

Automated identification and location of three dimensional atmospheric frontal systems

Stefan Niebler¹, Bertil Schmidt¹^[0000-0003-2597-8331], Holger Tost²^[0000-0002-3105-4306], and Peter Spichtinger²^[0000-0003-4008-4977]

¹ Institut für Informatik, Johannes Gutenberg-Universität, 55099 Mainz, Germany

² Institut für Physik der Atmosphäre, Johannes Gutenberg-Universität, 55099 Mainz, Germany {stnieble}@uni-mainz.de

Abstract. We present a novel method to identify and locate weather fronts at various pressure levels to create a three dimensional structure using weather data located at the North Atlantic. It provides statistical evaluations regarding the slope and weather phenomena correlated to the identified three dimensional structure. Our approach is based on a deep neural network to locate 2D surface fronts first, which are then used as an initialization to extend them to various height levels. We show that our method is able to detect frontal locations between $500hPa$ and $1000hPa$.

Keywords: Weather Prediction · Atmospheric Physics · Deep Learning · Climate Change

1 Introduction

Weather fronts are important synoptic scale phenomena influencing the atmospheric environment and weather conditions surrounding it. They can be correlated to extreme weather events such as extreme precipitation [4, 14] or cyclones [16]. Fronts are usually depicted on a 2D-spatial grid, showing the global position of a front at the surface level [14, 8]. While this depiction can be sufficient to satisfy the information about the current location of a front, it neglects the fact that a front is actually a three dimensional, inclined structure instead of just a line in 2D-space. Previous methods for detecting fronts relied on data with low spatial resolution, which makes it hard to analyze frontal inclination due to horizontal coarseness. With recent high resolution data sets such as ERA5 [6] we can now provide spatially more accurate locations of fronts which allows us to determine improved representations of the 3D-structure of fronts.

Classical computational approaches for the detection of spatial 2D fronts rely on derivations of a thermal field such as wet bulb temperature, or equivalent potential temperature, respectively [8, 1, 7]. Many of these methods need coarser resolutions or a smoothed thermal field, since they were developed for synoptic scale features on low resolutions. The field of frontal detection is still highly active and tools are still refined regularly; e.g. [17]. Deep learning (DL) based methods have recently been successfully applied to several different problems in atmospheric physics from cyclone tracking [5] to forecasting [11, 15]. These

methods have shown to be able to exceed the performance of traditional numerical methods for example in the field of numerical weather prediction (NWP). Machine Learning (ML) has also been applied to the task of front detection. Previous work to detect 2D fronts used random forests [3] and neural networks [12, 10, 2]. In contrast to traditional algorithms these methods can be trained on high resolution grids and may therefore be directly applied to data sets at this resolution. In recent work [14], we presented a deep neural network for detecting and classifying frontal lines directly on ERA5 reanalysis data. The used data is provided as a latitude-longitude (lat-lon) grid at a resolution of 0.25° along both horizontal axes. In contrast to other DL-based front detection methods this model directly outputs the identified fronts as thin lines allowing for high spatial accuracy. Additionally, the model classifies the detected fronts into 4 types (*warm, cold, stationary, occluded*). While the evaluation was restricted to areas covering parts of the northern Hemisphere extending from North America over the Atlantic to Europe, we also showed qualitatively plausible results for other parts of the globe.

None of these methods is able to detect the actual three dimensional shape of a front. So far there is only one approach for the detection of three dimensional frontal fields [9]. It applies Hewson’s method to several height levels.

However, the approach performs several iterations of smoothing before frontal detection, which removes fine scale features, reducing the gained information from high resolution data. This is necessary due to the applied frontal detection algorithm, which relies on a smoothed temperature field for good results, since it was developed for coarse resolution data sets.

A typical problem for DL-based methods is the need for labeled training data. While the ECMWF provides atmospheric grids for several decades there exists - to the best of our knowledge - no data set regarding three dimensional structure of weather fronts. As such supervised training of a network to detect three dimensional fronts is not directly possible and other methods are needed to complement DL-based front detection.

In this work we propose a new method based on our recently described DL model [14]. We use the 2D fronts inferred by the network and add a new post processing pipeline, which extends them to frontal surfaces within the three dimensional ERA5 grid. To avoid processing a complete 3D grid, we also use the information obtained from the network output such as the frontal location and classification for more efficient computation.

Furthermore our model does not rely on fixed thermal thresholds to filter out wrong results such as the method proposed by [9]. Instead we use an adaptive threshold to filter false positives per frontal object.

As such our method is – to the best of our knowledge – the first to enable statistical analysis of three dimensional fronts over multiple timestamps. It is able to provide qualitatively good results, which are in agreement with theories regarding the shape and characteristics of a front. As we are not restricted to singular cases, our method can be a valuable tool in research of front related weather phenomena. In comparison to previous methods our algorithm can be

directly applied onto the ERA5 grid, without any smoothing, making full use of high resolution data.

