defined as

$$\text{CRPS} = \frac{1}{N} \sum_{i=1}^N |x_i - y| - \frac{1}{2N^2} \sum_{i=1}^N \sum_{j=1}^N |x_i - x_j|. \quad (3.9)$$

Lower CRPS values indicate better probabilistic forecast performance, as the score evaluates the discrepancy between the full predictive distribution and the observation while balancing forecast accuracy and ensemble dispersion.

The CTT regions are defined using the final input CTT frame:

$$\text{cold} : \text{CTT}_{\text{last}} \leq 245 \text{ K}, \quad \text{warm} : \text{CTT}_{\text{last}} > 245 \text{ K}. \quad (3.10)$$

The radar-observed rainfall regions are defined as

$$\text{rain} : y \geq 1 \text{ mm h}^{-1}, \quad \text{below-threshold} : y < 1 \text{ mm h}^{-1}. \quad (3.11)$$

The resulting regions are cold-rain, warm-rain, cold-below-threshold, and warm-below-threshold.

For method m and region r , the regional mean CRPS is

$$\overline{\text{CRPS}}_m^{e,r,t} = \frac{1}{|\Omega_{e,r,t}|} \sum_{p \in \Omega_{e,r,t}} \text{CRPS}_m^{e,t,p}, \quad (3.12)$$

where $\Omega_{e,r,t}$ denotes the pixels belonging to region r . The regional CRPS change is

$$\Delta \text{CRPS}^{e,r,t} = \overline{\text{CRPS}}_{\text{Gaussian}}^{e,r,t} - \overline{\text{CRPS}}_{\text{Post}}^{e,r,t}. \quad (3.13)$$

Positive values indicate improved CRPS after post-processing, whereas negative values indicate deterioration.

CRPS change across observed rainfall intensity

Finally, the pixel-wise CRPS change is analysed as a function of the radar-observed rainfall intensity. The evaluated pixels are grouped into observed-rainfall intensity bins. For cluster c and intensity bin k , the mean CRPS change is calculated as

$$\overline{\Delta \text{CRPS}}^{c,k} = \frac{1}{|\mathcal{B}_{c,k}|} \sum_{(e,t,p) \in \mathcal{B}_{c,k}} \Delta \text{CRPS}^{e,t,p}, \quad (3.14)$$

where $\mathcal{B}_{c,k}$ contains the validation pixels assigned to cluster c whose observed rainfall falls within bin k .

3.5. Training setup

All formal experiments were conducted on DelftBlue, the Delft High Performance Computing Centre [52], using an NVIDIA Tesla A100 GPU with 80 GB of memory. The training setup generally followed the original NowcastNet configuration [53]. However, smaller batch sizes were adopted because of GPU-memory limitations, depending on the memory requirements of each model variant.

The main training configurations of the five generative model variants are summarized in Table 3.2.

Table 3.2: Training configurations used for the five generative model variants included in the final comparison.

Parameter	Value
Past frames, T_{in}	9
Forecast frames, T_{out}	20
Rainfall clipping threshold	128 mm h ⁻¹
Adversarial-loss weight, α	6
Pool-loss weight, β	20
Latent samples, k	4
Pooling kernel / stride	5 / 2
Maximum epochs	50 for Baseline; 100 for the other variants
Early-stopping patience	50 for Baseline; 100 for the other variants
Generator learning rate	10^{-6}
Discriminator learning rate	10^{-7}
Adam (β_1, β_2)	(0, 1)

The evolution-network models were initialized using the pretrained weights released with the original NowcastNet model, as described in Section 3.1.2. All compatible layers of the models in Table 3.1 were initialized with these weights. This approach was considered appropriate because the evolution network is trained with a supervised loss defined directly between predicted and observed rainfall fields and is therefore generally less sensitive to initialization than the adversarial generative network.

In contrast, all modified generative models were trained from scratch using the same initialization scheme. The original generative checkpoint is fully compatible only with the baseline architecture. For the CTT-based models, loading only the shared pretrained layers while randomly initializing the newly introduced inputs or fusion modules could create unequal initialization conditions. Training the modified models from scratch therefore provided a more consistent basis for comparing the different CTT-integration strategies.

The generative models reached a sufficiently stable state for comparative evaluation. The original NowcastNet generative network was trained for 5×10^5 optimization iterations, whereas one epoch in the present experiments required approximately 50 minutes. This computational cost limited the feasibility of substantially extending the training duration.

4

Results

This chapter first presents two preliminary experiments results on checkpoint initialization and input-level CTT fusion, as introduced in Chapter 3.2.1 and 3.5 . These experiments provide the basis for selecting the subsequent training and model-comparison strategies. Three representative rainfall events selected in Section 3.4.1 are then analysed in detail. The comparison considers both qualitative aspects, including forecast visualizations and event-specific spatial characteristics, and quantitative performance based on the evaluation metrics. Finally, the clustering results are presented, followed by the ensemble evaluation and event-based spatial analysis of the proposed post-processing method, as described in 3.4.2.

4.1. Preliminary Experiment

This section first presents two preliminary experiments used to establish the training and model-comparison strategy. The first evaluates whether loading the pretrained generative checkpoint improves the unchanged NowcastNet baseline. The second evaluates whether two simple input-level CTT fusion approaches improve forecast performance. The performance of the feature-level CTT-conditioned models is then evaluated using aggregated metrics and event-based forecast analyses.

4.1.1. Checkpoint Initialization

A preliminary experiment was conducted to examine the potential benefit of checkpoint loading for the unchanged generative baseline. One validation event from 21 July 2025 was evaluated at a rainfall threshold of 10 mm h^{-1} using CSI, POD, and FAR, as defined in Section 3.4.1. Three strategies were compared: training from scratch, direct inference using the pretrained checkpoint, and fine-tuning from the pretrained checkpoint. The results are shown in Table 4.1.

Table 4.1: Preliminary comparison of generative-network checkpoint loading strategies for the validation event on 21 July 2025.

Model / training strategy	CSI	POD	FAR
EvolutionNet baseline	0.1734	0.2717	0.7718
Generative model trained from scratch	↓ 0.1719	↓ 0.2409	↓ 0.5941
Generative model using pretrained weights without further training	↓ 0.1334	↓ 0.2449	↑ 0.8315
Generative model fine-tuned from pretrained weights	↑ 0.2007	↑ 0.3715	↓ 0.7547

Red values indicate improvement relative to the EvolutionNet baseline. Higher CSI and POD values indicate better performance, whereas lower FAR values indicate better performance.

The preliminary experiment showed that fine-tuning from the pretrained checkpoint improved the unchanged generative baseline. However, the same initialization could not be applied consistently to the modified CTT-conditioned architectures because their newly introduced components were not included in the original checkpoint. To maintain a consistent comparison between CTT-integration strategies, the checkpoint experiment was therefore used only to inform the training setup and was not included in the main model comparison.

4.1.2. Input-Level CTT Fusion

Before examining the spatial characteristics of the forecasts, the prediction ranges of the two input-level fusion variants are first compared with the observations across three representative events (See in table 4.2).

Table 4.2: Prediction ranges of the input-level fusion models and observations.

Event	Model	Prediction range	Observation range
4417	Merged input CTT	[0, 37.76]	[0, 128.00]
	Grouped input CTT	[0, 33.39]	
4818	Merged input CTT	[0, 38.53]	[0, 105.96]
	Grouped input CTT	[0, 26.02]	
4858	Merged input CTT	[0, 32.20]	[0, 128.00]
	Grouped input CTT	[0, 24.08]	

The generative outputs of the input-level fusion models are strongly compressed compared with the observations. For the three selected events, Merged input CTT and Grouped input CTT only produce maximum intensities of approximately 24–39 mm/h, while the observed maxima reach about 106–128 mm/h. This indicates that directly injecting CTT at the input level fails to preserve high-intensity precipitation signals in the generative prediction.

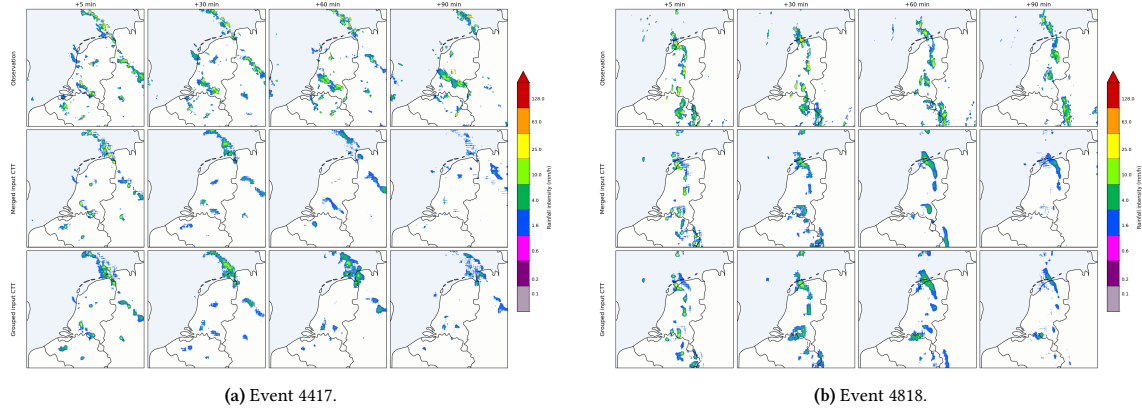


Figure 4.1: Qualitative examples comparing observations with the two input-level CTT fusion variants. The Merged-input CTT model corresponds to the concatenate setting, while the Grouped-input CTT model corresponds to the stack setting. Both variants partly retain some precipitation locations, but the predicted rainfall fields are more fragmented and spatially sparse than the observations, especially at longer lead times.

Figure 4.1a shows that both input-level fusion variants produce weak and fragmented rainfall patterns. The Merged-input CTT model retains some approximate precipitation locations at early lead times, whereas the Grouped-input CTT model misses more rainfall structures; in both cases, the predictions become increasingly sparse at longer lead times. This suggests that directly introducing CTT at the input level does not preserve a reliable spatial rainfall structure.

4.2. Model Results

In the following section, the forecasts for individual events, as described (in Section 3.4.1), are shown along with their metrics evaluation.

4.2.1. Forecast Event on 2025-07-21 17:35

Event 4417 is characterized by a central-to-northern precipitation area within the Netherlands, accompanied by several scattered convective cells in the south. From the observations, 2025-07-21 17:35 event shows a clear evolution of rainfall structures across different lead times. A line sloping around 45 degrees cuts diagonally from the North Sea into central Germany from the top left corner. Meanwhile, there is a flat, horizontally shaped rainfall structure from around Rotterdam to Eindhoven. Over time, this flat, horizontally shaped rainfall structure grows bigger and its influence is wider, whereas influence of the line shaped rainfall shrinks but still keeping its high intensity rainfall core.

Over the larger domain, including western Germany and regions near Belgium, all three models produce false rainfall predictions. They tend to generate an overly continuous rainfall band outside the eastern Dutch boundary, extending into western and south-western Germany, where no corresponding rainfall is observed. Figure 4.2 presents the observed rainfall field and the corresponding forecasts produced by the three models at different lead times.

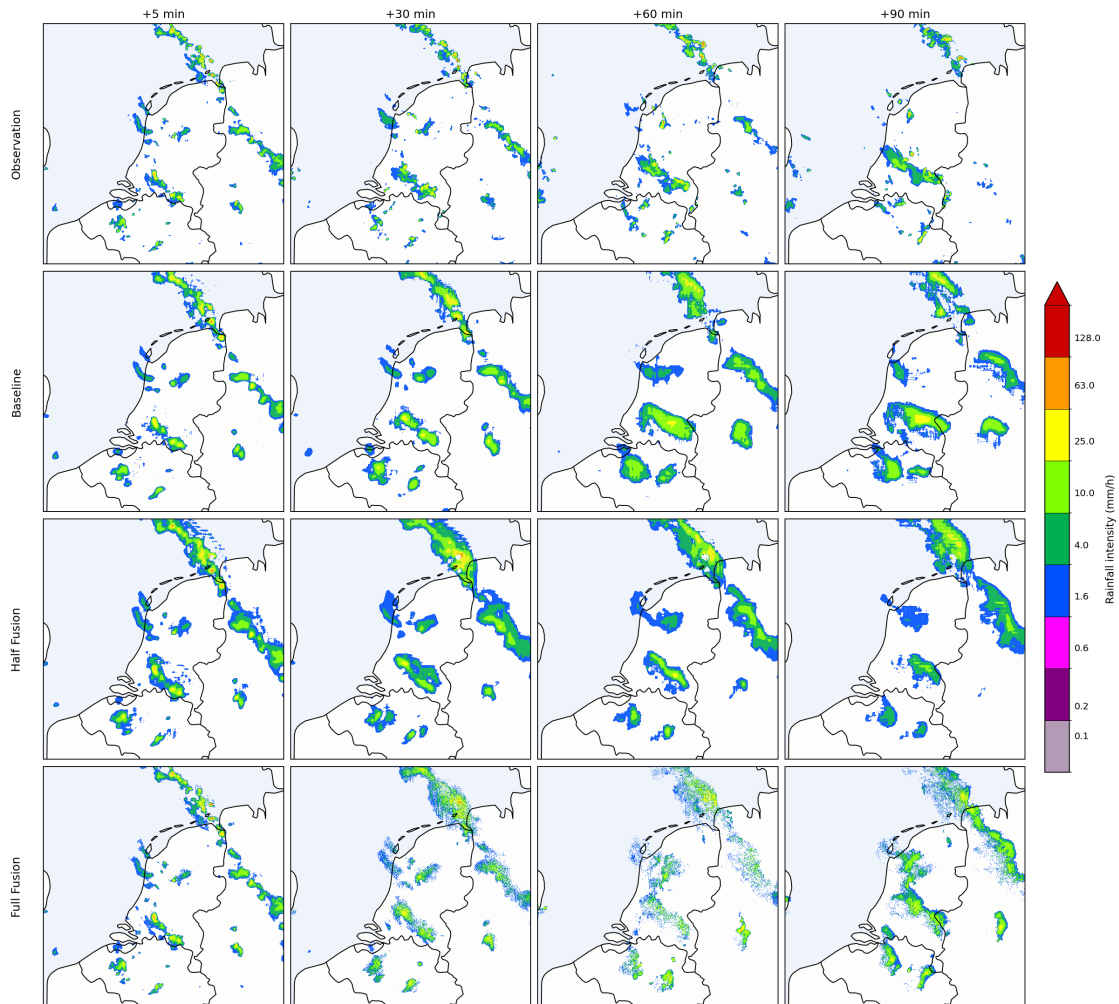


Figure 4.2: Qualitative comparison for event 4417 at 2025-07-21 17:35. Within the Dutch boundary, the observed rainfall has a central-northern precipitation area together with scattered convective cells in the south.

