



Jet Propulsion Laboratory
California Institute of Technology

OPERA Algorithm Theoretical Basis Document for SAR Disturbance from RTC data

D-108776, Revision A

Version 1.0

May 9, 2026

Richard West, Charles Marshak, Talib Oliver Cabrera, Jungkyo Jung

National Aeronautics and
Space Administration
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California

Signature Page

Prepared by:

Electronic Signature on File

Richard West, OPERA ADT/DIST-S1 Lead Date

Electronic Signature on File

Charles Marshak, OPERA ADT Engineer Date

E-SIGNED by Talib Cabrera on 2026-06-03 23:49:06 GMT 2026-06-03 23:49:06

Talib Oliver Cabrera, OPERA ADT Engineer Date

E-SIGNED by Jungkyo Jung on 2026-06-03 22:41:32 GMT 2026-06-03 22:41:32

Jungkyo Jung, OPERA ADT Engineer Date

Approved by:

Electronic Signature on File

Steven Chan, OPERA Project Scientist Date

Electronic Signature on File

Luca Cinquini, OPERA Project Manager Date

EPDM Approvals:

D-108776/A-OPERA Algorithm Theoretical Basis Document (ATBD) for SAR Disturbance from RTC data

Status: In Approval

Creation Date: 03-Jun-2026 15:32

Status Date: 03-Jun-2026 15:35

Type: JPL Document Item Revision

Release Date:

Overview

Universal Viewer

Reference Files

Participants

History

Signoffs

Relations

Audit Logs

Mode

All

Task	Status	Performer	Assignee Origin	Due Date	End Date
Document Approver List Approval : Signoff	Approve	Chan, Steven K (chan)	Document Approvers	12-Jun-2026 09:00	04-Jun-2026
Document Approver List Approval : Signoff	Approve	West, Richard D (richw)	Document Approvers	12-Jun-2026 09:00	03-Jun-2026
Document Approver List Approval : Signoff	Approve	Marshak, Charles Z (cmarshak)	Document Approvers	12-Jun-2026 09:00	03-Jun-2026
Document Approver List Approval : Signoff	Approve	Cinquini, Luca (cinquini)	Document Approvers	12-Jun-2026 09:00	03-Jun-2026

Document Change Log

Revision	Cover Date	Sections Changed	ECR #	Reason, ECR Title, LRS #
Initial	November 20, 2024	All	N/A	New Document
0.2	Dec 11, 2024	Product Table	N/A	Minor Adjustment of Dist-Status to include No Data
1.0, Revision A	May 9, 2026	All	N/A	High level workflow diagram (Fig 1) updated for operational input curation and confirmation process. Reorganized algorithm description sections to align with algorithm execution and current operational system including accurate description of 3-year anniversary lookback procedure for curating inputs and confirmation process for accumulating disturbances over time. Added figures for confirmation and sample product. Removed calibration section and integrated into the algorithm description including new studies about threshold selection. Discussions of other metrics using our event database and discussion of accuracy measures has been moved to appendices.

TBD/TBC Log

Section	Description	Due Date

1	Introduction	1
1.1	Background	1
2	The DIST-S1 Algorithm	4
2.1	Despeckling	4
2.2	Baseline Curation	6
2.3	Model Description and the DIST-S1 Disturbance Metric	7
2.4	Neural Network Architecture, Training, and Inference	13
2.5	Disturbance Delineation	15
2.5.1	Additional Disturbance Threshold Calibration	17
2.6	Aggregating Burst Disturbance into a Tiled Product	18
2.7	Disturbance Confirmation and Accumulation	20
2.8	The DIST-S1 Product: Formatting and Organization	24
3	Remarks on Validation	28
3.1	Requirements for DIST-S1 Accuracy	29
4	Data Access	30
5	Acronyms	30
6	Acknowledgements	31
A	Classical Disturbance Metrics	37
B	Accuracy Measures	40
C	Comparison of the Transformer Metric to CuSum	42

1 Introduction

The Earth’s surface undergoes continuous changes driven by a combination of seasonal and interannual processes, natural disasters, anthropogenic activity, and a warming climate. These changes manifest through diverse environmental stressors: urbanization alters local hydrology and climate; industrial mining degrades topsoil composition and water quality; deforestation disrupts vegetation structure and increases carbon emissions; hurricanes destroy infrastructure and uproot biomass; drought restricts primary production; and heavy rains trigger landslides in sloped terrain. Delineating and mapping surface disturbances as they occur has important implications for ecosystems, water resources, food production, disaster response efforts, and other essential systems for human society. As such, the OPERA project is committed to providing a suite of products derived from Earth-orbiting sensors for reliable, global-scale disturbance monitoring. Our SAR-derived *generic* disturbance algorithm identifies pixels in space and time that deviate from the estimated statistics determined from a set of baseline images. In this document, we detail the design of the OPERA DIST-S1 algorithm—its theoretical basis and product structure—for tracking generic disturbances from the Sentinel-1 constellation using the OPERA Radiometric and Terrain Corrected Imagery (RTC-S1). The DIST-S1 product will provide a complementary view of the disturbances captured in the existing OPERA optical disturbance product DIST-HLS (derived from harmonized Landsat and Sentinel-2), which began production in April 2023. The RTC-S1 input imagery provides structural information about the surface being observed: double bounce from tall buildings, volumetric scattering from dense vegetation, and specular reflection from smooth surfaces such as roads and water. The RTC-S1 also reveals information about water content in soil and in vegetation. DIST-S1 therefore characterizes generic disturbance through statistical deviations of the latest image from the baseline. This is in contrast to the DIST-HLS disturbance mechanism, which is derived from the interannual vegetation loss determined by modeling per-pixel vegetation cover. Because the OPERA RTC-S1 input data is weather- and illumination-independent, DIST-S1 will provide monitoring capabilities particularly where continuous cloud cover or smoke obscure a scene. The DIST-S1 product is scheduled for release in mid-2026.