2 Data

2.1 Atmospheric Data

Processing and evaluation is performed on ERA5 reanalysis data using pressure levels extending from 1000 hPa (surface) to 500 hPa (approx. 5km above sea level). For our algorithm we use data covering the area $[-90, 50]^\circ E$ and $[90, 0]^\circ N$. As we want to extract wide cross sections we only take surface front pixels into account, that are located within the region spanning $[-65, 25]^\circ E$ and $[65, 25]^\circ N$. This region mostly covers the North Atlantic, minimizing influences of orography such as mountains. The area also contains the northern mid latitudes, where frontal systems are commonly located [18]. The used ERA5 data does not mask invalid grid points which would be located within the terrain (e.g. data at 1000 hPa in Greenland), but rather uses interpolation to provide values for these areas. We do not alter this data except for the calculation of the equivalent potential temperature θ_e , which is a conserved quantity of reversible phase transitions, and thus well suited for investigations of fronts. It can be calculated from the atmospheric variables (pressure p , temperature T , water vapor mixing ratio q_v). We used the approximation provided by MetPy v1.3.1 [13].

2.2 Frontal types

Fronts are distributed into the 4 categories: **warm, cold, stationary, occluded**. Warm and cold fronts have a clear structure consisting of a warm and a cold side. Stationary fronts are mainly defined by their travelling speed and not by their gradient direction. Occluded fronts on the other hand are the result of a cold and a warm front overlapping. Thus gradient strength is reduced at lower levels, while they are split into two parts at higher altitudes, where the two fronts have not overlapped yet. Thus, we mainly focus on determining the three dimensional structure of warm and cold fronts, even though the network is able to provide surface fronts for all 4 categories [14].

3 Methods

Our pipeline consists of 4 stages processing the detected 2D surface fronts provided by the deep neural network. For each pixel in the surface front we calculate the normal direction and extract a cross section of the underlying thermal field along this vector. We evaluate each point of a cross section and perform local optimization to find the positions of the front at each processed pressure level within each cross section. The workflow is illustrated in Figure 1 and explained in more detail in the following subsections.

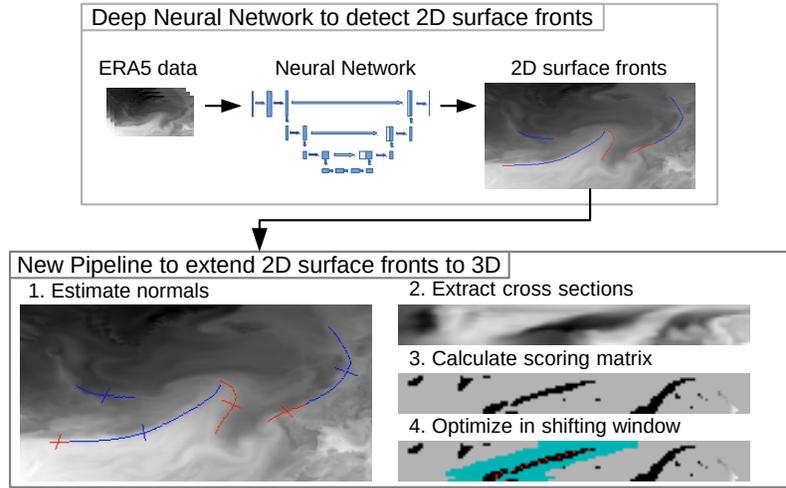


Fig. 1: Top Box: 2D surface fronts are detected from ERA5 data using a deep neural network. Bottom Box: Steps performed in our pipeline for one normal per front: (1) Normal directions of surface fronts are estimated; (2) cross sections along these normal directions are extracted; (3) each pixel is evaluated, creating a scoring matrix; (4) the minimum within a shifting window (turquoise area) is calculated and returned as front (red dots).

3.1 Surface front detection

Prior to our new pipeline we locate and classify surface fronts within the ERA5 data grid. For this task we use our neural network described in [14]. The network was trained using the frontal labels provided by two different weather services (German Weather Service, North American Weather Service), both of which covering parts of the Northern Atlantic region in their analysis. Our evaluation also shows that the network performs well over the sea covered surface - such as the Northern Atlantic, which we use in this work. The network output consists of a multi channel 2D grid where each channel denotes the locations of the different frontal classes (*warm*, *cold*, *stationary*, *occluded*) on the ERA5 grid.

3.2 Cross sections

Given the locations of the surface fronts for each class we identify frontal objects by their connectivity and ignore small objects in the following steps. For each front remaining we extract a cross section along an estimated normal for each pixel in each determined surface front. Each cross section contains values for the equivalent potential temperature θ_e , at each pressure level in our data set for 100 sample locations before as well as 100 sample locations behind the location of the surface fronts. Sample locations are determined by estimating a normal

and then sampling the temperature field at 20 km intervals along the normal direction. As the earth can be approximated by a sphere we estimate the normal on the tangential plane of each pixel on a sphere rather than on the flat lat-lon grid. Variable values for sampling locations not located directly on a grid point are interpolated. Points where the cross section exceeds the limits of our input region, as well as cross sections containing invalid data (e.g. due to an error in the files or interpolation with values outside the data region) are removed. The orientation of each cross section is chosen such that the sign of the dot product with the wind is positive.