The prediction results suggest a trade-off between rainfall detection and false alarms, as shown in table 4.3. The baseline model provides the best balance at the lower and medium rainfall thresholds, achieving the highest CSI at 1 and 10 mm/h and the lowest FAR across all three thresholds. In contrast, Full Fusion detects a larger proportion of rainfall, achieving the highest POD at all thresholds and the highest CSI at 20 mm h⁻¹. This indicates that Full Fusion captures a larger proportion of the observed rainfall area, particularly for heavier precipitation, but at the cost of more false alarms. Half Fusion shows an intermediate response. Its POD at 10 and 20 mm/h is higher than that of the baseline, although the improvement is smaller than that achieved by Full Fusion and is also accompanied by an increase in FAR. Overall, this event indicates that improved detection of intense rainfall does not necessarily result in better performance across all categorical metrics.

Table 4.3: Categorical metrics for event 4417 over the 0–90 min lead-time window.

Event	Model	CSI (↑)			POD (↑)			FAR (↓)		
		thr=1	thr=10	thr=20	thr=1	thr=10	thr=20	thr=1	thr=10	thr=20
4417	Baseline	0.410↑	0.261↑	0.124	0.684	0.417	0.154	0.506↓	0.637↓	0.683↓
	Half fusion	0.365	0.237	0.139	0.679	0.428	0.201	0.570	0.673	0.718
	Full fusion	0.407	0.246	0.190↑	0.756↑	0.590↑	0.383↑	0.547	0.716	0.747

Note. Metrics are averaged over the 0–90 min lead-time window. Red values indicate the best-performing model for each metric and threshold. For CSI and POD, higher values are better; for FAR, lower values are better.

4.2.2. Forecast Event on 2025-09-04 17:00

For event 4818, Full Fusion shows the strongest overall detection performance, particularly at the 1 mm/h and 20 mm/h thresholds compared to other models, as shown in Table 4.4.

Table 4.4: Categorical metrics for event 4818 over the 0–90 min lead-time window.

Event	Model	CSI (↑)			POD (↑)			FAR (↓)		
		thr=1	thr=10	thr=20	thr=1	thr=10	thr=20	thr=1	thr=10	thr=20
4818	Baseline	0.432	0.251	0.083	0.796	0.465	0.113	0.518	0.664↓	0.805
	Half fusion	0.386	0.262↑	0.149	0.815↑	0.477	0.218	0.578	0.674	0.715↓
	Full fusion	0.462↑	0.254	0.182↑	0.809	0.534↑	0.310↑	0.491↓	0.701	0.761

Note. Metrics are averaged over the 0–90 min lead-time window. Red values indicate the best-performing model for each metric and threshold. For CSI and POD, higher values are better; for FAR, lower values are better.

The spatial forecasts in Figure 4.3 help explain these quantitative differences.

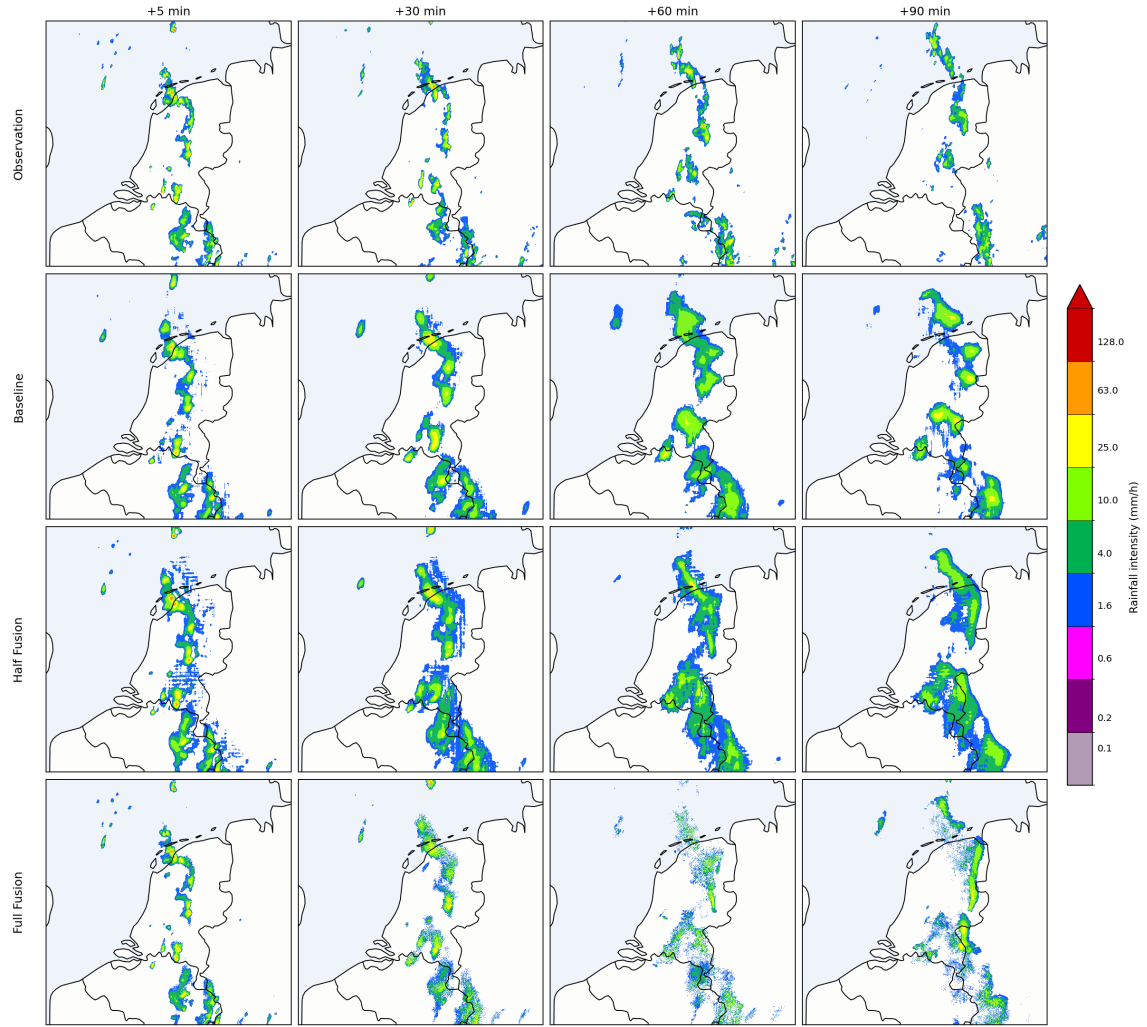


Figure 4.3: Qualitative comparison for event 4818 at 2025-09-04 17:00. The observations show a narrow elongated rainfall band over the Netherlands.

In this event, the observed rainfall appears as a relatively narrow, line-shaped precipitation band extending from the northern to the central and southern Netherlands, with several embedded rainfall cores. The system generally moves northward and becomes increasingly displaced and fragmented at longer lead times. The baseline broadly widened the rainfall band, but generates smoother and wider rainfall patches and underestimates the strongest embedded cores. Half Fusion maintains the elongated structure but is dominated by low-to-moderate rainfall intensities, making the stronger cores less pronounced. Full Fusion provides a clearer representation of the main rainfall band and assigns higher rainfall intensities to the core regions. This spatial behaviour is consistent with the categorical metrics. The stronger and broader rainfall predictions produced by Full Fusion contribute to its higher POD and CSI at the stronger thresholds, while additional predicted rainfall outside the observed band contributes to its higher FAR at 10 and 20 mm/h.

4.2.3. Forecast Event on 2025-09-05 13:50

For event 4858, Full Fusion achieves the strongest CSI-based performance across all three rainfall thresholds and improves the detection of moderate and heavy rainfall. However, this improvement is accompanied by a substantially higher false-alarm tendency at the 10 and 20mm/h thresholds, as shown in Table 4.5.

Table 4.5: Categorical metrics for event 4858 over the 0–90 min lead-time window.

Event	Model	CSI ($\uparrow$)			POD ($\uparrow$)			FAR ($\downarrow$)		
		thr=1	thr=10	thr=20	thr=1	thr=10	thr=20	thr=1	thr=10	thr=20
4858	Baseline	0.299	0.088	0.033	0.569 $\uparrow$	0.122	0.055	0.624	0.674 $\downarrow$	0.298 $\downarrow$
	Half fusion	0.261	0.122	0.090	0.469	0.193	0.171	0.642	0.814	0.697
	Full fusion	0.343 $\uparrow$	0.131 $\uparrow$	0.098 $\uparrow$	0.529	0.227 $\uparrow$	0.182 $\uparrow$	0.546 $\downarrow$	0.803	0.849

Note. Metrics are averaged over the 0–90 min lead-time window. Red values indicate the best-performing model for each metric and threshold. For CSI and POD, higher values are better; for FAR, lower values are better.

Full Fusion achieves the highest CSI at all three thresholds and the highest POD at 10 and 20 mm/h. At the 1 mm/h threshold, it also produces the lowest FAR, whereas the baseline achieves the highest POD. At the two higher thresholds, however, the baseline retains the lowest FAR, while Full Fusion produces the highest FAR. Half Fusion improves CSI and POD relative to the baseline at the higher thresholds, but it does not achieve the best value for any of the evaluated metric–threshold combinations. These results indicate that Full Fusion improves the detection and spatial overlap of moderate and heavy rainfall, but does so by predicting more intense rainfall in locations where it is not observed.

The spatial forecasts in Figure 4.4 provide the previous categorical analyses with more details.

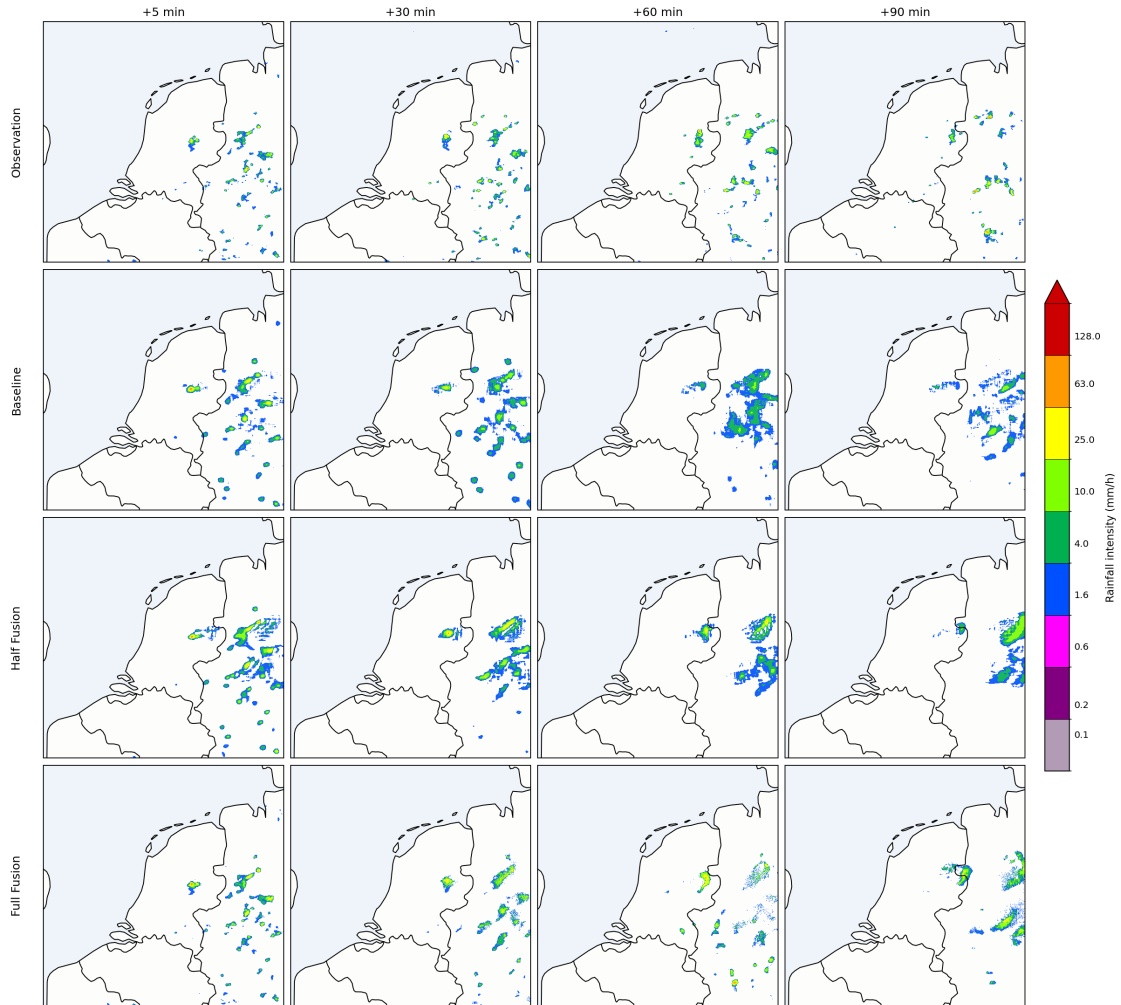


Figure 4.4: Qualitative comparison for event 4858 at 2025-09-05 13:50. Observed rainfall is sparse and mainly located over the eastern to central-eastern Netherlands and adjacent western Germany, while most of the Netherlands remains dry.

This event contains several locally intense rainfall cores, the observed precipitation field is spatially sparse. Most rainfall cells are small, isolated, and located over western Germany and near the eastern border of the Netherlands, while large parts of the Netherlands remain dry. The baseline produces fragmented and locally noisy rainfall patterns. Nevertheless, this behaviour partly resembles the observed event structure, in which the rainfall cells are naturally small, isolated, and spatially scattered. However, the baseline generally underestimates the intensity of the strongest cells, which is consistent with its lower CSI and POD at the 10 and 20 mm/h thresholds.

Half Fusion generates more spatially coherent precipitation structures and tends to merge originally separated rainfall cells into larger connected patches or elongated bands. In this event, this smoothing becomes a disadvantage because it overestimates the spatial continuity of the observed rainfall field. This behaviour is consistent with its relatively high FAR, particularly at the stronger rainfall thresholds.

Full Fusion more clearly preserves several isolated and localized rainfall cells and assigns higher intensities to some of the convective cores. This helps explain its higher CSI and POD at the 10 and 20 mm/h thresholds. However, several of the predicted intense cells do not fully coincide with the observed locations, resulting in the very high FAR at these thresholds.