1.1 Background

Disturbance delineation is the process of mapping deviations from a baseline of reference images. It can also be viewed as the process of selecting a time and area within a time-series of co-registered images in which a disturbance or change occurs. Disturbance delineation is frequently referred to as *change detection* in the literature. We will equate *change* and *disturbance* in this document. Our goal is to use the OPERA RTC-S1 product for disturbance mapping alongside the OPERA optical disturbance product derived from Harmonized Landsat and Sentinel-2 (DIST-HLS). This goal motivates many of our algorithm choices.

Before discussing our approach, we review two operational SAR disturbance products

that demonstrate SAR’s power for detecting interannual forest loss. These products are the SAR-based RAdar for Detecting Deforestation (RADD) products [1] and the ALOS-2 Forest Alerts [2, 3]. The input imagery for these products is the C-band Sentinel-1 constellation and ALOS-2, respectively. These products’ algorithms focus on tropical *forest loss*, which is highly visible in SAR imagery because volumetric scattering from trees significantly decreases as trees are cleared. Such products are invaluable for large-scale monitoring where the persistent cloud cover makes consistent optical observations challenging. The DIST-S1 product will have a wider geographic scope: it will be nearly global, covering all DIST-HLS tiles with sufficient RTC-S1 availability. In contrast, OPERA DIST-S1 will identify *generic* disturbances, i.e., disturbances defined by statistical deviations from a baseline of reference images. One obvious distinction between the aforementioned forest loss products and OPERA DIST-S1 in forested areas is that DIST-S1 also detects anomalous interannual regrowth.

We now provide a broader overview of disturbance delineation and its challenges. Any method for delineating disturbance will depend on the following factors:

- (a) the modality of the sensor (i.e., optical or SAR),
- (b) its temporal sampling rate,
- (c) its spatial resolution,
- (d) the category of change being studied,
- (e) a change’s duration,
- (f) the baseline (i.e., reference) time-series of undisturbed pre-image observations, and
- (g) the quantity of post-image data compared against the baseline time-series.

Items (a) - (c) are sensor constraints. Items (d) - (e) are related to the type of disturbances being tracked. Items (f) - (g) address the balance between the preceding sensor and disturbance constraints. In (f), the reference image time-series is referred to in this document as *pre-images*. In (g), the imagery compared to the set of pre-images is called *post-image*. The OPERA DIST-S1 product will have a 30 meter spatial resolution following the OPERA RTC-S1 input data and will be reprojected into the Military Grid Reference System (MGRS) tile system so the products can be easily compared to DIST-HLS products. The DIST-S1 will have a nominal 6-day repeat pass through a single observation geometry in forward processing mode when continuous data generation is possible using two platforms from the Sentinel-1 constellation. As we will discuss in later sections, we will accumulate disturbances through multiple SAR geometries so the sampling rate for DIST-S1 can be much higher if data is available. For example, over Los Angeles, the sampling rate for DIST-S1 is approximately 3 days; in fact there are descending and ascending tracks that pass over the same area in a 24-hour period.

We provide broad categories of change detection algorithms following the OPERA DIST-HLS ATBD [4]. These categories contextualize the algorithms used to analyze change. A change detection algorithm can be either *signal*-based or *theme*-based. A signal-based approach uses a radiometric measure directly from the sensor, e.g., optical reflectance or SAR

backscatter, to delineate change. A classical signal-based SAR change detection algorithm is the log-ratio, denoted here as ℓ , proposed in [5], where the amount of change between a pre-image I_0 and post-image I_1 is measured via:

$$\ell := \log \left(\frac{I_1}{I_0} \right) = \log(I_1) - \log(I_0). \quad (1)$$

A theme-based approach first models a geophysical variable, e.g., forest cover, and then uses the modeled variable for change detection. The OPERA DIST-HLS optical product is theme-based in that it models *percent vegetation cover* to detect change. The SAR-based RADD algorithm is theme-based in that it constructs forest and nonforest distributions based on backscatter values, iteratively updating the distribution based on Bayes' rule [1, 6, 7]. Signal-based approaches are tied directly to acquired imagery, while theme-based approaches clearly communicate the type of change detected.

Change detection algorithms can also be categorized as *metric* or *classification* approaches. A metric approach assigns a non-negative real value to each pixel measuring the degree of disturbance relative to the pre-images. By convention, larger metric values indicate greater change and smaller values indicate less change. For DIST-HLS, vegetation cover loss is the disturbance metric [4]: greater loss corresponds to greater disturbance. For RADD, the metric is the probability of non-forest pixels exceeding a threshold in the interval $[0.5, 1)$ [7]. A classification approach directly labels each pixel as changed or unchanged. Classification approaches are typically data-driven, trained on ground-truth datasets targeting specific change types such as urban growth [8,9] or floods [10,11]. Such algorithms often employ *bi-temporal* methods, using a single pre-image and a single post-image to determine change.