Oversampling The determined surface fronts may be smaller - in terms of pixel count - than the same front at higher altitudes. For example a front may extend in a radial fashion where the radius increases with altitude, leading to a larger circumference of the circular sector. This may result in comparatively sparse samples of higher altitudes, even though the lower levels may be densely sampled. To compensate for this we create multiple samples from each surface front pixel by additionally creating cross sections with slightly deviated positions and/or angles. For angular deviations of each point we additionally sample cross sections where the normal is rotated by $\pm 30^\circ$, resulting in 3 different rotations including the non-rotated cross sections. Additionally we sample each of those 3 normal directions at shifted positions. We evaluate a total of 9 shifted positions. These shifted sample points are calculated as the points at a distance of $[-4, -3, \dots, 3, 4]$ from the surface front pixel in the direction perpendicular to the non rotated normal direction. This results in a total sampling of 27 cross sections per point.

Orientation correction It is important that cross sections are correctly oriented, because the algorithm may not correctly process wrongly oriented samples, as they exhibit contrary features such as an inverted sign of the gradient. However, in some cases the wind may be oriented (near) perpendicular to the front-normal. In these cases the orientation based on the wind direction may be unreliable, leading to wrongly oriented cross sections. Oversampling - mainly caused by the rotation - may further add some wrongly oriented samples. As such we adjust the orientation of samples by taking batches of each frontal objects pixels and reorient each sample within a batch based on their sign of the dot product with the mean direction of the batch.

3.3 Extension to other pressure levels

Once we obtain the 2D cross-section slices for each surface front pixel we can calculate the front location for each height level within that slice.

In a first step we flip the orientation of our cold fronts such that the cold region is located right to the center, as it should already be the case for warm fronts. This way we can calculate the positions regardless of the frontal type and potential differences in the sign of the gradients. A front typically bends over its cold region, meaning that the purely warm side has little to no information.

We can therefore cut large parts of the warm region located left of the center of each cross section to reduce computation, without a loss of information. We cut the first 60 pixels of each cross section, keeping a buffer of 40 pixels left of the center.

To find the frontal location in each cross section we first score each pixel of the cross section to create a score matrix. This score matrix is created evaluating the scoring functions as described in Subsection 3.4. Once the scoring matrix is calculated we can determine the frontal structure. Starting from the surface front which we initially locate at the center of the bottom of our cross section, we crop a smaller region of 40 pixels width centered at the x position of our surface front position and a height (extent in y -coordinate) of $h = 1$ pixel. We call this the current window. The optimal front location is then determined by minimizing a scoring function within the current window. Once we determined h separation points (1 point per pressure level) we shift our window such that it is now centered at the mean x position determined as our front location and covering the next set of h pressure levels. With this windowing approach we restrict the front detection to a smaller region where the simple optimization is applicable, as we assume that only one front is located within the window. The shifting of the window with increasing altitude allows us to potentially follow the fronts inclination across the whole width of each cross section, without the need to increase the window size. This method may fail if the difference between two succeeding height levels is larger than the distance covered by the window. Overall our approach essentially reduces the problem of locating a front in a three dimensional space to a set of 2D local optimization problems. As there is no dependency between cross section - even of the same object - the method can be easily adapted for parallel processing using multiple threads on GPUs or CPUs.

3.4 Scoring function

The presented algorithm finds the optimal position as the position that optimizes a scoring function at each pressure level. Our scoring function consists of three parts each being connected to a typical characteristic of weather fronts.

$$L_g = \nabla_x \theta_e \quad (1)$$

$$L_e = \left\| \left(\text{var}[\theta_e^{i,j+1}, \dots, \theta_e^{i,j+k}], \text{var}[\theta_e^{i,j-k}, \dots, \theta_e^{i,j-1}] \right) \right\|_2, i, j \in \mathbf{N} \quad (2)$$

$$L_d = \text{mean}[\theta_e^{i,j+1}, \dots, \theta_e^{i,j+k}] - \text{mean}[\theta_e^{i,j-k}, \dots, \theta_e^{i,j-1}], i, j \in \mathbf{N} \quad (3)$$

where index i indicate the vertical level, j denotes the horizontal grid position, and we set $k = 20$ as half the running window size of 40 respectively. Equation 1 describes the horizontal gradient of the temperature field. As we orient each sample such that the cold side is located at higher x indices the location of the strongest negative gradient is considered to be the ideal position for our front. This is consistent with the classic definition where the front is located at the zero pass of the thermal front parameter, which in our case simply becomes the 2nd

derivative of θ_e . We can use an optimization approach for this equation instead of a commonly seen threshold based decision, as

- a) we only look at a locally restricted area, where we can assume that the frontal gradient is the most dominant effect
- b) due to our initialization we already know that a front is located here, so we have a lower risk of false positives.