Overall, event 4858 shows that Full Fusion improves the detection of localized moderate-to-heavy rainfall, but its stronger intensity response also increases the risk of high-intensity false alarms.

4.2.4. Overall Categorical Performance

After examining the models in individual rainfall events, their performance is evaluated across the full validation dataset to provide an overall assessment of their categorical behaviour.

At the 10 mm h⁻¹ threshold, Half Fusion provides the best overall balance between rainfall detection and false alarms. It achieves the highest CSI and POD among the three models, although its FAR remains higher than that of the baseline (See in table 4.6).

The baseline is more conservative, producing the lowest FAR but also the lowest CSI and POD. Full Fusion shows no clear advantage at this threshold, as it has lower CSI and POD than Half Fusion and the highest FAR.

Table 4.6: Generative categorical verification results at the 10 mm/h rainfall threshold.

Where (>) indicates higher values are better and (<) indicates lower values are better.			
	CSI (>)	POD (>)	FAR (<)
Model	≥ 10 mm/h	≥ 10 mm/h	≥ 10 mm/h
Baseline	0.064	0.091	0.176
Half Fusion	0.083	0.136	0.374
Full Fusion	0.077	0.128	0.556

These results indicate that the baseline reduces false alarms at the cost of missing more rainfall, whereas Half Fusion detects a larger proportion of rainfall but also produces more false alarms. Full Fusion does not improve this trade-off at the 10 mm h⁻¹ threshold.

The false alarms produced by both fusion models are concentrated just above the evaluation threshold. Approximately 86% of the false-alarm pixels from both Half Fusion and Full Fusion fall within the 10–20 mm h⁻¹ interval (Table 4.7), while false alarms above 40 mm h⁻¹ are rare. This indicates that most false alarms are associated with relatively weak predicted rainfall rather than intense rainfall.

Table 4.7: Distribution of false-alarm pixels across predicted rainfall-intensity intervals.

Predicted rainfall intensity (mm h ⁻¹)	Full Fusion (%)	Half Fusion (%)
10–20	86.59	85.62
20–40	12.54	13.51
40–80	0.81	0.84
80–128	0.05	0.03
≥ 128	0.01	0.00
Total	100	100

Increasing the degree of CTT fusion does not substantially change the intensity distribution of false alarms, but Full Fusion produces a lower proportion of correct detections among predicted heavy-rain pixels, resulting in a higher FAR

4.3. Clustering Results

The clustering analysis identified two distinct CTT–rainfall regimes introduced Section 2.5.3. To examine how the rainfall characteristics differ between these regimes, the event-level rainfall $Q_{0.90}$ values were grouped according to their assigned cluster and fitted with parametric probability distributions. These distributions also provide the statistical basis for the cluster-dependent ensemble post-processing method introduced in Section 3.3. The empirical distributions and their fitted probability density functions are presented in Figure 4.5.

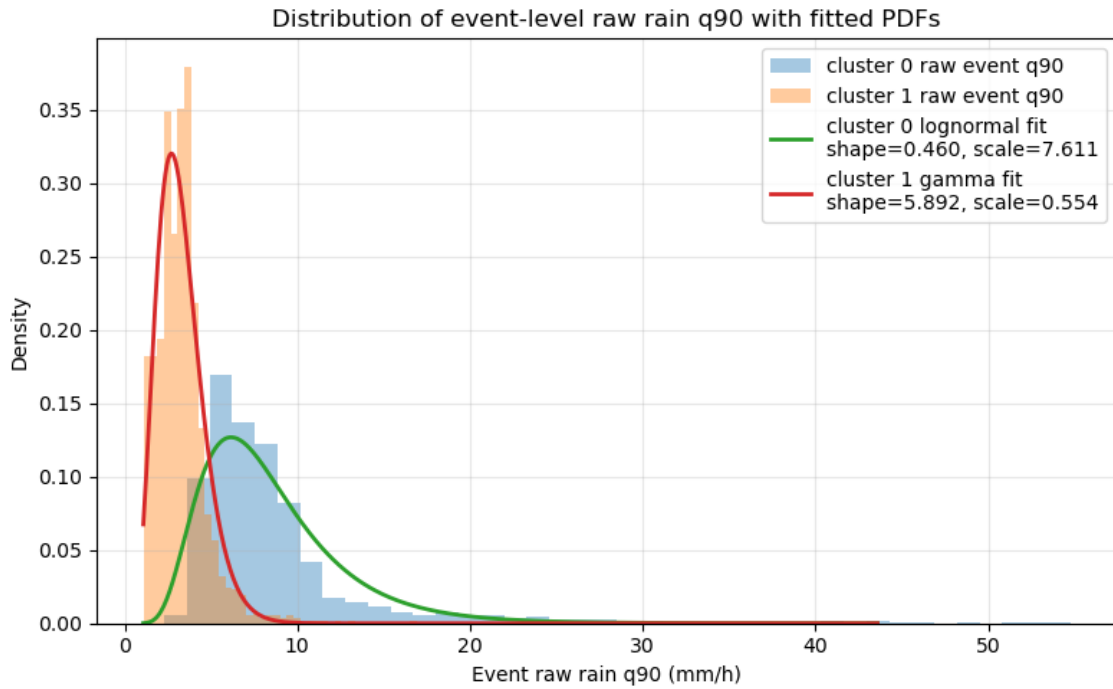


Figure 4.5: Fitted probability distributions of event-level raw rainfall q90 for the two identified CTT–rainfall clusters.

Figure 4.5 shows the fitted probability distributions of the event-level raw rainfall q90 for the two identified clusters. Cluster 0 is fitted with a lognormal distribution, with $\sigma = 0.460$ and scale = 7.611. This indicates that Cluster 0 has a higher rainfall intensity level and a longer right tail, corresponding to events with stronger and more variable precipitation. In contrast, Cluster 1 is fitted with a gamma distribution, with shape = 5.892 and scale = 0.554. This distribution is more concentrated at lower rainfall intensities, suggesting that Cluster 1 mainly represents weaker rainfall events with a narrower intensity range.

These fitted distributions provide a statistical interpretation of the two CTT-rainfall regimes. Based on this cluster-dependent rainfall uncertainty, the following experiment further tests whether the generative noise can be adjusted according to the assigned cluster. Different combinations of the affine parameters α and β are therefore evaluated to control the strength of the cluster-based scale and shift applied to the Gaussian noise. (as shown in Figure 4.6).

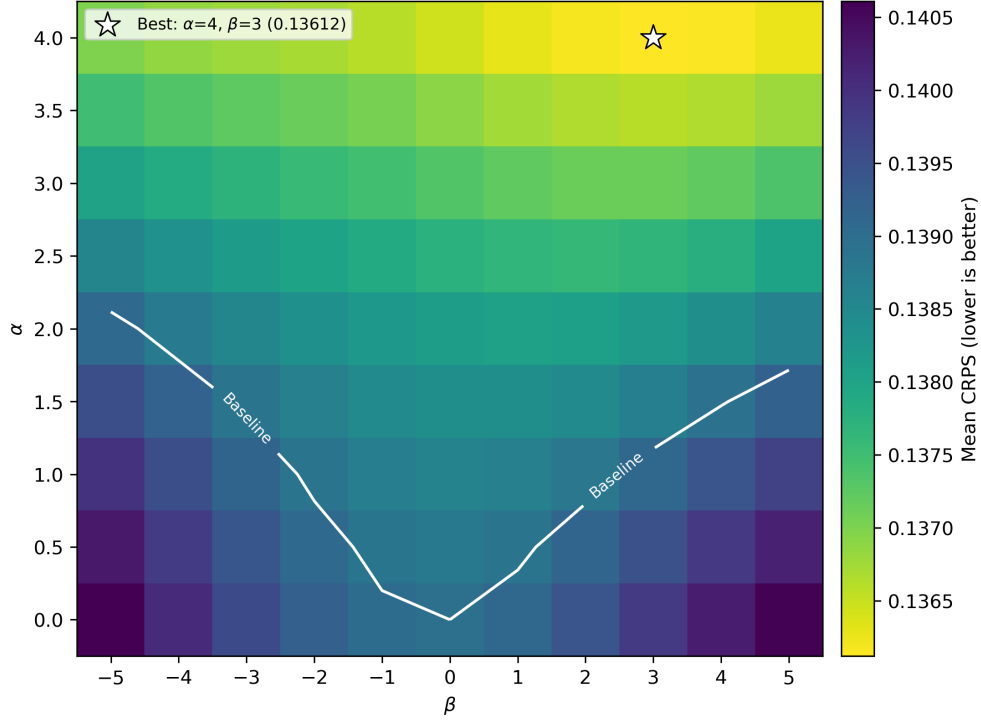


Figure 4.6: Sensitivity of the mean CRPS to the scale parameter α and shift parameter β in the CTT-regime-dependent Gaussian-noise adjustment. The white contour indicates parameter combinations with the same mean CRPS as the standard Gaussian baseline. The star marks the lowest mean CRPS within the tested parameter range.

Figure 4.6 shows the sensitivity of the mean CRPS to the scale parameter α and shift parameter β . The standard Gaussian baseline corresponds to $\alpha = 0$ and $\beta = 0$, while the white contour indicates parameter combinations with approximately the same mean CRPS as this baseline.

The contour has an approximately V-shaped pattern, indicating that larger positive or negative shifts generally require a stronger scale adjustment to maintain the baseline CRPS. Its slight asymmetry shows that shifts of equal magnitude but opposite signs do not produce exactly the same mean CRPS. Within the evaluated parameter range, the lowest mean CRPS of 0.136 was obtained at $\alpha = 4$ and $\beta = 3$. This combination was therefore selected as the optimal tested configuration for the subsequent ensemble evaluation.

The cluster-dependent scale and shift terms can be then computed from the fitted rainfall distributions by using mean and standard deviation at quantile 90 for each cluster aforementioned in Section 3.3 The resulting statistics and transformation parameters are summarized in Table 4.8.

Table 4.8: Cluster-dependent Gaussian noise parameters calculated using $\alpha = 4$ and $\beta = 3$.

Cluster	mean _c	std _c	scale(c)	shift(c)	Variance	Resulting distribution
Cluster 0	8.460	4.107	5.153	2.859	26.556	$\mathcal{N}(2.859, 26.556I)$
Cluster 1	3.264	1.345	0.059	-2.859	0.004	$\mathcal{N}(-2.859, 0.004I)$
mean _{ref} = 5.862		std _{ref} = 2.726		$\alpha = 4, \beta = 3$		

4.3.1. Ensemble results

Using the method described in Section 3.4.2, we compare the mean ΔCRPS between the post-processed ensemble and the original standard Gaussian ensemble in four regions: cold-rain, cold-no-rain, warm-rain, and warm-no-rain. This analysis examines whether the benefit of cluster-dependent noise transformation varies with local CTT and rainfall conditions, thereby addressing where and under which conditions CTT-informed post-processing improves probabilistic forecast skill (see in figure 4.7).

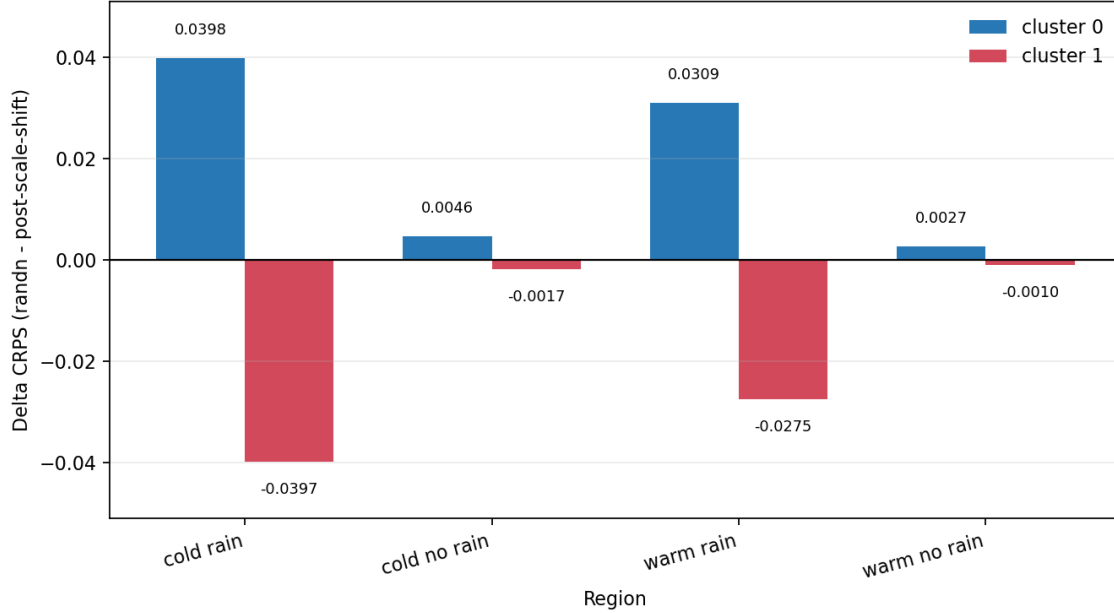


Figure 4.7: Regional change in CRPS after applying cluster-dependent post-scale-shift noise transformation. This test is performed on all validation events. Positive values indicate lower CRPS than the standard Gaussian ensemble.

The sign of the CRPS change is primarily determined by the event-level CTT-rainfall cluster. The transformation consistently improves the forecasts for cluster 0, but consistently degrades them for cluster 1. Moreover, the magnitude of the change is substantially larger in rainy regions than in no-rain regions. Also, within the rainy regions, the effect is slightly stronger for cold-rain pixels than for warm-rain pixels. These results indicate that the effectiveness of the noise transformation is governed mainly by the event-level precipitation regime.

For cluster 0, positive changes are obtained in all four regions. The largest improvements occur in the cold-rain and warm-rain regions, with ΔCRPS values of 0.0398 and 0.0309, respectively. The corresponding improvements in the cold-no-rain and warm-no-rain regions are much smaller, at 0.0046 and 0.0027. In contrast, cluster 1 produces negative changes in every region. The strongest degradations again occur in the rainy regions, reaching -0.0397 in the cold-rain region and -0.0275 in the warm-rain region, whereas the changes in the two no-rain regions remain close to zero.

In the no-rain regions, both the observed rainfall and a large proportion of the ensemble members are generally close to zero. Because the model output is bounded at zero and many ensemble members may already predict dry conditions, moderate changes in the latent noise do not necessarily produce substantial changes in the final rainfall field. The resulting CRPS differences therefore remain small. This explains why the distinction between rainy and non-rainy pixels has a much stronger influence on the magnitude of ΔCRPS than the distinction between cold and warm conditions.