This taxonomy allows us to more precisely characterize the DIST-S1 algorithm. We highlight important differences between optically-derived and SAR-derived disturbances. While the DIST-S1 product will complement the DIST-HLS product, the set of mappable disturbances is not strictly overlapping. DIST-S1 measures structural change using a signal-based metric of statistical deviations in interannual backscatter. DIST-HLS thematically tracks interannual vegetation cover loss through the spectral signature of its input. All disturbances in the OPERA disturbance suite (both DIST-HLS and DIST-S1) do not attribute *why* a disturbance occurred (fire, development, etc.), though DIST-HLS's thematic measure indicates *what* was disturbed, namely vegetation. DIST-S1 generic disturbances indicate that the current imagery deviated from the multi-year baseline and can include changes as varied as urban development, floods, or wildfires. DIST-S1 and DIST-HLS will agree for dense vegetation clearing. However, SAR will not detect burn scars or vegetation loss over short, dry, sparse vegetation, e.g., plains that appear visibly burnt in post-fire optical imagery. DIST-S1 will detect a broader range of change types than DIST-HLS, including urban development and demolition; vegetation regrowth; drought and sandbar expansion; agricultural rotation; and snow cover. We discuss additional differences in Section 3.

2 The DIST-S1 Algorithm

The DIST-S1 Algorithm is a signal-based approach to delineating changes in OPERA RTC-S1 images. The algorithm uses a probabilistic metric to quantify changes in recent data collection (post-image) when comparing to an interannual baseline (pre-images). Specifically, the metric utilizes a vision transformer to estimate the per-pixel distribution from the baseline and the metric quantifies the per-pixel deviation from these estimates. To establish a baseline (i.e. estimate the per-pixel distribution), 10 reference dates from the previous 3 years are used. The details of this baseline curation is found in Section 2.2. The Vision Transformer (ViT) [12] captures rich spatial and temporal patterns within the baseline that in turn can allow the algorithm to identify deviations across numerous environments and terrain. In Figure 1, we provide a high-level view of our algorithm, its inputs and outputs. We also include Table 1 showing the ancillary data used in our product. We highlight that our methodology will output a new DIST-S1 ALERT product for each new Sentinel-1 acquisition. The rest of this section details the processes (blue squares) in the flow chart. In Section 2.1, we discuss how the OPERA RTC-S1 data is despeckled. In Section 2.2, we describe the mechanism for curating the baseline set. In Section 2.3, we detail the technical aspects of our disturbance model and metric for the DIST-S1 product, which is dubbed the “transformer metric”. In Section 2.4, we detail the neural network architecture utilized for this transformer metric. Specifically, we detail the network’s structure, how the network is trained, and its use in estimating the probabilistic parameters required for the metric. In Section 2.5, we describe how transformer metric is used to delineate change and some calibration considerations. In Section 2.6, we provide an overall summary of how the disturbances are aggregated into a single tiled product. In Section 2.7, we introduce the mechanism we use for confirming disturbances in time through a given MGRS tile. In Section 2.8, we provide a summary of the layers and its expected formatting. The code and its documentation to generate the operational DIST-S1 product can be found at the `dist-s1` Github repository [13]. Specific details about the implementation can found in that repository.

2.1 Despeckling

In this section, we discuss how we despeckle each of the RTC-S1 products as a preprocessing step. We utilize total variation denoising from [14] implemented in [15]. Rather than following a strict model for the SAR speckle noise as in [16], we use the so called homomorphic approach in [17]. That is, we transform our imagery to decibels (dB) via

$$X \rightarrow 10 \log_{10}(X)$$

and denoise the image assuming approximately Gaussian noise. The optimization problem can be stated as follows. Let X_{noisy} be the original, speckled image in dB. We obtain

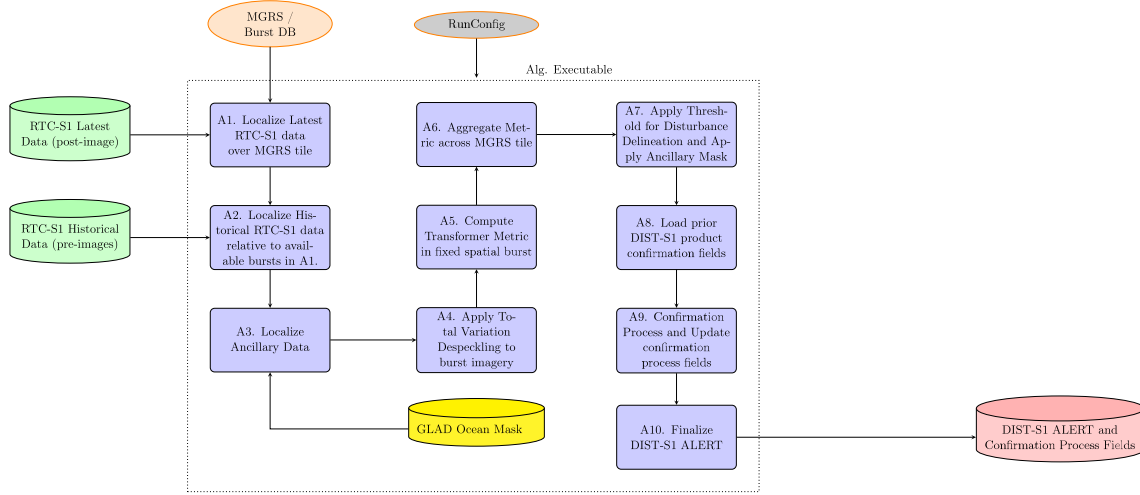


Figure 1: High-level workflow of the DIST-S1 algorithm including the ingestion of input datasets and the algorithm’s processing steps. This document is an explanation of this flow diagram. The output of the algorithm is the generation of the DIST-S1 ALERT product, which provides information about disturbance from the latest Sentinel-1 pass in three fields, and information about how consistently disturbances were observed over time in an additional seven confirmation process fields. There will be a configurable file for setting relevant algorithm parameters (which is called the *RunConfig* as is the case with other OPERA algorithms). Details of the ancillary data used is found in Table 1.