However, we need to respect that fronts may not extend all the way up to 500 hPa. We therefore added a variable threshold that only points where equation 1 is among the lowest 10% of all gradients within a front at each pressure level are considered valid locations for a front. As a result, points with very weak gradients are omitted and not seen as fronts. The latter threshold is adaptable to the general surroundings meaning that it works for both very strong fronts with a strong gradient as well as weaker fronts where the gradient may be not as dominant.

Equation 2 describes our assumption that the air masses before and after a front should be locally consistent, i.e., we do not expect high variation in the temperature within one air mass. As a result regions with very high temperature variance in any air mass are considered less ideal locations for a front.

Finally, Equation 3 describes the fact that the cold air mass is considered to have a lower mean equivalent potential temperature than the warm air mass. Our final scoring function therefore can be described as

$$L = w_g L_g + w_e L_e + w_d L_d \quad (4)$$

with positive weights $w_g, w_e, w_d \in \mathbf{R}$. For our evaluations we set all weights to 1.

3.5 Final processing

Once we calculated the separation point for each height level, we have obtained a 1D array consisting of the optimal frontal position for each level. If we combine all these result vectors for all cross section of a frontal object, we can create a point cloud sampling of the 3D frontal object. If desired one could use post processing techniques to create a 2D-surface instead. However, for our purposes point clouds are sufficient as we are interested in statistical evaluation of the resulting fronts. As our initial surface fronts are already classified we also have a classification for the three dimensional fronts by simply copying the initial frontal type for all levels.

4 Results

We present several applications of our method and show results for the evaluated dataset consisting of hourly reanalysis data for 2016. Due to a lack of a ground truth dataset of annotated 3D fronts we can only provide qualitative evaluations.

These evaluations show that the characteristics of our three dimensional fronts are in accordance with classical theoretical knowledge surrounding fronts and their expected behaviour. Further we can see that aggregated results as well as the visual presentation are plausible. Figure 2 shows warm and cold fronts as identified by our proposed method at 1000 hPa, 850 hPa and 650 hPa for 2016-09-23 00:00 UTC over a background of θ_e (interval: [270, 360] K). We can clearly see an inclination of the central meridional warm front, as it "moves" eastward with increasing altitude. While the central cold front stays roughly at the same location, indicating a much steeper inclination instead. Note that occluded and stationary fronts are not shown.

For all our tests we first calculated surface front positions and saved them to disk. This way the rest of the pipeline could be executed without the need for a GPU. In these evaluations we removed results where the optimal separation point was found to be at 0 location within the reduced cross section grid. In addition, all locations where no valid separation location was found were also set to 0 before the filter was applied, removing those as well from the evaluation. This results in approximately 7% to 20% of cross sections being filtered per level. Here we can observe that the lower and upper levels contain more filtered samples compared to the levels around 850 hPa, which indicates that frontal characteristics are more pronounced at this level instead of high altitudes or near surface. Potential causes why no valid separation could be found include falsely identified surface fronts, weakly expressed frontal characteristics at the edges of a front, bad samples caused by the additional sampling or that the front is not present at this level at all.

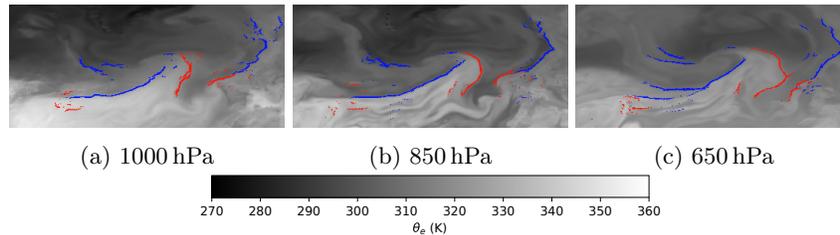


Fig. 2: Locations of cold (blue) and warm (red) fronts at 1000 hPa (a), 850 hPa (b) and 650 hPa (c) for 2016-09-23 00:00 UTC. Shown Snippet Coordinates: $[70, 30]^\circ\text{N}$, $[-80, 10]^\circ\text{E}$. Background: θ_e (interval: [270, 360] K)

4.1 Probability density distribution of temperature difference

A typical warm or cold front is classified by a strong temperature gradient between two air masses. During the passing of a warm front the temperature rises, while for a cold front it falls. For our cross sections this means that a warm front

should have a cold side to the right side of the center (i.e. the side the front has not yet passed) and a warm side at the left of the center (i.e. the side where the front already passed through). For a cold front the location of the warm and cold side are swapped.

To highlight the effect of a passing front we plotted the distribution of the difference of the mean temperature in a region 200 km behind the identified front and the mean temperature in the region 200 km ahead of the front at 500 hPa, 850 hPa and 1000 hPa for the results of our algorithm. We further plot the results of a baseline at 850 hPa, where the locations at which we evaluate the temperature difference within a cross section were randomly sampled instead.