Overall, the results show that the post-scale-shift transformation is not universally beneficial. The assigned event regime determines on whether the transformed latent-noise distribution is appropriate, therefore should be calibrated independently. For regimes in which the adjusted noise does not outperform

standard Gaussian sampling, retaining the standard Gaussian distribution or applying a validation-based selection mechanism would be more appropriate.

4.3.2. Event-Based Spatial Analysis of the Cluster-Conditioned Ensembles

To illustrate the event-dependent effect of the cluster-conditioned affine transformation, one representative event from each cluster is examined below.

Probabilistic Event Performance for Cluster 0

For cluster 0, observed on 21 June 2023 at 13:00 UTC (event 327), the post ensemble adjustment clearly improves probabilistic forecast performance.

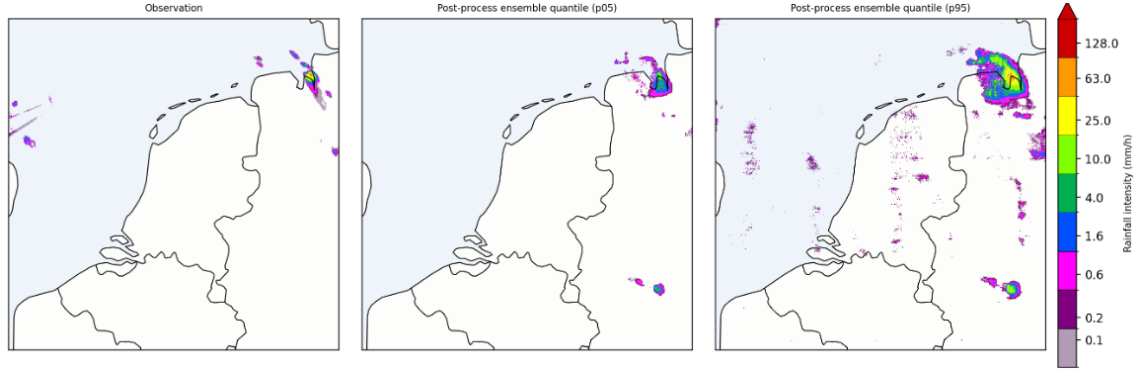


Figure 4.8: Lead-time-90 ensemble and intensity-dependent CRPS diagnostics for the selected cluster 0 and cluster 1 events. The upper row shows event 327 on 21 June 2023 at 13:00 UTC. The three spatial panels show the observation, 5th percentile (p05), and 95th percentile (p95) of the post-processed Gaussian (*randn*) full-fusion ensemble.

Compared between ensemble members at p05 and p95, the differences lie in the size and intensity of a small cell in northeastern Germany, whereas p95 shows a much larger and more intense precipitation area at the same location, together with additional weak cells over eastern and southeastern Germany. Such differences show the main spatial uncertainty in this event. Figure 4.9 summarises the corresponding lead-time-90 CRPS results.

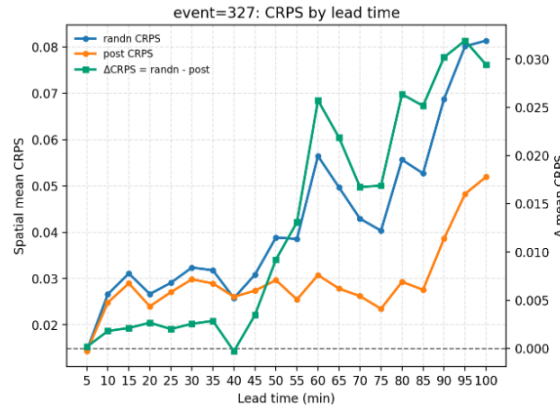


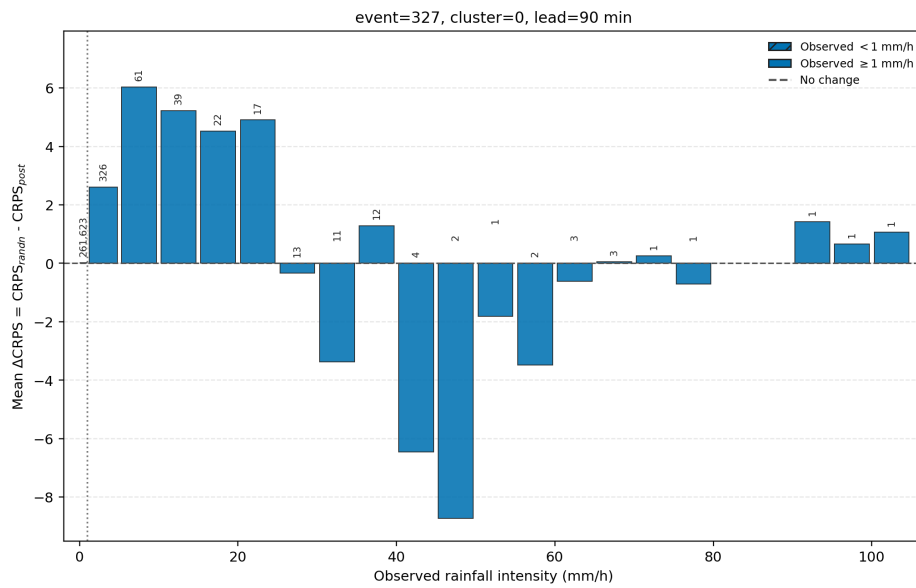
Figure 4.9: Spatially averaged CRPS across lead times for cluster0 event 327.

The adjusted ensemble yields a lower CRPS than the standard Gaussian ensemble at nearly all lead times, with the improvement generally increasing after approximately 50 min. At 90 min, the mean Δ CRPS reaches +0.030.

Table 4.9: Overall CRPS comparison for cluster 0 event 327 at 90 min. Positive $\Delta\text{CRPS} = \text{CRPS}_{\text{randn}} - \text{CRPS}_{\text{adjusted}}$ indicates improvement.

Metric	Value
Standard Gaussian ensemble	0.069
Adjusted ensemble	0.039
ΔCRPS	+0.030
Relative CRPS reduction	43.9%

Table 4.9 summarises the overall CRPS values and the corresponding changes across observed rainfall-intensity ranges. To further investigate which rainfall-intensity ranges contribute most to the CRPS reduction, the change in CRPS is examined using 5 mm/h intensity bins. The largest mean improvement occurs in the 5–10 mm h⁻¹ interval, with a mean ΔCRPS of +6.04. Although positive values are also found at higher rainfall intensities, no robust conclusion can be drawn above 20 mm h⁻¹ because of the limited number of pixels (see figure 4.10).



Numbers above the bars indicate the pixel count in each observed rainfall-intensity bin.

Observed intensity (mm h ⁻¹)	Mean ΔCRPS
< 1	+0.02
1–5	+2.61
5–10	+6.04
10–15	+5.23
15–20	+4.52
20–25	+4.91
> 25	Mixed

Note: Results above 20 mm h⁻¹ should be interpreted cautiously because of the limited number of pixels.

Figure 4.10: Intensity-dependent CRPS diagnostics for cluster 0 event 327 at a 90-min lead time. Positive values indicate improved performance after the cluster-dependent post-processing. The table below the plot summarises the mean ΔCRPS for the main observed rainfall-intensity ranges.

Probabilistic Event Performance for Cluster 1

The cluster 1, occurred on 24 May 2024 at 11:25 UTC (event 2865, see figure 4.11), is a clear case in which the same cluster-dependent post-processing framework degrades probabilistic forecast performance.

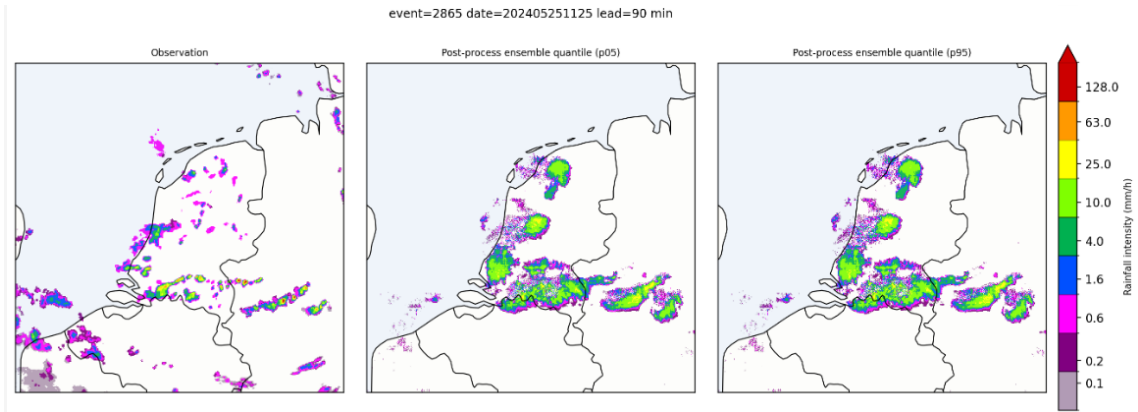


Figure 4.11: Lead-time-90 ensemble and intensity-dependent CRPS diagnostics for the selected cluster 0 and cluster 1 events. The upper row shows event 2865 on 24 May 2024 at 11:25 UTC. The three spatial panels show the observation, 5th percentile (p05), and 95th percentile (p95) of the post-processed Gaussian (*randn*) full-fusion ensemble.

The observation shows mainly scattered structure and the post-processed ensemble produces a narrower and more continuous precipitation band across the central and southern Netherlands and into western Germany. The post-processed ensemble has very limited spread, as the p05 and p95 panels are nearly identical. This strong underdispersion is consistent with the low-variance noise distribution assigned to cluster 1, which prevents the ensemble from representing uncertainty in the possible spatial extent and intensity of the precipitation. Figure 4.12 summarises the corresponding lead-time-90 CRPS results.

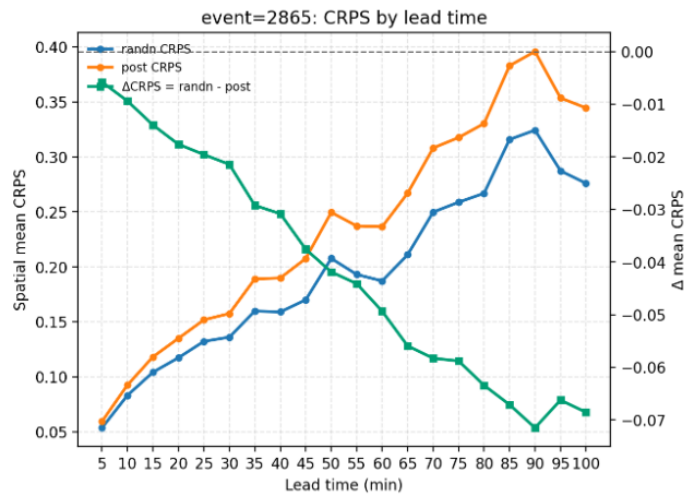


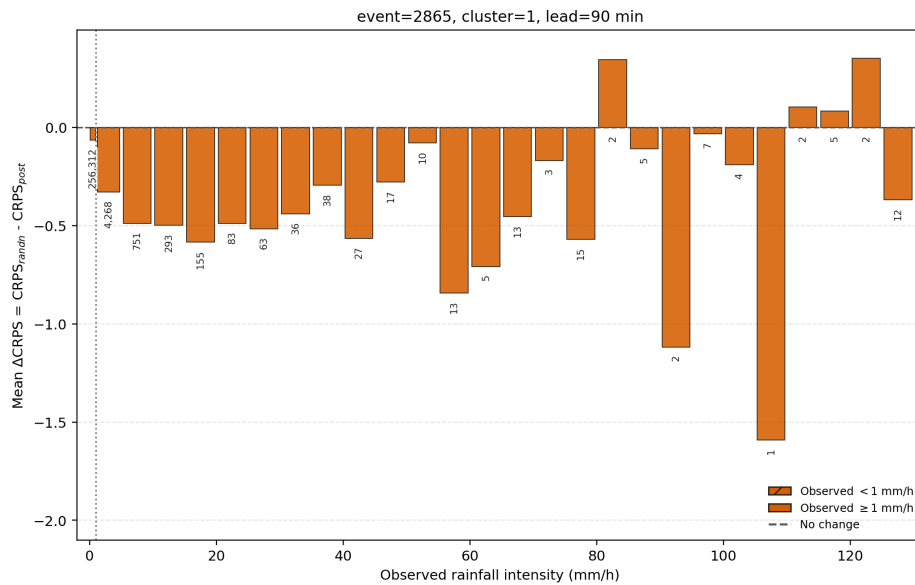
Figure 4.12: Spatially averaged CRPS across lead times for cluster 1 event 2865.

The post-processed ensemble yields a higher CRPS than the standard Gaussian ensemble at all lead times, with the deterioration generally increasing after approximately 45 min. At 90 min, the mean ΔCRPS reaches -0.072 . Table 4.10 summarises the corresponding overall CRPS values.

Table 4.10: Overall CRPS comparison for cluster 1 event 2865 at a lead time of 90 minutes. Positive $\Delta\text{CRPS} = \text{CRPS}_{\text{randn}} - \text{CRPS}_{\text{post}}$ indicates improvement.

Metric	Value
Standard Gaussian ensemble	0.324
Post-processed ensemble	0.396
ΔCRPS	-0.072
Relative deterioration	22.0%

Having shown that the post-processing degrades CRPS across all rainfall-intensity intervals at 90 min, the following figure examines how this deterioration develops with lead time. The post-processing degrades CRPS across all analysed rainfall-intensity ranges, indicating that the deterioration is not confined to a single intensity interval. Its magnitude generally increases with rainfall intensity and is largest in the 15–20 mm h^{−1} bin, where the mean Δ CRPS reaches -0.583 . Results above 20 mm h^{−1} remain uncertain because of the limited number of pixels (see Figure 4.13).



Numbers above the bars indicate the pixel count in each observed rainfall-intensity bin.

Observed intensity (mm h ^{−1})	Number of pixels	Mean Δ CRPS
< 1	256 312	−0.065
1–5	4 268	−0.329
5–10	751	−0.488
10–15	293	−0.498
15–20	155	−0.583
20–25	83	−0.489

Note: Negative values indicate that the post-processed ensemble performs worse than the standard Gaussian ensemble. Results above 20 mm h^{−1} should be interpreted cautiously because of the limited number of pixels.

Figure 4.13: Intensity-dependent CRPS diagnostics for cluster 1 event 2865 at a 90-min lead time. Positive values indicate improved performance after the cluster-dependent post-processing. The table below the plot summarises the mean Δ CRPS and pixel count for the main observed rainfall-intensity ranges.