X_{denoised}

$$X_{\text{denoised}} := \operatorname{argmin}_X \lambda \|\nabla X\|_1 + \frac{1}{2} \|X - X_{\text{noisy}}\|_2^2$$

where $\lambda > 0$ and $\|\cdot\|_p$ is the p -norm. We set $\lambda = 5$ for VV and VH from ad-hoc visual inspection. After despeckling, we return the image back into its original scale γ^0 [18]. During total variation despeckling, we need all pixels to have valid numerical values. However, there are pixels with Not a Number (NaN) values and we need to fill these in to perform the optimization above. We have two cases for NaN pixels: 1. pixels within the valid image area (e.g. due to high incidence angle) and 2. pixels on the exterior of the image. For those within the image area that have NaN values, we interpolate the pixels within the boundary of the image using bilinear interpolation of nearby pixel backscatter values. For those on the exterior of the image we set pixels to be -25 db, which is beneath the expected noise floor in dB. During our workflow, we perform careful accounting to ensure the NaN values are preserved in the final product, i.e. we do not consider pixels as disturbed or undisturbed if they have NaN values in the input RTC-S1 product.

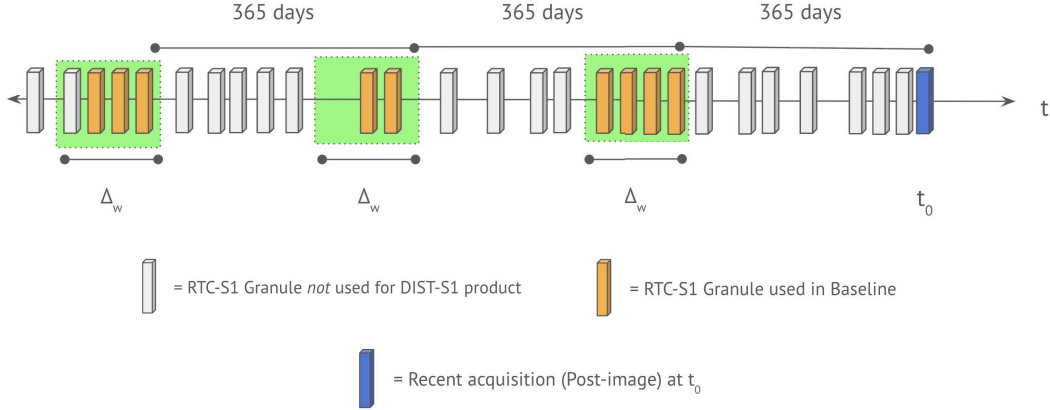


Figure 2: Illustration of baseline curation with a the search window size $\Delta_w = 60$ days. The blue squares are the pre-images and green square the post-image. Above there are a total of 9 images because in the second year there are no available images. We always have at most $n_c = 10$ total images in the baseline as dictated in Section 2.4.

2.2 Baseline Curation

In this section, we discuss how to curate the reference baseline that will be used to establish the per-pixel statistics by which we measure disturbance. Ultimately, we curate each burst time-series with respect to a given burst image or RTC-S1 product independently and the product is merged in a systematic way (see Section 2.6). This baseline curation is paramount to the type of changes being studied as this effectively constrains the sampling rate and the duration of changes that can be observed by the algorithm (see Section 1). We have curated our baseline to *align* the DIST-S1 product with the DIST-HLS products. Before introducing our baseline strategy we provide three important goals that will motivate the curation strategy.

1. The baseline should be from previous years so that we can accumulate the disturbance metric with each new acquisition.
2. Our baseline should capture phenological processes that are present in the latest acquisition.
3. Our baseline should be able to adapt to the irregularity of Sentinel-1 acquisition strategy.

We accomplish these goals by looking at the anniversary date of the latest acquisition over the last 3 years and collecting images up to 60 days prior (approximately 2 months) before each anniversary date. Specifically, we collect at most 10 images over this 3-year period, selecting at most 4 images from the most recent year, at most 3 from the year before that, and at most 3 from the year before that. This maximum aligns with the

context length n_c discussed in the previous section. By using this interannual strategy, each new acquisition has a rolling baseline and consistent changes can be tracked over time and we can accumulate the metric with each new acquisition. Further, restricting to 2 months prior to an anniversary date year-to-year allows us to capture the seasonal effects within a baseline e.g. leaf-on/leaf-off processes. Because our model can have a variable length baseline, regardless of what baseline images are available, we are still able to establish a baseline with at least one image. This approximates the curation strategy followed by DIST-HLS. The primary difference as noted in [4] is that we use 60 days prior to the anniversary date and DIST-HLS looks at ± 30 days around the anniversary date. We choose our strategy because our parameter estimates care about sequence and we wanted to have a baseline that represents the regular lead up to the current date. We call this window δ_w in days. See Figure 2 for an example of how this strategy works in practice. In this figure we have a baseline of 9 because there is one year when not enough Sentinel-1 data is available. The code used to curate the baseline can be found in our repository `dist-s1-enumerator` [19].