Algorithm results Figure 3 shows the histogram of temperature differences for fronts identified by our algorithm. As expected we can see for all presented levels that the temperature does indeed rise (fall) with the passing of warm (cold) fronts. We can further see that the strongest temperature difference can be found at 850 hPa, a level commonly used for detecting fronts. Further the plot shows that the temperature difference at 1000 hPa is higher than at 500 hPa, indicating that the temperature gradient at higher levels is less expressed than at (near) surface level. For both cold and warm fronts we can see that a small percentage of identified front locations express a wrong sign in the temperature difference, which is most likely caused by false positive frontal positions or wrongly oriented samples.

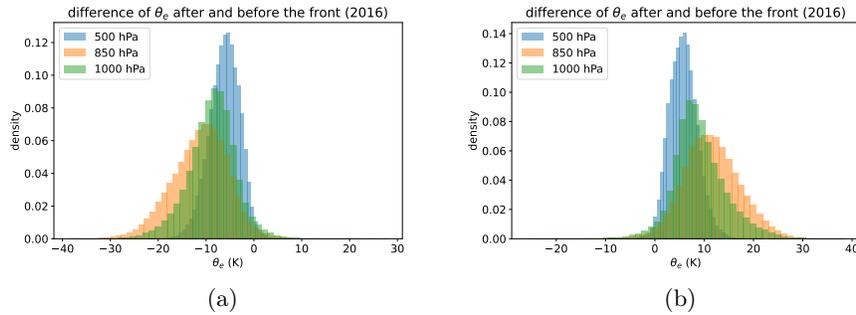


Fig. 3: Histograms showing the distribution of temperature differences across (a) cold and (b) warm fronts as calculated by our method at different height levels.

Randomly sampled baseline As a comparison we plotted the temperature differences at 850 hPa if the frontal separation was uniformly randomly sampled instead of using our algorithm in Figure 4. The data for the random samples consists of the same cross sections used for the cold (warm) fronts. To create these plots we aggregated the results every 168 hours (1 week) within 2016. In

addition to the histogram we also fitted both a normal and a Laplace distribution to the data. As can be seen the Laplace distribution provides a good fit for the temperature difference of the random samples. Further the choice of warm or cold front cross sections barely changes the distribution of the random samples. In both cases the mean μ and the peak is located near 0. We realize that in this case the position of the standard deviation σ of both distributions almost coincide, thus it makes sense to use σ as a measure for testing the quality of the derived probability densities.

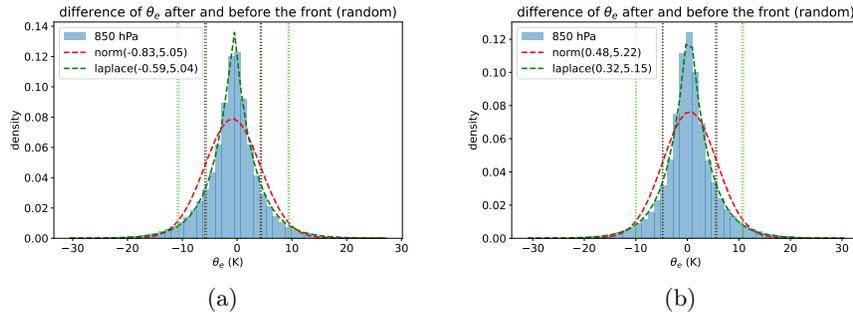


Fig. 4: Histograms of temperature differences for randomly separated cross sections. Additionally displays a fitted laplace and normal probability density function as well as mean ± 1 or 2 standard deviations.

4.2 Temperature difference at various levels

In this section we present results for all height levels, showing how the algorithm fairs against the randomly sampled case and showing that the algorithm does provide meaningful results. In Figure 5, we plotted the mean temperature difference of the random separation case, as well as $\mu \pm \sigma$ and $\mu \pm 2\sigma$ for each height level. Additionally boxplots of the results of our algorithm at each evaluated pressure level were added to highlight the differences to the randomly sampled case.

As seen in the previous section the most dominant temperature difference, even for the random case, lies around 850 hPa. This is in agreement with various non-DL-based front detection algorithms, that rely on the thermal field at 850 hPa to determine the location of fronts. Our results confirm that this commonly used but empirically chosen pressure level is indeed a good choice.

In comparison to the random case, the plots show that for most cases our results are at least at a σ distance from μ , with the more extreme samples even exceeding the 2σ distance. Mean (green triangle) and median (orange line) also tend to be located closely to the 2σ line, sometimes even exceeding it.