5

Conclusion

This thesis investigated the implementation of NowcastNet for precipitation nowcasting over the Netherlands and examined whether cloud-top temperature (CTT) can provide additional information for deterministic rainfall prediction and ensemble uncertainty. The main findings are presented below by directly answering the research questions introduced in Chapter 1.5.

Sub-RQ1: How reliable and sharp are the baseline probabilistic forecasts of NowcastNet over the Netherlands when only radar information is used?

The radar-only NowcastNet baseline was able to reproduce the general movement and large-scale spatial development of precipitation systems over the Netherlands. However, its performance depended strongly on the rainfall structure and forecast lead time. Forecast skill generally decreased at longer lead times, while intense and spatially concentrated rainfall structures were more difficult to preserve than broader precipitation areas.

The baseline performed differently across individual events. It was relatively successful in representing some isolated rainfall cells, but it sometimes broadened, displaced, or merged organised precipitation structures. The ensemble forecasts also showed signs of underdispersion, indicating that the ensemble spread did not fully represent the observed forecast uncertainty. Therefore, although the radar-only baseline provided useful short-term forecasts, its probabilistic forecasts were not consistently reliable across all rainfall structures and lead times.

Sub-RQ2: How can CTT-rainfall regimes provide physically meaningful information for conditioning ensemble uncertainty?

The event-level relationship between CTT and rainfall intensity could be separated into two distinguishable regimes. The first regime was characterised by colder cloud tops and generally higher rainfall intensities, which may be associated with deeper or more strongly developed convective clouds. The second regime contained warmer cloud tops and predominantly lower rainfall intensities, representing weaker or less deeply developed precipitation systems.

These differences were visible in both the CTT-rainfall profiles and their statistical distributions. The resulting clusters therefore represented more than a purely numerical separation as they described physically different cloud and rainfall conditions. This demonstrates that CTT can provide meaningful information for distinguishing precipitation regimes and can consequently be used as a basis for regime-dependent conditioning of ensemble uncertainty.

Nevertheless, the two clusters should not be interpreted as strict categories of “heavy rainfall” and “light rainfall”. Considerable variation remained within each cluster, and the rainfall characteristics of an indi-

vidual event also depended on its spatial structure and temporal development.

Sub-RQ3: How does CTT-regime-dependent adjustment of Gaussian noise affect the probabilistic forecast skill and ensemble uncertainty of Nowcast-Net?

The regime-dependent affine adjustment changed both the location and spread of the latent Gaussian noise and consequently modified the generated ensemble distribution. Within the tested parameter combinations, the scale and shift setting with $\alpha = 4$ and $\beta = 3$ produced the lowest aggregate mean CRPS for the all validation events. These results show that, for the evaluated events, applying regime-dependent post-processing to the Gaussian latent noise resulted in lower CRPS than using the original standard Gaussian noise directly.

However, the effect of the adjustment was not consistent across events, forecast lead times, or observed rainfall intensities. For the examined cluster 0 event, the adjusted ensemble reduced the overall CRPS, but the improvement was mainly associated with the large number of weak-rainfall and near-dry pixels. At moderate and high observed rainfall intensities, the CRPS frequently increased. For the examined cluster 1 event, the adjustment did not provide an overall improvement and produced higher CRPS across most populated rainfall-intensity intervals.

The effectiveness of the proposed affine adjustment was closely linked to the event cluster and the specific precipitation structure. The current results do not support the conclusion that a single regime-dependent scale and shift transformation can consistently improve probabilistic forecast skill for all events.

Answer to the main research question

This thesis asked whether incorporating CTT information into NowcastNet can improve the forecasting of high-intensity rainfall and its associated uncertainty. To answer this question, several configurations were developed and evaluated: the radar-only baseline, merged-input models using concatenated and grouped-input radar–CTT inputs, a grouped-input model that introduced CTT into the evolution network, and a full-fusion model that also conditioned the generative network through multi-scale feature modulation.

Overall, incorporating CTT allowed NowcastNet to detect a larger proportion of the observed precipitation, as reflected by an increase in the probability of detection. However, the higher detection rate was generally accompanied by an increase in the false alarm ratio. The results therefore revealed a clear trade-off between detecting more observed rainfall and generating additional false rainfall predictions.

The additional rainfall produced by the CTT-conditioned models could not be consistently classified as either weak or intense rainfall. Its intensity and spatial characteristics varied between events, forecast lead times, and model configurations. The effect of CTT also depended on where it was introduced into the network. The simple merged-input approaches did not convert the additional CTT information into consistent forecast improvements, whereas the full-fusion configuration produced the clearest increase in rainfall detection but also introduced more false alarms.

CTT can therefore be considered a useful supplementary source of information for NowcastNet, particularly for increasing sensitivity to developing precipitation. However, it does not unconditionally improve the prediction of high-intensity rainfall or probabilistic forecast skill. Its benefits are event-dependent and are accompanied by a POD–FAR trade-off. Similarly, the ensemble experiments indicate that CTT-derived rainfall regimes can help characterise forecast uncertainty, but regime-dependent calibration requires more events and a more flexible adjustment method before it can provide consistent improvements.

6

Recommendation

Based on the findings and limitations of this study, several directions can be considered to improve both the performance and interpretability of the proposed framework. The following recommendations focus on the rainfall-regime classification, the size of the event dataset, the temporal representation in the evolution network, the influence of CTT conditioning, and the unexplained spatial artefacts in the full-fusion predictions.

This study used data from May to December during 2023 to 2025 and selected periods containing rainfall intensities above 10 mm h^{-1} . Approximately 100 rainfall events were identified, where one event here refers to the complete period from the beginning to the end of a high-intensity rainfall period rather than to an individual radar frame. Although these events include different rainfall structures, the dataset remains relatively small for training a complex nowcasting model and estimating stable rainfall-regime characteristics. Including additional years and a wider range of rainfall events would improve the stability and robustness of the trained model and reduce the influence of individual cases on the results.

CTT was introduced at the input level through merged and grouped - input. These two methods mainly differ in how the radar and CTT frames are arranged before entering the model. A possible extension is to treat time as an explicit third axis and process the radar and CTT sequences using three-dimensional convolutions. This would allow radar and CTT information from the same reference time to be processed together, while also learning their development across consecutive time steps.

In the full-fusion model, spatial feature maps extracted from CTT are used to modulate the rainfall features in the generator. The intention is to strengthen rainfall features where the cloud conditions support further development and weaken them elsewhere. The current results confirm that the CTT branch changes the predicted rainfall fields, but they do not show how strong this influence is or which spatial regions are affected most. A sensitivity analysis could scale the CTT features, remove individual conditioning blocks, or compare the CTT feature maps with the corresponding rainfall features. This would help determine whether CTT mainly affects rainfall intensity, spatial extent, or both.

Furthermore, the cluster-dependent post-processing assigns every event either to cluster 0, representing the more developed and higher-intensity rainfall regime, or to cluster 1, which is associated with weaker and more scattered precipitation. This hard assignment lacks the ability of expressing the uncertainty of the classification, even though some events may share characteristics of both clusters. A soft-assignment method could be used to represent the uncertainty in the event classification. In soft assignment, each event is given a probability of belonging to each cluster rather than being assigned a single fixed label. For example, a Gaussian mixture model could estimate these cluster-membership probabilities. This would be useful for events that share characteristics of both rainfall regimes and are difficult to assign to a single cluster. The soft assignment would help represent this classification uncertainty more clearly. The two

fitted distributions also overlap over a noticeable range, so ensemble draws from this region cannot be clearly interpreted as belonging to only one cluster. Using soft clusters as weights in a mixture distribution may help to better preserve this ambiguity rather than forcing every event or ensemble draw into a single regime.

Additionally, some full-fusion predictions were observed to contain isolated and grainy rainfall structures. These pixels roughly follow the outline of the precipitation region but do not form a continuous rainfall field. These patterns were initially suspected to be checkerboard artefacts. However, typical checkerboard artefacts show regular horizontal and vertical structures, whereas the pixels found here are more irregular and scattered. Their origin may therefore be related to another part of the model, such as the upsampling operations, the fusion blocks, the CTT-conditioning pathway, or the loss function. Inspecting intermediate feature maps and comparing different upsampling methods would help identify where these pixels first appear and which component is responsible for them.

This study demonstrated that integrating CTT information and rainfall-regime-dependent post-processing can improve specific aspects of probabilistic rainfall nowcasting in NowcastNet. By comparing different fusion strategies and analysing cluster-dependent uncertainty distributions, the study identified both the potential and the current limitations of using CTT to complement radar-based forecasts. This approach combines the direct observation of precipitation from radar with additional information on cloud development, while the regime-dependent post-processing provides a way to represent different rainfall conditions in the ensemble generation. The results suggest that this framework provides a useful basis for further research, although the temporal integration of CTT, cluster assignment, model interpretability, and prediction artefacts still require improvement. Future studies can build on these findings to develop more reliable and better-understood probabilistic nowcasting models, particularly for high-intensity rainfall events.

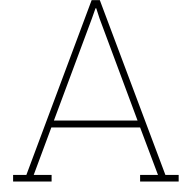
References

- [1] Charlotte Cambier van Nooten et al. “Improving Precipitation Nowcasting for High-Intensity Events Using Deep Generative Models with Balanced Loss and Temperature Data: A Case Study in the Netherlands”. EN. In: *Artificial Intelligence for the Earth Systems* 2.4 (Nov. 2023). issn: 2769-7525. doi: 10.1175/AIES-D-23-0017.1. URL: <https://journals.ametsoc.org/view/journals/aies/2/4/AIES-D-23-0017.1.xml>.
- [2] Seppo Pulkkinen et al. “Pysteps: an open-source Python library for probabilistic precipitation nowcasting (v1.0)”. English. In: *Geoscientific Model Development* 12.10 (Oct. 2019), pp. 4185–4219. issn: 1991-959X. doi: 10.5194/gmd-12-4185-2019. URL: <https://gmd.copernicus.org/articles/12/4185/2019/>.
- [3] Royal Netherlands Meteorological Institute. *Local precipitation*. URL: <https://www.knmiprojects.nl/projects/future-weather/local-precipitation>.
- [4] PBL Netherlands Environmental Assessment Agency. *Low probabilities, large consequences*. URL: <https://themasites.pbl.nl/o/flood-risks/>.
- [5] Juanzhen Sun et al. “Use of NWP for Nowcasting Convective Precipitation: Recent Progress and Challenges”. EN. In: *Bulletin of the American Meteorological Society* 95.3 (Mar. 2014), pp. 409–426. issn: 0003-0007, 1520-0477. doi: 10.1175/BAMS-D-11-00263.1. URL: <https://journals.ametsoc.org/view/journals/bams/95/3/bams-d-11-00263.1.xml>.
- [6] *The quiet revolution of numerical weather prediction | Nature*. URL: <https://www.nature.com/articles/nature14956>.
- [7] Urs Germann et al. “Radar precipitation measurement in a mountainous region”. en. In: *Quarterly Journal of the Royal Meteorological Society* 132.618 (2006), pp. 1669–1692. issn: 1477-870X. doi: 10.1256/qj.05.190. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1256/qj.05.190>.
- [8] Zbyněk Sokol et al. “Probabilistic precipitation nowcasting based on an extrapolation of radar reflectivity and an ensemble approach”. In: *Atmospheric Research* 194 (Sept. 2017), pp. 245–257. issn: 0169-8095. doi: 10.1016/j.atmosres.2017.05.003. URL: <https://www.sciencedirect.com/science/article/pii/S0169809516305142>.
- [9] Sepp Hochreiter and Jürgen Schmidhuber. “Long Short-Term Memory”. In: *Neural Computation* 9.8 (Nov. 1997), pp. 1735–1780. issn: 0899-7667. doi: 10.1162/neco.1997.9.8.1735. URL: <https://ieeexplore.ieee.org/abstract/document/6795963>.
- [10] Xingjian SHI et al. “Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting”. In: *Advances in Neural Information Processing Systems*. Vol. 28. Curran Associates, Inc., 2015. URL: <https://proceedings.neurips.cc/paper/2015/hash/07563a3fe3bbe7e3ba84431ad9d055af-Abstract.html>.
- [11] Yuchen Zhang et al. “Skilful nowcasting of extreme precipitation with NowcastNet”. en. In: *Nature* 619.7970 (July 2023). tex.copyright: 2023 The Author(s), pp. 526–532. issn: 1476-4687. doi: 10.1038/s41586-023-06184-4. URL: <https://www.nature.com/articles/s41586-023-06184-4>.
- [12] Pengjie Cai, He Huang, and Taoli Liu. “Improving nowcasting of intense convective precipitation by incorporating dual-polarization radar variables into generative adversarial networks”. In: *Sensors* 24.15 (2024), p. 4895. URL: <https://www.mdpi.com/1424-8220/24/15/4895>.
- [13] Jussi Leinonen et al. “Thunderstorm nowcasting with deep learning: a multi-hazard data fusion model”. In: *Geophysical Research Letters* 50.8 (Apr. 2023), e2022GL101626. issn: 0094-8276, 1944-8007. doi: 10.1029/2022GL101626. URL: <http://arxiv.org/abs/2211.01001>.