2.3 Model Description and the DIST-S1 Disturbance Metric

In this section, we detail the model and the metric used to assess per-pixel disturbance over a burst time-series. We refer to the model and metric simply as the “transformer model” and “transformer metric”, respectively. These names stem from the fact that our model and metric utilize a neural network leveraging a transformer architecture [12, 20], which is discussed in Section 2.4. The model and metric will provide a probabilistic tool to quantify how much the current observed post-image acquisition *deviates* from the reference pre-images. Additionally, the model and metric will be able to take into account spatiotemporal trends.

We now provide some notation to precisely describe the model and its associated metric. Consider a time-series of coregistered SAR imagery over a fixed spatial burst. Let I_k for $k = 1, \dots, n$ represent acquisitions times over a fixed spatial area such that I_1 is the earliest acquired image and I_n is the most recent. Specifically, let $\{I_1, \dots, I_n\}$ be the ordered set of pre-images each with spatial dimension $H \times W$. We assume that the temporal dimension is indexed by k so that $I_1, I_2, \dots, I_k, \dots, I_n$ represent sequential images. Let the post image I_{n+1} be of the same spatial dimension $H \times W$. We assume that all burst time-series are coming from a single orbit track. We will detail how to combine disturbance delineations across ascending and descending orbit tracks later in the document (Section 2.7). It is worth also noting that while this discussion assumes the baseline images $I_1, \dots, I_n$ are sequential, our baseline curation (Section 2.2) will use images around the anniversary date of the acquisition from the last 3 years. Our model’s architecture does not have explicit date encoding (i.e. month or day or year). We also found that the recency bias in predicting the per-pixel distribution provided greater interpretability, particularly with respect to hazards such as fires and landslides. Also, for this discussion, we consider single-channel

(i.e. single polarization) imagery, but the discussion will be extended to two-channel (dual-polarization) imagery later in this section. A metric m_{ij} is a per-pixel measurement of disturbance and can be seen as distance function between the set of pre-images and a post-image:

$$m_{ij} = m(\{I_1, \dots, I_n\}, I_{n+1})$$

We will typically assume $m_{ij} \geq 0$ and that larger values of m indicate a greater likelihood of disturbance. Now, let X be the random variable associated to the post-image I_{n+1} . We will approximate the per-pixel distribution of the post-image as normal, i.e. $X_{ij} \sim N(\mu_{ij}, \sigma_{ij})$ for $i = 1, \dots, H$ and $j = 1, \dots, W$. We do not know the underlying per-pixel distribution within the imagery, but we are able to approximate this distribution in a standard manner as we now discuss. Suppose that p is an unknown distribution of the scatterers in a pixel and q_θ is our known parametrized distribution, parametrized by θ , that we use to approximate p . To find the optimal approximation of p with respect to θ , one utilizes Kullback-Leibler Divergence $KL(p||q_\theta)$ with respect to distribution parameters θ in order to best approximate p ; in our case, θ is uniquely characterized by the parameters μ and σ . It is well known that KL minimization is equivalent to minimizing the Negative Log-Likelihood of q_θ with respect to θ using samples *drawn* from p [21], which is precisely what is done during training of the neural network model that estimates the parameters μ and σ (see Section 2.4). Thus, our selection of the normal distribution (the highest entropy continuous distribution) approximates the unknown distribution of image in a precise, systematic manner. Indeed, when SAR pixels are dominated by a single type of scatterer, normal approximations are used to heuristically identify disturbance [2, 3]. We will use our normal approximation model to identify outliers with respect to the distribution. A single channel backscatter SAR image (either VV or VH) encodes the amplitude of the returned signal [18, 22], whose values are restricted to the interval $(0, 1)$. Following [23], before estimating the normal parameters, we transform the range of observed values in γ^0 to $\mathbb{R}$ via the logit function:

$$\text{logit}(x) = \log\left(\frac{x}{1-x}\right).$$

that is a monotonically increasing bijection between the open interval $(0, 1)$ and $\mathbb{R}$. The transformation ensures the domain of the per-pixel distribution matches that of the normal distribution.

We reiterate that the image time series $I_1, \dots, I_n, I_{n+1}$ that we analyze (i.e. the set of reference pre-images and post-image) will come from single fixed burst [24]. This ensures the SAR acquisition geometry and terrain to be consistent through the time-series. This also removes the burden of onerous input curation that would be required to train such a model including providing information to the model regarding the SAR geometry. However, even operating under constraint that the metric only assesses disturbances within a single orbit track, we are still able to track disturbances for *all* orbit tracks that pass over a given DIST-S1 product tile. This process will be detailed in Section 2.7.