As mentioned in Subsection 4.1 and Figure 4, we can see that the random fronts temperature difference distribution can be approximated using a laplace distribution $L(\mu, b)$ with mean $\mu \in \mathbf{R}$ and scale $b \in \mathbf{R}$. We know that the standard deviation can be calculated as $\sigma = \sqrt{2}b$ and from this we can calculate that a point randomly sampled from a laplace distribution has a probability of approx. 75% (resp. 94%) to lie within the interval $[\mu - \sigma, \mu + \sigma]$ (resp. $[\mu - 2\sigma, \mu + 2\sigma]$). This further enhances our point, that the extracted 3D fronts are meaningful.

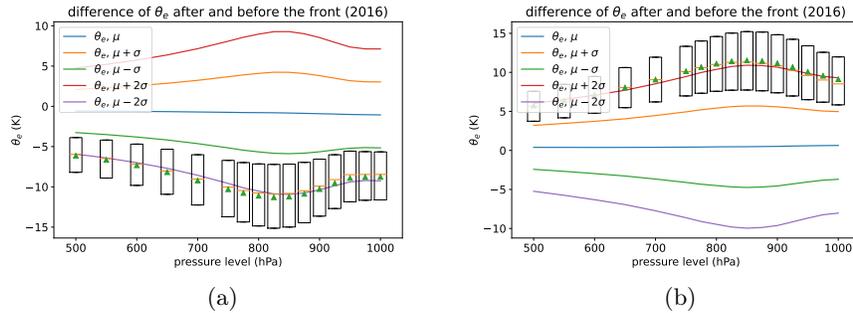


Fig. 5: Temperature difference across the detected fronts for different pressure levels for (a) cold and (b) warm fronts as box plots including their mean (green triangle) and median (orange line). Additionally the mean (μ) as well as the one and two σ intervals for the randomly separated fronts is inserted as orientation.

4.3 Evaluation of frontal inclination

In a third evaluation we determine the inclination of weather fronts and provide a histogram of how the inclination is distributed across fronts of each type. We determine the inclination in degrees using the following formula:

$$\text{inclination} = \arctan \left(\frac{|\text{pressure_difference (hPa)}|}{\text{horizontal_distance (km)}} \right) \quad (5)$$

While the formula uses different units, it has the advantage of better showing the inclination, as opposed to describing the vertical difference in terms of km, since the atmosphere's length scales are very different and thus the aspect ratio (vertical vs. horizontal scale) is very small. For instance, the horizontal extent of a pixel is 20 km while the vertical distance from 1000 hPa to 500 hPa is only approximately 5 km. Determining the inclination on this scale would make it hard to see differences in the angles and results in mostly near zero inclinations. In Figure 6 we show the mean cold (a) and warm (b) front as well as the average inclination of the frontal objects from 1000 hPa to 600 hPa (left) and from 1000 hPa to 850 hPa (right). To calculate the inclination of a frontal object

we first determine the median position of the frontal separation points for each pressure level. The inclination between two pressure levels is then described by the line connecting the resulting positions for the corresponding pressure levels (e.g. 1000 hPa and 850 hPa or 600 hPa). The median position is chosen as it is more robust towards outliers, where the algorithm may have failed to determine one of the separation points correctly. The mean inclination of frontal objects as depicted in Figure 6 is then obtained by connecting the median separation positions averaged over all fronts. In case of the warm fronts (Fig. 6b) we can clearly see the expected flat inclination, while for the cold fronts (Fig. 6a) we can see a rather steep inclination of near 90° and a slight backward tilt for the inclination. Both of these observations are in accordance with the typical theory surrounding the shape and inclination of warm and cold fronts.

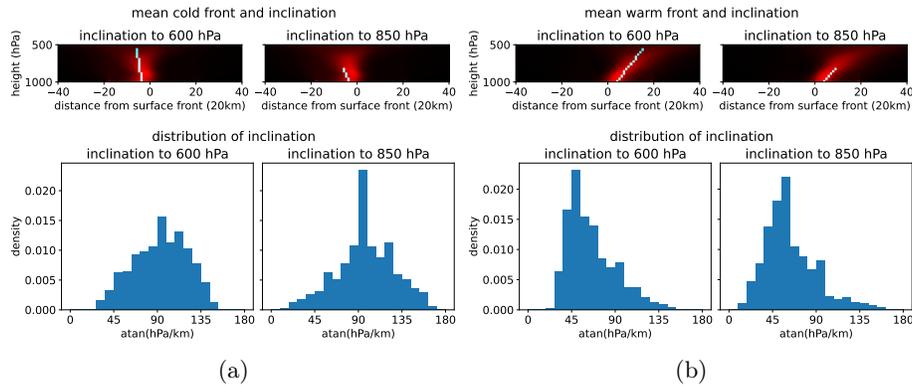


Fig. 6: (a): Average frontal separation for cold fronts in red. Light blue line indicates the average inclination from surface ($1000hPa$) to $600hPa$ (left) or $850hPa$ (right). Histograms show the distribution of inclinations accordingly. (b): as (a), but for warm fronts.

4.4 Runtime

Tests were run on a compute node with an Intel 2630v4 CPU with 10 cores and 128GB RAM. For the pipeline we used precalculated fronts from the mentioned neural network. Thus, frontal locations can be directly forwarded to the pipeline. The pipeline itself was implemented in Python.