- [14] Jenna Ritvanen et al. "Cell-tracking-based framework for assessing nowcasting model skill in reproducing growth and decay of convective rainfall". In: *Geoscientific Model Development* 18.5 (2025), pp. 1851–1878. URL: <https://gmd.copernicus.org/articles/18/1851/2025/gmd-18-1851-2025.html>.
- [15] R. O. Imhoff et al. "Large-Sample Evaluation of Radar Rainfall Nowcasting for Flood Early Warning". en. In: *Water Resources Research* 58.3 (2022). tex.copyright: © 2022. The Authors., e2021WR031591. ISSN: 1944-7973. DOI: 10.1029/2021WR031591. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1029/2021WR031591>.
- [16] Mohammad Reza Ehsani et al. "Nowcasting-Nets: Deep Neural Network Structures for Precipitation Nowcasting Using IMERG". In: *IEEE Transactions on Geoscience and Remote Sensing* 60 (2022). arXiv:2108.06868 [cs.LG], pp. 1–21. ISSN: 0196-2892, 1558-0644. DOI: 10.1109/TGRS.2022.3158888. URL: <http://arxiv.org/abs/2108.06868>.
- [17] Xingjian Shi et al. "Deep Learning for Precipitation Nowcasting: A Benchmark and A New Model". In: *Advances in Neural Information Processing Systems*. Vol. 30. Curran Associates, Inc., 2017. URL: <https://proceedings.neurips.cc/paper/2017/hash/a6db4ed04f1621a119799fd3d7545d3d-Abstract.html>.
- [18] XuDong Ling et al. "Spacetime Separable Latent Diffusion Model With Intensity Structure Information for Precipitation Nowcasting". In: *IEEE Transactions on Geoscience and Remote Sensing* 62 (2024), pp. 1–15. ISSN: 1558-0644. DOI: 10.1109/TGRS.2024.3444789. URL: <https://ieeexplore.ieee.org/abstract/document/10638096>.
- [19] Zhifeng Ma, Hao Zhang, and Jie Liu. "MM-RNN: A Multimodal RNN for Precipitation Nowcasting". In: *IEEE Transactions on Geoscience and Remote Sensing* 61 (2023), pp. 1–14. ISSN: 1558-0644. DOI: 10.1109/TGRS.2023.3264545. URL: <https://ieeexplore.ieee.org/abstract/document/10092888>.
- [20] Casper Kaae Sønderby et al. *MetNet: A Neural Weather Model for Precipitation Forecasting*. ADS Bibcode: 2020arXiv200312140K. Mar. 2020. DOI: 10.48550/arXiv.2003.12140. URL: <https://ui.adsabs.harvard.edu/abs/2020arXiv200312140K>.
- [21] Wonsu Kim, Chang-Hoo Jeong, and Seongchan Kim. "Improvements in deep learning-based precipitation nowcasting using major atmospheric factors with radar rain rate". In: *Computers & Geosciences* 184 (Feb. 2024), p. 105529. ISSN: 0098-3004. DOI: 10.1016/j.cageo.2024.105529. URL: <https://www.sciencedirect.com/science/article/pii/S0098300424000128>.
- [22] John R. Mecikalski and Kristopher M. Bedka. "Forecasting Convective Initiation by Monitoring the Evolution of Moving Cumulus in Daytime GOES Imagery". EN. In: *Monthly Weather Review* 134.1 (Jan. 2006), pp. 49–78. ISSN: 1520-0493, 0027-0644. DOI: 10.1175/MWR3062.1. URL: <https://journals.ametsoc.org/view/journals/mwre/134/1/mwr3062.1.xml>.
- [23] Sha Lu et al. "Precipitation Regime Classification Based on Cloud-Top Temperature Time Series for Spatially-Variied Parameterization of Precipitation Models". en. In: *Remote Sensing* 12.2 (Jan. 2020), p. 289. ISSN: 2072-4292. DOI: 10.3390/rs12020289. URL: <https://www.mdpi.com/2072-4292/12/2/289>.
- [24] Bent Harnist, Seppo Pulkkinen, and Terhi Mäkinen. "DEUCE v1.0: a neural network for probabilistic precipitation nowcasting with aleatoric and epistemic uncertainties". English. In: *Geoscientific Model Development* 17.9 (May 2024), pp. 3839–3866. ISSN: 1991-959X. DOI: 10.5194/gmd-17-3839-2024. URL: <https://gmd.copernicus.org/articles/17/3839/2024/>.
- [25] Velpuri Manikanta et al. "On the verification of ensemble precipitation forecasts over the Godavari River basin". In: *Journal of Hydrology* 616 (Jan. 2023), p. 128794. ISSN: 0022-1694. DOI: 10.1016/j.jhydrol.2022.128794. URL: <https://www.sciencedirect.com/science/article/pii/S0022169422013646>.

- [26] Hung-Ming Lin and Li-Pen Wang. “Enhancing Radar-Based Ensemble Nowcasting with CSGD: A UK Study”. In: Apr. 2024, p. 2799. doi: 10.5194/egusphere-egu24-2799. URL: <https://ui.adsabs.harvard.edu/abs/2024EGUGA...26.2799L>.
- [27] Congyi Nai et al. *Reliable precipitation nowcasting using probabilistic diffusion model*. en. Nov. 2023. doi: 10.22541/essoar.169945499.97460779/v1. URL: <https://essopenarchive.org/users/695859/articles/684370-reliable-precipitation-nowcasting-using-probabilistic-diffusion-model?commit=f94739ae91a5ed2b24b51fcc93b4c10064e1e2f6>.
- [28] Yingkai Sha, Ryan A. Sobash, and David John Gagne. “Improving ensemble extreme precipitation forecasts using generative artificial intelligence”. In: *Artificial Intelligence for the Earth Systems* 4.2 (2025), e240063. URL: <https://journals.ametsoc.org/view/journals/aies/4/2/AIES-D-24-0063.1.xml>.
- [29] *Skilful precipitation nowcasting using deep generative models of radar* | Nature. URL: <https://www.nature.com/articles/s41586-021-03854-z?fromPaywallRec=false>.
- [30] Josh Jacobson et al. *Beyond Univariate Calibration: Verifying Spatial Structure in Ensembles of Forecast Fields*. Jan. 2020. doi: 10.5194/npg-2019-63.
- [31] Stefan Siegert et al. *Spatial forecast postprocessing: The Max-and-Smooth approach*. Sept. 2022. doi: 10.48550/arXiv.2209.00477. URL: <http://arxiv.org/abs/2209.00477>.
- [32] Sojung An et al. “Self-clustered GAN for precipitation nowcasting”. en. In: *Scientific Reports* 14.1 (Apr. 2024), p. 9755. ISSN: 2045-2322. doi: 10.1038/s41598-024-60253-w. URL: <https://www.nature.com/articles/s41598-024-60253-w>.
- [33] Royal Netherlands Meteorological Institute (KNMI). *Radar-derived 5 minute precipitation accumulations, corrected by rain gauge data*. tex.howpublished: KNMI Data Platform. 2024. URL: <https://datapatform.knmi.nl/dataset/access/nl-rdr-data-rtcor-5m-1-0>.
- [34] Aart Overeem et al. “The Dutch real-time gauge-adjusted radar precipitation product”. English. In: *Earth System Science Data* 17.9 (Sept. 2025), pp. 4715–4736. ISSN: 1866-3508. doi: 10.5194/essd-17-4715-2025. URL: <https://essd.copernicus.org/articles/17/4715/2025/>.
- [35] Royal Netherlands Meteorological Institute (KNMI). *Clouds, precipitation and radiation - MSG-CPP products*. tex.howpublished: KNMI Data Platform. 2024. URL: <https://datapatform.knmi.nl/dataset/msg-cpp-products-1-0>.
- [36] *Estimating surface solar irradiance from METEOSAT SEVIRI-derived cloud properties*. nl. URL: <https://www.knmi.nl/research/publications/estimating-surface-solar-irradiance-from-meteosat-seviri-derived-cloud-properties>.
- [37] *rioxarray package — rioxarray 0.22.0 documentation*. URL: <https://corteva.github.io/rioxarray/html/rioxarray.html>.
- [38] Said Ouala et al. “Neural Network Based Kalman Filters for the Spatio-Temporal Interpolation of Satellite-Derived Sea Surface Temperature”. en. In: *Remote Sensing* 10.12 (Dec. 2018). Publisher: Multidisciplinary Digital Publishing Institute, p. 1864. ISSN: 2072-4292. doi: 10.3390/rs10121864. URL: <https://www.mdpi.com/2072-4292/10/12/1864>.
- [39] Thomas Vandal and Ramakrishna Nemani. *Temporal Interpolation of Geostationary Satellite Imagery with Task Specific Optical Flow*. arXiv:1907.12013 [cs.CV]. Mar. 2020. doi: 10.48550/arXiv.1907.12013. URL: <http://arxiv.org/abs/1907.12013>.
- [40] Liquan Zhou et al. “Fusion of Dual Wind Component for Radar Echo Nowcasting Based on a Deep Learning Model”. en. In: *Earth and Space Science* 12.9 (2025), e2024EA004128. ISSN: 2333-5084. doi: 10.1029/2024EA004128. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1029/2024EA004128>.
- [41] Guilin Liu et al. “Image Inpainting for Irregular Holes Using Partial Convolutions: 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XI”. In: Sept. 2018, pp. 89–105. ISBN: 978-3-030-01251-9. doi: 10.1007/978-3-030-01252-6_6.

- [42] Royal Netherlands Meteorological Institute (KNMI). *Daily in-situ meteorological observations, validated*. 2025. URL: <https://dataplatform.knmi.nl/dataset/daily-in-situ-meteorological-observations-validated-1->.
- [43] Seppo Pulkkinen, V. Chandrasekar, and Ari-Matti Harri. “Fully Spectral Method for Radar-Based Precipitation Nowcasting”. In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 12.5 (May 2019), pp. 1369–1382. ISSN: 2151-1535. DOI: 10.1109/JSTARS.2019.2908974. URL: <https://ieeexplore.ieee.org/document/8692385>.
- [44] Francisco Mena et al. “Common Practices and Taxonomy in Deep Multi-view Fusion for Remote Sensing Applications”. In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 17 (2024). arXiv:2301.01200 [cs.CV], pp. 4797–4818. ISSN: 1939-1404, 2151-1535. DOI: 10.1109/JSTARS.2024.3361556. URL: <http://arxiv.org/abs/2301.01200>.
- [45] Nikos Benas et al. “The MSG-SEVIRI-based cloud property data record CLAAS-2”. English. In: *Earth System Science Data* 9.2 (July 2017), pp. 415–434. ISSN: 1866-3508. DOI: 10.5194/essd-9-415-2017. URL: <https://essd.copernicus.org/articles/9/415/2017/>.
- [46] Konrad Gadzicki, Razieh Khamsehashari, and Christoph Zetzsche. “Early vs Late Fusion in Multi-modal Convolutional Neural Networks”. In: IEEE, 2020, pp. 1–6. DOI: 10.23919/FUSION45008.2020.9190246. URL: https://www.cognitive-neuroinformatics.com/publications/bibtexbrowser.php?bib=publications%2Fcni_all_publications.bib&key=Gadzicki2020Fusion&utm_source=chatgpt.com.
- [47] Christos Kaparakis and Siamak Mehrkanoon. *WF-UNet: Weather Fusion UNet for Precipitation Nowcasting*. Feb. 2023. DOI: 10.48550/arXiv.2302.04102. URL: <http://arxiv.org/abs/2302.04102>.
- [48] Dan Niu et al. “FsrGAN: A Satellite and Radar-Based Fusion Prediction Network for Precipitation Nowcasting”. In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 17 (2024), pp. 7002–7013. ISSN: 2151-1535. DOI: 10.1109/JSTARS.2024.3376987. URL: <https://ieeexplore.ieee.org/document/10472033>.
- [49] Alexandros Stergiou, Ronald Poppe, and Grigorios Kalliatakis. “Refining activation downsampling with SoftPool”. In: *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. Montreal, QC, Canada: IEEE, Oct. 2021, pp. 10337–10346. ISBN: 978-1-6654-2812-5. DOI: 10.1109/ICCV48922.2021.01019. URL: <https://ieeexplore.ieee.org/document/9711197/>.
- [50] Ziteng Gao, Limin Wang, and Gangshan Wu. “LIP: Local Importance-Based Pooling”. en. In: ().
- [51] Sanghyun Woo et al. “CBAM: Convolutional Block Attention Module”. In: *Computer Vision – ECCV 2018: 15th European Conference, Munich, Germany, September 8–14, 2018, Proceedings, Part VII*. Berlin, Heidelberg: Springer-Verlag, Sept. 2018, pp. 3–19. ISBN: 978-3-030-01233-5. DOI: 10.1007/978-3-030-01234-2_1. URL: https://doi.org/10.1007/978-3-030-01234-2_1.
- [52] Delft High Performance Computing Centre (DHPC). *DelftBlue Supercomputer (Phase 2)*. tex.ark: ark:/44463/DelftBluePhase2. 2024. URL: <https://www.tudelft.nl/dhpc/ark:/44463/DelftBluePhase2>.
- [53] Yuchen Zhang et al. “Skilful nowcasting of extreme precipitation with NowcastNet”. en. In: *Nature* 619.7970 (July 2023). tex.copyright: 2023 The Author(s), pp. 526–532. ISSN: 1476-4687. DOI: 10.1038/s41586-023-06184-4. URL: <https://www.nature.com/articles/s41586-023-06184-4>.



Events summary

Table A.1: Dates, time ranges, and mean rainfall intensities of the selected rainfall events. The original `mean_RHX` values were multiplied by 12 and are reported in mm h^{-1} .

Date	Time range	Mean rainfall intensity (mm h^{-1})
2023-05-07	13:00–16:00	127.8
2023-05-08	13:00–16:00	175.6
2023-05-12	05:00–18:00	162.6
2023-05-23	02:00–02:00	122.4
2023-06-21	14:00–20:00	168.8
2023-06-23	16:00–18:00	183.6
2023-07-05	11:00–24:00	129.6
2023-07-10	15:00–18:00	209.1
2023-07-12	22:00–22:00	194.4
2023-07-14	08:00–08:00	138.0
2023-07-18	14:00–16:00	124.2
2023-07-25	14:00–14:00	198.0
2023-07-30	13:00–13:00	160.8
2023-07-31	19:00–22:00	154.8
2023-08-03	13:00–22:00	144.0
2023-08-04	05:00–05:00	177.6
2023-08-06	22:00–22:00	121.2
2023-08-07	02:00–16:00	166.6
2023-08-09	21:00–21:00	132.0
2023-08-13	05:00–09:00	151.4
2023-08-26	08:00–16:00	222.0
2023-08-27	15:00–15:00	150.0
2023-08-28	13:00–18:00	152.4
2023-08-31	08:00–08:00	132.0
2023-09-02	24:00–24:00	151.2
2023-09-12	20:00–20:00	232.8
2023-09-13	18:00–19:00	373.8
2023-09-14	03:00–05:00	209.6
2023-09-17	18:00–18:00	195.6
2023-09-19	07:00–22:00	147.3
2023-09-22	19:00–19:00	187.2