Given our per-pixel mean μ and standard deviation σ estimates, we are now ready to define our transformer metric for a single polarization. The transformer metric quantifies how much the post-image deviates from our modeled distribution. The transformer metric is simply the Mahalanobis distance d_{ij} of the post-image with respect to the estimated distribution [25]:

$$d_{ij} := \frac{|(I_{n+1})_{ij} - \mu_{ij}|}{\sigma_{ij}}$$

Observe that the transformer metric not only measures the absolute deviation from the estimated mean μ , but is inversely proportional to the observed variation σ . This ensures that areas that have highly variable pixels with respect to the reference pre-images, the absolute deviation is weighted accordingly. We will now drop the spatial indices ij unless their use is required for clarity. Let's consider the random variable:

$$Z := \frac{X - \mu}{\sigma}$$

and its absolute value $|Z|$. We note that $|Z|$ is the random variable modeling our single channel transformer metric d . It is well known that $P(|Z| \geq t)$ for $t = 1, 2, 3$ is bounded by .32, .05 and .01, respectively, using well-known properties of the normal distribution. In fact, we can find a formula for $|Z| > t$ for $t > 0$. First, note that

$$Z \sim N(0, 1)$$

where $N(0, 1)$ is standard normal distribution. Define:

$$p(t) := P(|Z| > t)$$

In the equation above, we highlight that Z is a random variable while $t > 0$ is a positive number. Note that $p(d)$ can be interpreted as “the probability that the random variable modeling our transformer metric $|Z|$ exceeds the *observed* metric d .” In fact, we can compute p using the closed-form solution for the cumulative distribution function of the standard normal distribution:

$$\begin{aligned} p(t) &= P(|Z| > t) \\ &= 1 - P(-t < Z < t) \\ &= 1 - [P(t < Z) - P(Z < -t)] \\ &= 1 - [P(t < Z) - (1 - P(t < Z))] && \text{Symmetry of } Z \text{ about } 0 \\ &= 2 - 2 \cdot P(t < Z) \\ &= 2 \cdot (1 - F(t)) \end{aligned} \tag{2}$$

where F is the cumulative distribution of the standard normal distribution. Rather than dealing with $p(d)$ whose value can become extremely small if d is large, we found it easier

to deal with the Mahalanobis distance directly. Therefore, our transformer metric d for a single polarization is precisely this Mahalanobis distance.

This transformer metric d for a single polarization image has the following interpretation: the transformer metric d has units of the estimated standard deviation σ and the metric measures the distance of the post-image to the mean in these units. For example, if d is found to be 3.1, then post-image is approximately 3.1 standard deviations from the mean. Even though the estimated mean and standard deviation vary from pixel-to-pixel, the transformer metric allows us to measure deviation in a systematic manner throughout an entire image using our probabilistic interpretation. In Figure 3, we illustrate how the data is processed to calculate the transformer metric. The example uses data over the Papua New Guinea Landslide occurring in May 2024 [26]. First, the figure shows how the mean μ and standard deviation σ images for each polarization are estimated from a reference set of pre-images using our transformer model. The figure then shows how the transformer metric is calculated using the post-image and its deviation from these estimates.

Although we will use a deep neural network with spatiotemporal encoding to estimate parameters μ and σ , these parameters can be estimated rather simply from the set of pre-images. Specifically,

$$\begin{aligned}\hat{\mu} &:= \frac{1}{n} \sum_{k=1}^{k=n} I_k \\ \hat{\sigma} &:= \frac{1}{n-1} \sum_{k=1}^{k=n} (I_k - \hat{\mu})^2\end{aligned}\tag{3}$$

Thus, we can compute the mahalanobis distance directly from these parameter estimates. However, such estimates ignore temporal ordering of the data as well as rich spatial patterns. In Appendix A, we discuss conventional disturbance metrics and contextualize them in terms of the probabilistic setting we have established here. We also discuss the Cumulative Sum metric and its relation to the transformer metric introduced here.

Up to this point, our discussion has focused on single channel (i.e. single polarization) image time-series; indeed, our estimates for mean and standard deviation are real numbers. We can combine the metrics from each polarization into a new single metric as follows. We define $d_{\max} := \max(d^{VV}, d^{VH})$, then this new $d_{\max}$ metric is the logical “or” of the two metrics d^{VV} and d^{VH} in that if the metric exceeds a value t , then *at least one* metric (either d^{VV} , d^{VH} or both) exceeds t . To contrast this, we can define $d_{\min}$ and this represents a logical “and” of the two metrics in the same way. For DIST-S1, we track changes that can be seen in *any* polarization, so we utilize the maximum. The maximum of d^{VV} and d^{VH} is the final “transformer metric”. We note that the discussion above holds for any co- and cross-polarization time-series including HH and HV. A summary of how the metric is computed is shown in Figure 4. Although we could have estimated the 2×2 covariance matrix for dual polarization imagery, frequently the cross-correlation values (i.e. the anti-

diagonal values of the covariance matrix) are small and numerically unstable. Therefore, we consider each channel separately and take a maximum of the Mahalanobis distances.

Our transformer metric design was adapted from Stephenson et al. [23], which utilizes a Recurrent Neural Nets (RNN) to estimate damage in S1 Coherence time-series via the Mahalanobis distance. We adapted this method to use the transformer architecture [20] which has become a highly scalable computer vision tool for spatiotemporal modeling [12, 27–29]. Indeed, vision transformers are emerging as a powerful tool within remote sensing [28, 30–33]. It is worth highlighting that in [23] a model is trained for each new stack of imagery. However, with the ability of transformers to robustly encode information, we elected to train a single global model that could be applied to any time series. In this work, we utilize a basic vision transformer architecture from the original paper [12] with an additional temporal encoding to provide our estimates with an understanding of spatiotemporal trends and variability within the baseline. This is discussed in Section 2.4.