Parallelization is performed by splitting the dataset into multiple batches of 6 days at hourly time resolution resulting in 144 timestamps per batch. For each batch all types of fronts were calculated, with two calculations for occluded fronts. Since intermediate cross sections are stored in RAM instead of main memory we requested $10GB$ for each batch. Processing a batch took less than 150 minutes, varying with the amount of frontal pixels present at each

timestamp. Per sample timestamp this results in a runtime of approximately 1 minute. As we did not store the cross section results our evaluations also had to run the complete pipeline, resulting in similar runtimes. Reducing the size of cross sections as well as the oversampling rate could further reduce runtime and memory consumption, however we decided to keep those values high, to allow a more thorough evaluation of the results, as runtime optimization was not a main focus of this work.

4.5 Potential Extensions

ERA5 data contains theoretically invalid regions such as high pressure levels in Greenland. In addition to these errors there is also a strong influence at the ground level temperature based on the orography. As a result, some fronts, located in the vicinity of land surface, may exhibit dominating ground gradients based not on the frontal activity but rather the orography. These may lead to wrongly extracted frontal locations on the lower levels, where this effect is more dominant.

Our algorithm is only able to extend identified surface fronts. It is not designed to identify new fronts that may not extend all the way to the surface. Further, occluded fronts are currently not investigated, as the algorithm does not support the case, where a front splits into several parts at higher altitudes. Especially as in the case of an occluded front these parts have differently signed gradients.

Further, we only evaluated our algorithm within the North Atlantic region. Performance on other regions needs to be further evaluated, especially regarding orographic features such as high mountains, which are less represented in our data. However, it is well known that fronts as synoptic weather features are heavily perturbed by mountainous regions on continents, thus losing their clear structure. Therefore, it is even not clear that extending our algorithm to such regions is really meaningful.

5 Conclusions and Future Work

In this paper we presented a novel pipeline for finding three dimensional structures of frontal lines based on a surface front detection neural network. Our algorithm uses the prior knowledge of frontal characteristics and surface front positions to expand frontal lines from a two dimensional grid to three dimensional structures. Our model is the first that enables statistical analysis of three dimensional fronts. Therefore, we can provide a statistical evaluation of characteristics of automatically detected three dimensional fronts, which previous methods were not suited to do. Our method may thus provide important insights, deepening the understanding of fronts and their three dimensional structure and how these are connected to other weather phenomena. The code of the new pipeline will be available at <https://github.com/stnie>.

Possible enhancements of the proposed method include extensions to other regions and providing more statistical evaluations regarding similarities and differences in the frontal structure over the globe. This would enable further investigation on how the structure of a front influences phenomena often correlated to frontal activity such as extreme precipitation. Our current pipeline focuses on the warm and the cold weather fronts. While occluded fronts can possibly be detected and found by our algorithm, it is currently not suited to describe the nature of such fronts as two overlapping fronts. It may therefore be of interest to extend this approach to cover the more complex shapes of occluded fronts.

Interesting computational extensions would include the generalization of the proposed method to enable more complex structures including the detection of frontal three dimensional volumes instead of surfaces. Storing the cross sections on hard disk would require several TB of memory due to the oversampling of each pixel and the large horizontal extent. To circumvent this we process each timestamp individually. Even with multiple cores running in parallel and by splitting the dataset into multiple batches, computation still takes several hours. For a better applicability and faster experimentation we therefore plan to provide a parallelized version of the pipeline on GPU systems.

Another interesting research direction would be the design of a neural network similar to our previous work, to directly detect three dimensional fronts. However, as currently there is no valid ground truth 3D data regarding weather fronts, such an approach is not feasible at the moment. We do however believe that creation of such a network would allow us to leverage the deep neural networks' ability to model complex shapes and phenomena. This approach could, as proven in the 2D case, create clearer results and reduce computation time. In addition it may be able to better generalize, i.e. frontal activity over challenging terrain or detecting the more complex shaped frontal types.

Acknowledgements The study is supported by the project “Big Data in Atmospheric Physics (BINARY)”, funded by the Carl Zeiss Foundation (grant P2018-02-003). We acknowledge the ECMWF for providing access to the ERA5 reanalysis data and the ZDV of JGU for providing access to Mogon II. We further acknowledge Daniel Kunkel for supporting us with data management and thank Michael Wand for fruitful discussions.