Continued on next page

Table A.1 – continued from previous page

Date	Time range	Mean rainfall intensity (mm h ⁻¹)
2023-10-12	20:00–20:00	139.2
2023-10-13	13:00–14:00	197.1
2023-10-14	23:00–24:00	153.6
2023-10-19	21:00–22:00	134.4
2023-11-01	21:00–21:00	121.2
2024-04-18	11:00–11:00	123.6
2024-05-02	01:00–01:00	241.2
2024-05-03	18:00–21:00	210.0
2024-05-14	15:00–15:00	177.6
2024-05-17	08:00–23:00	135.0
2024-05-19	19:00–19:00	183.6
2024-05-20	14:00–17:00	167.4
2024-05-21	14:00–17:00	196.8
2024-05-22	17:00–17:00	198.0
2024-05-23	03:00–03:00	277.2
2024-05-25	11:00–15:00	149.4
2024-05-26	03:00–13:00	144.3
2024-05-28	21:00–21:00	169.2
2024-05-30	14:00–14:00	132.0
2024-06-11	09:00–09:00	123.6
2024-06-15	20:00–20:00	150.0
2024-06-22	11:00–11:00	135.6
2024-07-04	08:00–08:00	188.4
2024-07-10	17:00–21:00	159.1
2024-07-11	14:00–14:00	123.6
2024-07-13	09:00–16:00	203.1
2024-07-17	15:00–15:00	150.0
2024-07-22	10:00–17:00	216.0
2024-07-24	12:00–14:00	184.8
2024-07-28	21:00–21:00	213.6
2024-08-14	18:00–21:00	273.6
2024-08-15	14:00–17:00	195.6
2024-08-18	03:00–03:00	132.0
2024-08-21	19:00–21:00	214.8
2024-08-25	17:00–18:00	187.2
2024-09-03	16:00–21:00	257.1
2024-09-07	20:00–20:00	134.4
2024-09-11	19:00–21:00	162.9
2024-09-12	17:00–17:00	168.0
2024-09-28	10:00–12:00	155.4
2024-10-09	01:00–01:00	123.6
2024-10-13	21:00–21:00	129.6
2024-12-06	22:00–22:00	140.4
2024-12-20	03:00–03:00	139.2
2025-03-24	21:00–21:00	344.4
2025-04-24	18:00–18:00	153.6
2025-06-01	20:00–20:00	148.8
2025-06-08	08:00–08:00	164.4
2025-06-09	04:00–04:00	129.6
2025-06-15	22:00–22:00	147.6
2025-06-26	24:00–24:00	175.2

Continued on next page

Table A.1 – continued from previous page

Date	Time range	Mean rainfall intensity (mm h ⁻¹)
2025-06-27	01:00–11:00	167.2
2025-07-03	16:00–16:00	184.8
2025-07-07	12:00–23:00	179.3
2025-07-16	10:00–10:00	219.6
2025-07-17	11:00–15:00	150.6
2025-07-21	19:00–19:00	182.4
2025-07-22	16:00–20:00	146.7
2025-07-24	14:00–17:00	201.6
2025-07-27	24:00–24:00	128.4
2025-08-03	06:00–10:00	152.4
2025-08-04	23:00–23:00	126.0
2025-08-29	19:00–19:00	132.0
2025-08-31	05:00–05:00	235.2
2025-09-04	16:00–20:00	190.8
2025-09-05	16:00–16:00	128.4
2025-09-10	12:00–12:00	128.4
2025-09-17	06:00–06:00	122.4
2025-10-21	18:00–18:00	122.4

B

NowcastNet full structure

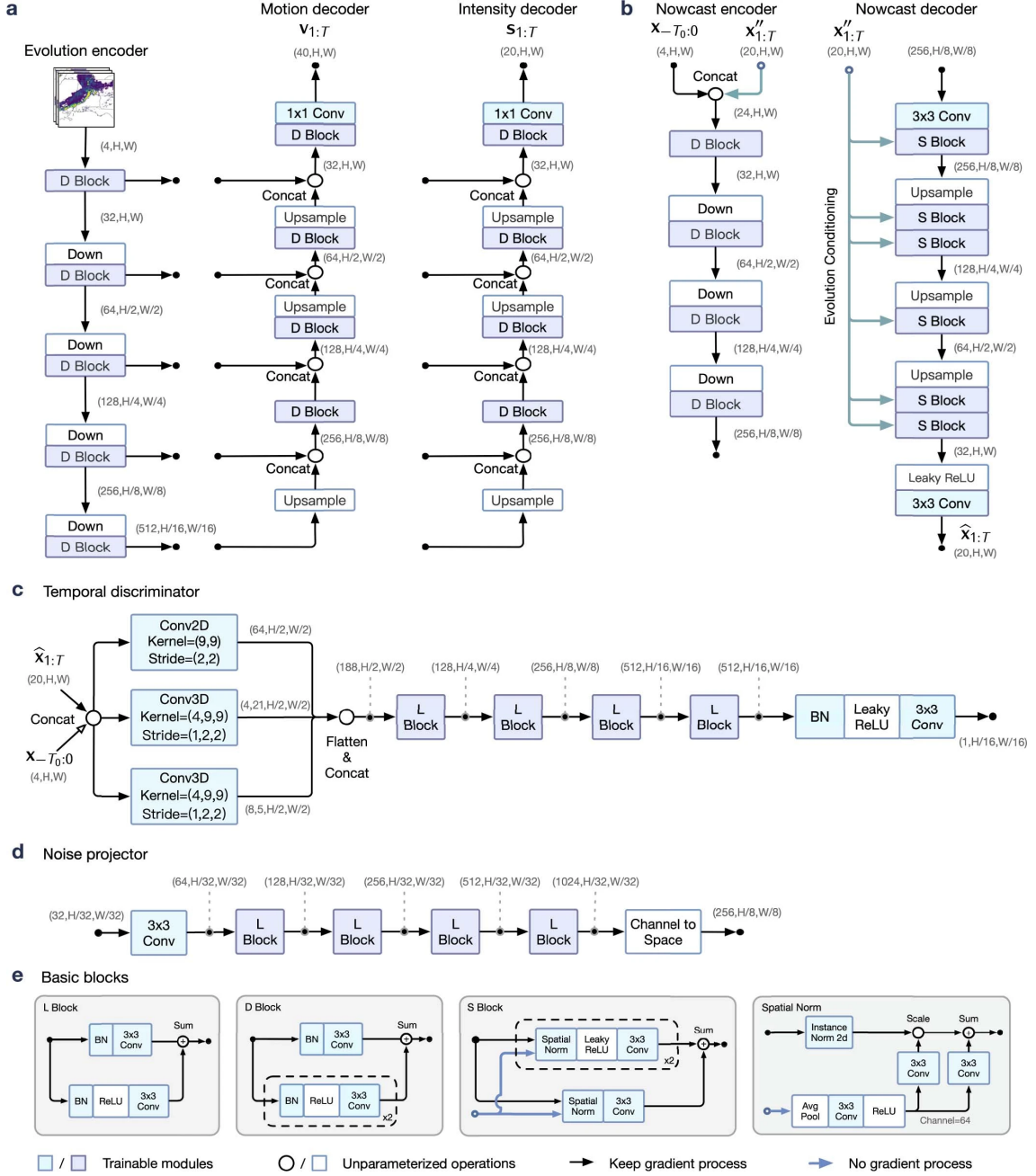
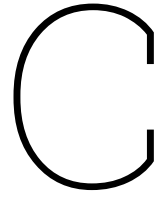


Figure B.1: Architecture of NowcastNet, a physics-conditional deep generative model. Reproduced from Zhang et al. [zhang2023nowcastnet], Fig. 1a, under the CC BY 4.0 licence.



Additional events

The below two cases showed that observed rainfall consists mainly of small, scattered precipitation areas, particularly at 90 minutes, whereas the baseline and half-fusion models produce broader and more intense rainfall regions. This suggests that both models have difficulty representing spatially fragmented events and tend to merge scattered cells into unrealistically extensive precipitation systems.

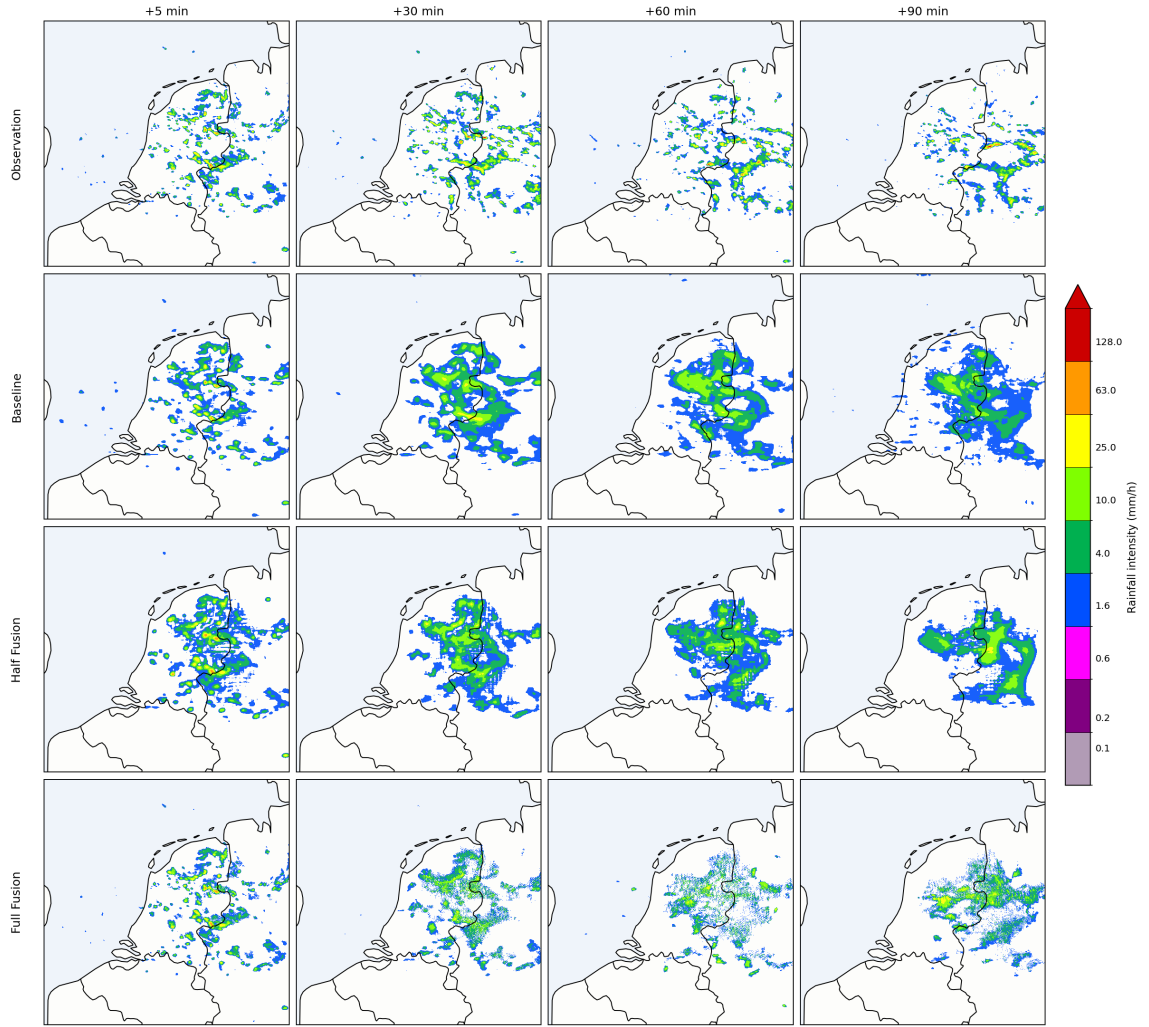


Figure C.1: Observed and predicted rainfall fields for the event initialized at 17:35 UTC on 21 July 2025. The rows show the observation, baseline, half-fusion, and full-fusion results, and the columns represent increasing forecast lead times. Rainfall intensity is given in mm h^{-1} .

Table C.1: Categorical metrics for event 4351 over the 5–90 min lead-time window at a rainfall threshold of 10 mm h^{-1} .

Event	Model	CSI ($\uparrow$)	POD ($\uparrow$)	FAR ($\downarrow$)
4351	Baseline	0.133	0.159	0.647
	Half fusion	0.147	0.181	0.613
	Full fusion	0.170	0.233	0.666

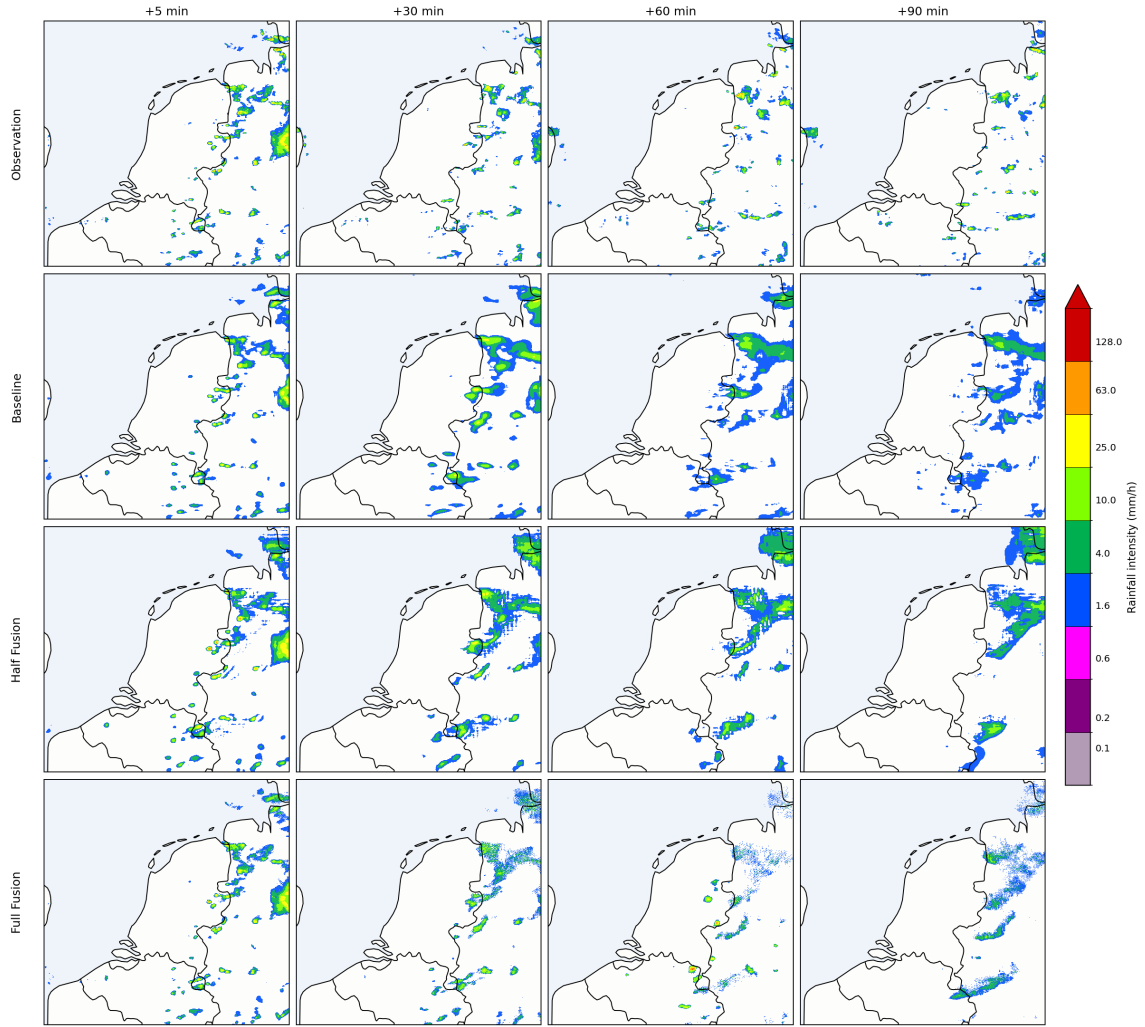


Figure C.2: Observed and predicted rainfall fields for the event initialized at 14:10 UTC on 22 July 2025. The rows show the observation, baseline, half-fusion, and full-fusion results, and the columns represent increasing forecast lead times. Rainfall intensity is given in mm h^{-1} .