Ancillary data	Spatial Resolution (m)	Static or Dynamic	Latency	Inputs	Geographic Scope
GLAD Ocean Mask [34]	30	Static	N/A	Landsat	Global

Table 1: Table of ancillary data to be used in the DIST-S1 workflow including its spatial resolution, latency, and geographic scope.

We reiterate that the DIST-S1 *transformer* model considers *temporal* trends within the reference pre-images. Specifically, the relative temporal positions of the inputs is a crucial part of spatiotemporal encoding of the transformer model and thus impacts the normal parameter estimates that inform the metric. We also note that the model was trained on image time-series coming from a fixed spatial burst. That is to say, the acquisition geometry and terrain remain fixed for each subsequent image in the time-series. While there are opportunities to train a similar model using time series spanning multiple tracks overlapping the same area, this was not pursued to ensure simplicity of the model and its training data. We anticipate models requiring inputs spanning both ascending and descending tracks would require some encoding of the incidence angle and greater care would have to be taken with nodata values as each track has different coverage. This was beyond the scope of this work. Although our transformer model is designed to detect disturbances coming from a single spatial burst, such information can still be combined across all bursts covering a given area in order to track *any* disturbances detected by the Sentinel-1 constellation. We discuss this process further in Section 2.7.

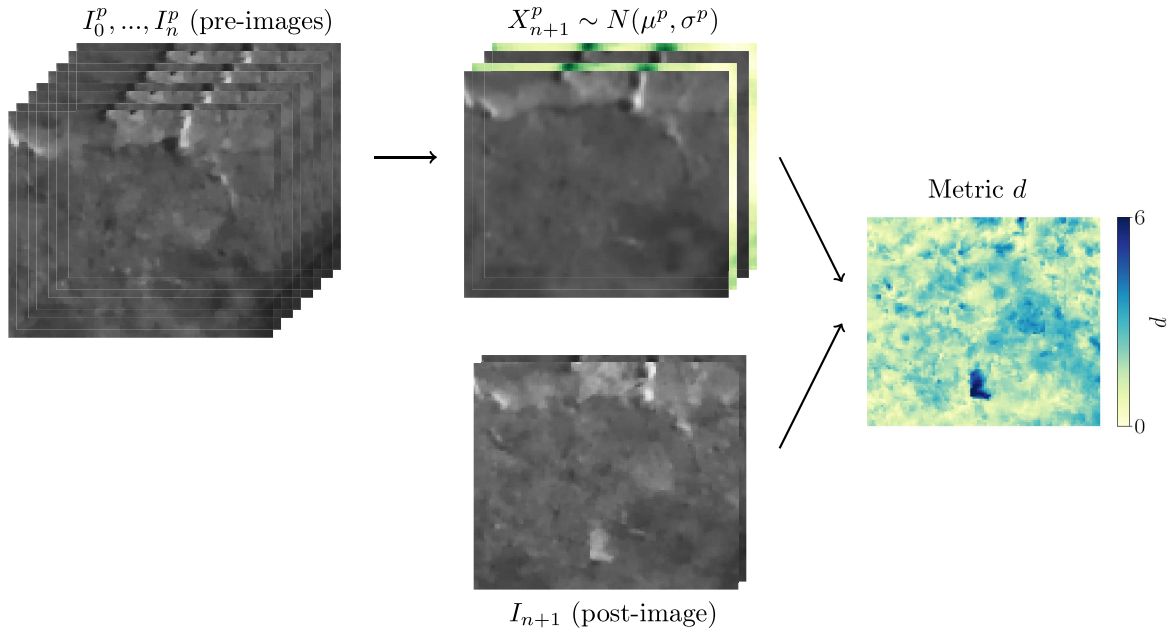


Figure 3: Above is a visual description of the metric computation required for the DIST-S1 algorithm utilizing a OPERA RTC-S1 time-series. The data is taken over the tragic Papua New Guinea landslide that occurred in May 2024 [26]. The landslide can be observed in the bottom-center of the image extent. In the upper-left panel, there is a dual-polarization sequence of pre-images before the landslide. The upper-center panel is the estimated mean μ and standard deviation σ of these pre-images, for each polarization. The bottom center panel shows the actual acquisition. In the rightmost panel, we show the transformer metric, which is the Mahalanobis distance of the the post-image acquisition from the mean μ .

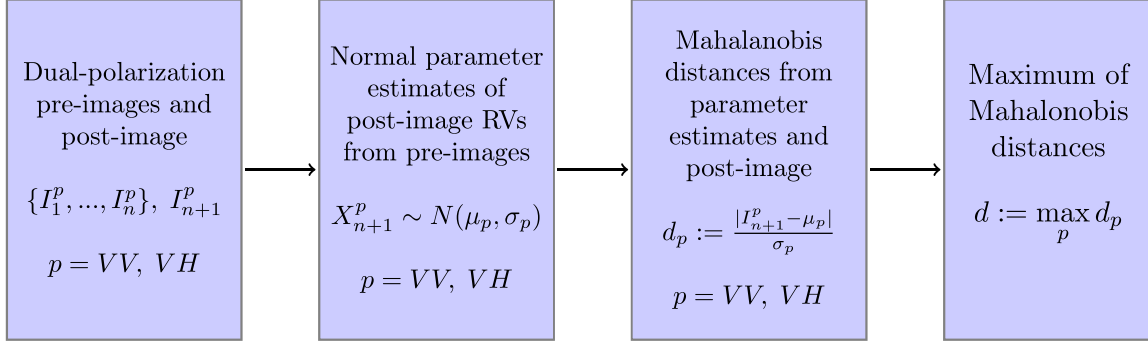


Figure 4: A summary of how the DIST-S1 metric is computed. The first arrow is where the deep-learning model is utilized to estimate the per-pixel distribution of the post-image. We note that this metric computation is applicable to both $VV + VH$ and $HH + HV$ data.