References

1. Berry, G., Reeder, M.J., Jakob, C.: A global climatology of atmospheric fronts. *Geophysical Research Letters* **38**(4) (2011)
2. Biard, J., Kunkel, K.: Automated detection of weather fronts using a deep learning neural network. *Advances in Statistical Climatology, Meteorology and Oceanography* **5**, 147–160 (11 2019)
3. Bochenek, B., Ustrnul, Z., Wypych, A., Kubacka, D.: Machine learning-based front detection in central europe. *Atmosphere* **12**(10) (2021)
4. Catto, J.L., Pfahl, S.: The importance of fronts for extreme precipitation. *Journal of Geophysical Research: Atmospheres* **118**(19), 10,791–10,801 (2013)

5. Giffard-Roisin, S., Yang, M., Charpiat, G., Kumler Bonfanti, C., Kégl, B., Monteleoni, C.: Tropical cyclone track forecasting using fused deep learning from aligned reanalysis data. *Frontiers in Big Data* **3** (2020)
6. Hersbach, H., Bell, B., Berrisford, P., Hirahara, S., Horányi, A., Muñoz-Sabater, J., Nicolas, J., Peubey, C., Radu, R., Schepers, D., Simmons, A., Soci, C., Abdalla, S., Abellan, X., Balsamo, G., Bechtold, P., Biavati, G., Bidlot, J., Bonavita, M., De Chiara, G., Dahlgren, P., Dee, D., Diamantakis, M., Dragani, R., Flemming, J., Forbes, R., Fuentes, M., Geer, A., Haimberger, L., Healy, S., Hogan, R.J., Hólm, E., Janisková, M., Keeley, S., Laloyaux, P., Lopez, P., Lupu, C., Radnoti, G., de Rosnay, P., Rozum, I., Vamborg, F., Villaume, S., Thépaut, J.N.: The era5 global reanalysis. *Quarterly Journal of the Royal Meteorological Society* **146**(730), 1999–2049 (2020)
7. Hewson, T.D.: Objective fronts. *Meteorological Applications* **5**(1), 37–65 (1998)
8. Jenkner, J., Sprenger, M., Schwenk, I., Schwierz, C., Dierer, S., Leuenberger, D.: Detection and climatology of fronts in a high-resolution model reanalysis over the alps. *Meteorological Applications* **17**(1), 1–18 (2010)
9. Kern, M., Hewson, T., Schätler, A., Westermann, R., Rautenhaus, M.: Interactive 3d visual analysis of atmospheric fronts. *IEEE Transactions on Visualization and Computer Graphics* **25**(1), 1080–1090 (2019)
10. Lagerquist, R., McGovern, A., II, D.J.G.: Deep learning for spatially explicit prediction of synoptic-scale fronts. *Weather and Forecasting* **34**(4) (2019)
11. Lam, R., Sanchez-Gonzalez, A., Willson, M., Wirnsberger, P., Fortunato, M., Pritzler, A., Ravuri, S., Ewalds, T., Alet, F., Eaton-Rosen, Z., Hu, W., Merose, A., Hoyer, S., Holland, G., Stott, J., Vinyals, O., Mohamed, S., Battaglia, P.: Graphcast: Learning skillful medium-range global weather forecasting (2022). <https://doi.org/10.48550/ARXIV.2212.12794>
12. Matsuoka, D., Sugimoto, S., Nakagawa, Y., Kawahara, S., Araki, F., Onoue, Y., Iiyama, M., Koyamada, K.: Automatic detection of stationary fronts around japan using a deep convolutional neural network. *SOLA* **15**, 154–159 (2019)
13. May, R.M., Goebbert, K.H., Thielen, J.E., Leeman, J.R., Camron, M.D., Bruick, Z., Bruning, E.C., Manser, R.P., Arms, S.C., Marsh, P.T.: Metpy: A meteorological python library for data analysis and visualization. *Bulletin of the American Meteorological Society* **103**(10), E2273 – E2284 (2022). <https://doi.org/10.1175/BAMS-D-21-0125.1>, <https://journals.ametsoc.org/view/journals/bams/103/10/BAMS-D-21-0125.1.xml>
14. Niebler, S., Miltenberger, A., Schmidt, B., Spichtinger, P.: Automated detection and classification of synoptic-scale fronts from atmospheric data grids. *Weather and Climate Dynamics* **3**(1), 113–137 (2022)
15. Pathak, J., Subramanian, S., Harrington, P., Raja, S., Chattopadhyay, A., Mardani, M., Kurth, T., Hall, D., Li, Z., Azizzadenesheli, K., Hassanzadeh, P., Kashinath, K., Anandkumar, A.: Fourcastnet: A global data-driven high-resolution weather model using adaptive fourier neural operators (2022). <https://doi.org/10.48550/ARXIV.2202.11214>
16. Pfahl, S., Sprenger, M.: On the relationship between extratropical cyclone precipitation and intensity. *Geophysical Research Letters* **43**(4), 1752–1758 (2016)
17. Sansom, P.G., Catto, J.L.: Improved objective identification of meteorological fronts: a case study with era-interim. *Geoscientific Model Development Discussions* **2022**, 1–19 (2022)
18. Schemm, S., Sprenger, M., Wernli, H.: When during their life cycle are extratropical cyclones attended by fronts? *Bulletin of the American Meteorological Society* **99**(1), 149–166 (JAN 2018). <https://doi.org/10.1175/BAMS-D-16-0261.1>