Table C.2: Categorical metrics for event 4442 over the 5–90 min lead-time window at a rainfall threshold of 10 mm h^{-1} .

Event	Model	CSI ($\uparrow$)	POD ($\uparrow$)	FAR ($\downarrow$)
4442	Baseline	0.169	0.230	0.590
	Half fusion	0.175	0.237	0.660
	Full fusion	0.136	0.178	0.773

This below example shows that the baseline model produces a pronounced chessboard artifact, especially at longer lead times, and this artifact is not observed in the other models. In contrast, the half-fusion model captures the rainfall structure more realistically, but, consistent with the previous cases, it still generates extensive areas of low-intensity precipitation, with many pixels around 1.6 mm h^{-1}

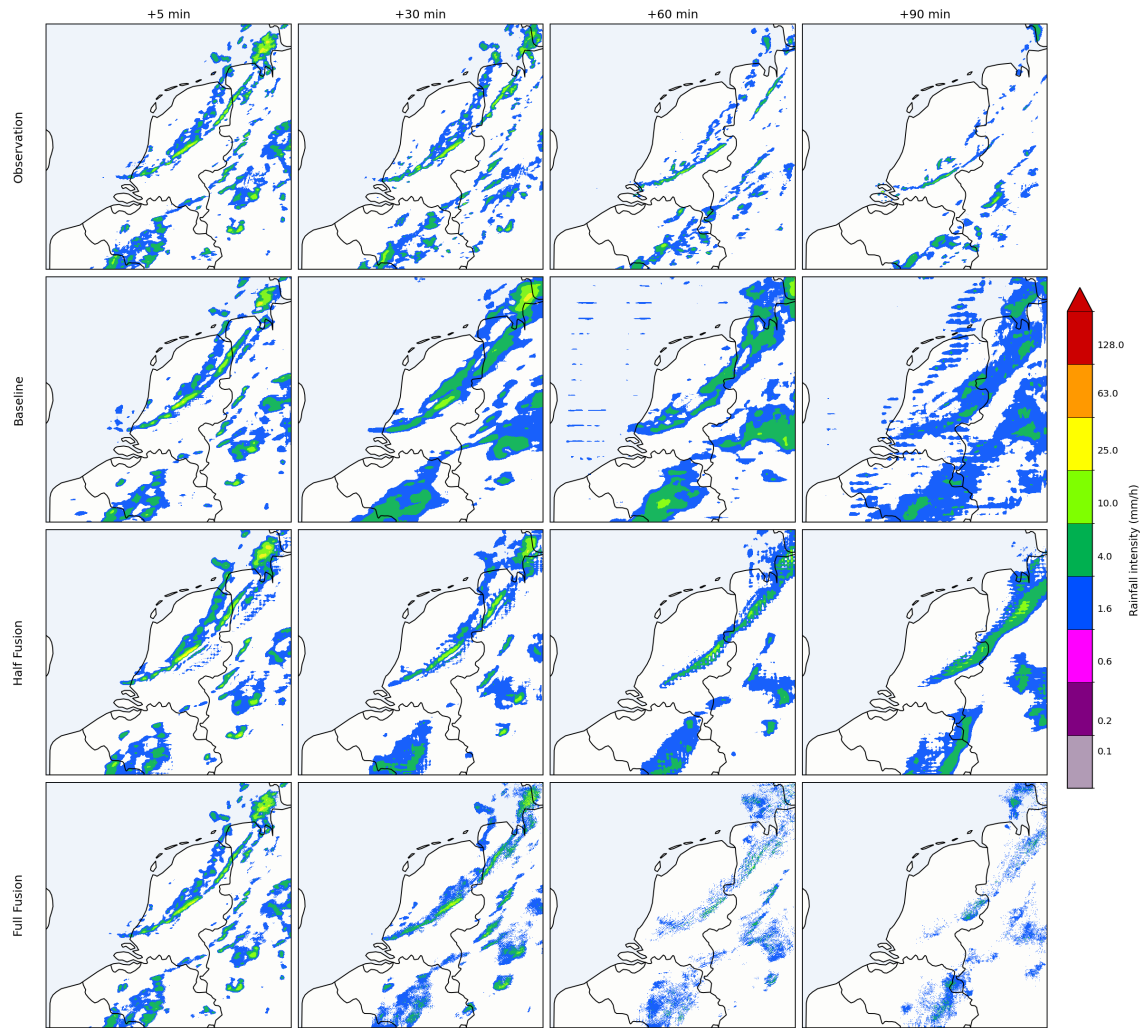


Figure C.3: Observed and predicted rainfall fields for the event initialized at 20:50 UTC on 04 August 2025. The rows show the observation, baseline, half-fusion, and full-fusion results, and the columns represent increasing forecast lead times. Rainfall intensity is given in mm h^{-1} .

Table C.3: Categorical metrics for event 4690 over the 5–90 min lead-time window at a rainfall threshold of 10 mm h^{-1} .

Event	Model	CSI ($\uparrow$)	POD ($\uparrow$)	FAR ($\downarrow$)
4690	Baseline	0.090	0.151	0.526
	Half fusion	0.123	0.160	0.682
	Full fusion	0.146	0.200	0.497

D

CTT-Conditioned Model Architecture

This appendix reports the implemented CTT-conditioned NowcastNet architecture. The model receives nine radar frames and nine cloud-top-temperature (CTT) frames at a spatial resolution of 512×512 and predicts 20 radar frames. A CTT validity mask is concatenated with the CTT sequence, hence the CTT input has 18 channels. The base channel width is 32. In all tables, feature-map dimensions are reported as channels $\times$ height $\times$ width; the batch dimension is omitted. Parameter counts include trainable weights, biases, normalization parameters, and the learned intensity-scaling parameter γ .

D.1. CTT-conditioned evolution network

The evolution network predicts both an intensity residual and a two-component motion field for each of the 20 future time steps. Radar and CTT information is fused after the first downsampling operation.

Table D.1: Layer-wise architecture of the CTT-conditioned evolution network.

Layer	Operation / input	Output feature map	Parameters
Radar input	Nine historical radar frames	$9 \times 512 \times 512$	0
inc	Residual DoubleConv, $9 \rightarrow 32$	$32 \times 512 \times 512$	14 596
down1	MaxPool + DoubleConv, $32 \rightarrow 64$	$64 \times 256 \times 256$	74 176
CTT + mask input	Nine CTT frames and nine masks	$18 \times 512 \times 512$	0
ctt_inc	Residual DoubleConv, $18 \rightarrow 32$	$32 \times 512 \times 512$	19 816
ctt_down1	MaxPool + DoubleConv, $32 \rightarrow 64$	$64 \times 256 \times 256$	74 176
fuse_x2	Concatenate $64 + 64$, 1×1 convolution	$64 \times 256 \times 256$	8256
down2	MaxPool + DoubleConv, $64 \rightarrow 128$	$128 \times 128 \times 128$	295 808
down3	MaxPool + DoubleConv, $128 \rightarrow 256$	$256 \times 64 \times 64$	1 181 440
down4	MaxPool + DoubleConv, $256 \rightarrow 256$	$256 \times 32 \times 32$	1 771 776
up1	Bilinear upsample, skip x_4 , DoubleConv	$128 \times 64 \times 64$	2 067 456
up2	Bilinear upsample, skip x_3 , DoubleConv	$64 \times 128 \times 128$	517 632

Layer	Operation / input	Output feature map	Parameters
up3	Bilinear upsample, skip fused x_2 , DoubleConv	$32 \times 256 \times 256$	129 792
up4	Bilinear upsample, skip x_1 , DoubleConv	$32 \times 512 \times 512$	46 496
outc + γ	1×1 convolution; intensity residual	$20 \times 512 \times 512$	680
up1_v--up4_v	Independent motion decoder	$32 \times 512 \times 512$	2 761 376
outc_v	1×1 convolution; x/y motion	$40 \times 512 \times 512$	1320
Evolution-network total			8 964 796

The four motion-decoder blocks have the same intermediate sizes as the intensity decoder: $128 \times 64 \times 64$, $64 \times 128 \times 128$, $32 \times 256 \times 256$, and $32 \times 512 \times 512$. The reported 680 parameters for the intensity head comprise 660 parameters in outc and 20 learned scalars in γ .

D.2. Generative encoder and CTT encoder

The generative encoder processes the concatenation of nine historical radar frames and 20 deterministic evolution predictions. The separate CTT encoder processes the same 18-channel CTT-plus-mask tensor used by the evolution network.

Table D.2: Generative and CTT encoder feature maps.

Branch	Layer	Output feature map	Parameters
Generative	Input (radar + evolution)	$29 \times 512 \times 512$	0
	inc	$32 \times 512 \times 512$	26 196
	down1	$64 \times 256 \times 256$	74 176
	down2	$128 \times 128 \times 128$	295 808
	down3	$256 \times 64 \times 64$	1 181 440
	Generative-encoder total		1 577 620
CTT	Input (CTT + mask)	$18 \times 512 \times 512$	0
	stem (h_1)	$32 \times 512 \times 512$	5216
	down2 (h_2)	$64 \times 256 \times 256$	32 832
	down4 (h_4)	$128 \times 128 \times 128$	131 200
	down8 (h_8)	$128 \times 64 \times 64$	262 272
	CTT-encoder total		431 520

D.3. Noise projector and generative decoder

A Gaussian noise tensor is sampled at $1/32$ resolution. The projector expands its channels while preserving spatial resolution. Its $1024 \times 16 \times 16$ output is rearranged by a parameter-free pixel reorganization into $64 \times 64 \times 64$. Concatenation with the $256 \times 64 \times 64$ generative feature produces the decoder input $320 \times 64 \times 64$.

Table D.3: Noise projector and CTT-conditioned generative decoder.

Layer	Operation / input	Output feature map	Parameters
Noise input	Gaussian sample	$32 \times 16 \times 16$	0
conv_first	3×3 convolution, $32 \rightarrow 64$	$64 \times 16 \times 16$	18 848
L1	Projection block, $64 \rightarrow 128$	$128 \times 16 \times 16$	227 712

Layer	Operation / input	Output feature map	Parameters
L2	Projection block, $128 \rightarrow 256$	$256 \times 16 \times 16$	905 984
L3	Projection block, $256 \rightarrow 512$	$512 \times 16 \times 16$	3 614 208
L4	Projection block, $512 \rightarrow 1024$	$1024 \times 16 \times 16$	14 437 376
Rearrange	Parameter-free pixel reorganization	$64 \times 64 \times 64$	0
		Noise-projector total	19 204 128
Feature concat.	Evolution feature + rearranged noise	$320 \times 64 \times 64$	0
fc	3×3 convolution, $320 \rightarrow 256$	$256 \times 64 \times 64$	737 536
head_0	SPADE residual block	$256 \times 64 \times 64$	1 794 176
ctt_fuse_h8	CTT spatial-channel modulation	$256 \times 64 \times 64$	82 820
Upsample	Nearest-neighbour, $\times 2$	$256 \times 128 \times 128$	0
G_middle_0	SPADE block, $256 \rightarrow 128$	$128 \times 128 \times 128$	1 248 704
G_middle_1	SPADE residual block	$128 \times 128 \times 128$	613 760
ctt_fuse_h4	CTT spatial-channel modulation	$128 \times 128 \times 128$	37 364
Upsample	Nearest-neighbour, $\times 2$	$128 \times 256 \times 256$	0
up_0	SPADE block, $128 \rightarrow 64$	$64 \times 256 \times 256$	522 944
ctt_fuse_h2	CTT spatial-channel modulation	$64 \times 256 \times 256$	9516
Upsample	Nearest-neighbour, $\times 2$	$64 \times 512 \times 512$	0
up_1	SPADE block, $64 \rightarrow 32$	$32 \times 512 \times 512$	249 152
up_2	SPADE residual block	$32 \times 512 \times 512$	115 520
ctt_fuse_h1	CTT spatial-channel modulation	$32 \times 512 \times 512$	2764
conv_img	3×3 output convolution	$20 \times 512 \times 512$	5780
		Generative-decoder total	5 420 036

D.4. Temporal discriminator

Table D.4: Temporal discriminator architecture.

Layer	Description	Output feature map	Parameters
Input	20-frame sequence	$20 \times 512 \times 512$	0
2-D branch	9×9 , stride 2	$64 \times 256 \times 256$	103 744
3-D branch ($k_t = 4$)	4 output channels	$4 \times 17 \times 256 \times 256$	1300
3-D branch ($k_t = 4$)	8 output channels	$8 \times 17 \times 256 \times 256$	2600
Concatenate	Flatten temporal dimension	$268 \times 256 \times 256$	0
block1	3×3 , stride 2	$128 \times 128 \times 128$	308 864
block2	3×3 , stride 2	$256 \times 64 \times 64$	295 168
block3	3×3 , stride 2	$512 \times 32 \times 32$	1 180 160
block4	3×3 , stride 1	$512 \times 32 \times 32$	2 359 808
BatchNorm	Batch normalization	$512 \times 32 \times 32$	1024
Head	3×3 patch-logit convolution	$1 \times 32 \times 32$	4609
		Discriminator total	4 257 277

D.4.1. Parameter summary

Table D.5: Parameter-count summary.

Component or model scope	Parameters
CTT-conditioned evolution network	8 964 796
Generative encoder	1 577 620
CTT encoder	956 064
Noise projector	19 204 128
Generative decoder	5 420 036
Generator training subset (evolution network excluded)	27 157 848
Complete inference model	36 122 644
Temporal discriminator (training only)	4 257 277
All instantiated networks during training	40 379 921