2.4 Neural Network Architecture, Training, and Inference

In this section, we describe the neural network used to estimate the normal parameters introduced in Section 2.3 that is used to compute the transformer metric. The training code, the model’s architecture, and additional details can be found in the Github repository `dist-s1-model` [35]. Indeed, neural networks have proven to be effective at delineating a wide-range of disturbances and land-use dynamics using SAR and optical input datasets [8, 23, 36–38]. The OPERA RTC-S1 is a relatively new SAR dataset and there does not exist labeled disturbance data associated with it. Moreover, *self-supervised* strategies have been shown to be extraordinarily effective in training deep transformers [27, 29, 39]. Very large transformer models (i.e. those with billions of parameters) trained in a self-supervised fashion are frequently called “foundation models” because such models can be fine-tuned on a small set of labeled data [30, 31, 33, 40, 41]. Many of the training techniques rely on masked pre-training following [27].

The transformer network used for the DIST-S1 algorithm is relatively small compared to those in industrial settings; the model has approximately 10^6 parameters. It directly adapts the architecture described in the original Vision Transformer architecture in [12]. We utilize a transformer encoder that utilizes an additional temporal positional embedding alongside the standard spatial positional embedding and the patch embedding [12]. There are more complex spatiotemporal embeddings considered in [28, 30, 31] so that the models can be applied to a variety of sensors at varying resolutions. We did not need such embeddings here because our input data is restricted to the OPERA RTC-S1. We now detail the self-supervised training strategy used for our model. The model estimates the per-pixel parameters μ and σ introduced in Section 2.3 as follows: from a reference set of pre-images, the network estimates these parameters and compares it to the next acquisition via the

negative log-likelihood. Let the neural network be represented via the function q_θ with parameters (or weights) θ . Our function q_θ has domain $\mathbb{R}^{H \times W} \times \dots \times \mathbb{R}^{H \times W}$ i.e. the function accepts a sequence of reference pre-images as inputs. Note that the sequence can be variable length up to the context length n_c , which is defined to be the maximum number of pre-images that can be supplied to q_θ . That is to say, we can use 1, 2, ..., n_c reference pre-images as input to q_θ . The context length is not a model choice, but a computational resource constraint. In our case, $n_c = 10$. As noted in Section 2.2, the 10 images span 3 different years (targeting at least 3 in each). The function $q_\theta(I_1, \dots, I_n)$ outputs (μ, σ) where μ and σ are of the same shape as I_k for $k = 1, \dots, n$. We use the post-image at time $n + 1$ to train the model via the negative-log-likelihood (*NLL*):

$$\begin{aligned} NLL(I_{n+1}|q_\theta(I_1, \dots, I_n)) &:= -\log P(I_{n+1}|\mu, \sigma) \\ &= \sum_{\substack{i=1, \dots, H \\ j=1, \dots, W}} \frac{1}{2} \log(2\pi\sigma_{ij}^2) + \frac{((I_{n+1})_{ij} - \mu_{ij})^2}{2\sigma_{ij}^2} \end{aligned} \quad (4)$$

where we obtain μ and σ from $q_\theta(I_1, \dots, I_n)$. We determine the model parameters (or weights) θ via standard stochastic optimization. We highlight that the loss in Eqn. 4 assumes that the post-image's pixels are independent. Of course, the independence assumption is false; but it allows us to compute the conditional pixel-wise probability $\log P(I_{n+1}|\mu, \sigma)$ of the entire post-image at once. However, each of the per-pixel parameters is computed using a 16×16 time-series from the supplied *pre-images*, which allows the rich spatiotemporal correlations to be utilized by the model. An even more detailed model description and training mechanism can be found in [42] and the architecture can be carefully inspected in our metric library [43]. As noted above, this training strategy is *self-supervised* as it entirely relies on the OPERA RTC-S1 dual-polarization burst time-series without labels. Each training sample can be thought of as a pair: a sequence of reference pre-images $\{I_1, \dots, I_n\}$ and post-image I_{n+1} . The model is trained on any subsequence of the pre-images of length 1, ..., n_c . The 106 bursts used for training cover 59 sites out of 300 DIST-HLS validations sites from [44, 45]. We selected the sites with the greatest number of temporal sampling in 2024, and whenever possible, tried to get an ascending and descending track from the same site. Due to Sentinel-1's selective processing of the imagery it collects, some of the sites, for example, have no available RTC-S1 data available in either ascending or descending geometry.

We now discuss the application of our model to a OPERA-RTC image. As we will discuss in Section 2.6, we apply our model to individual burst products. Even though bursts are relatively small ($\approx 1000 \times 3000$ pixels), we are constrained to apply our model on 16×16 windows or *chips* as in [12]. If we were to apply the model to contiguous chips, the pixels at the edge of each chip would contain artifacts as edge pixel estimates are more uncertain than those on the interior of the chip. Indeed, pixels in the center of the chip are able to utilize all of their spatially closest neighbors, whereas edge pixels are not. Thus

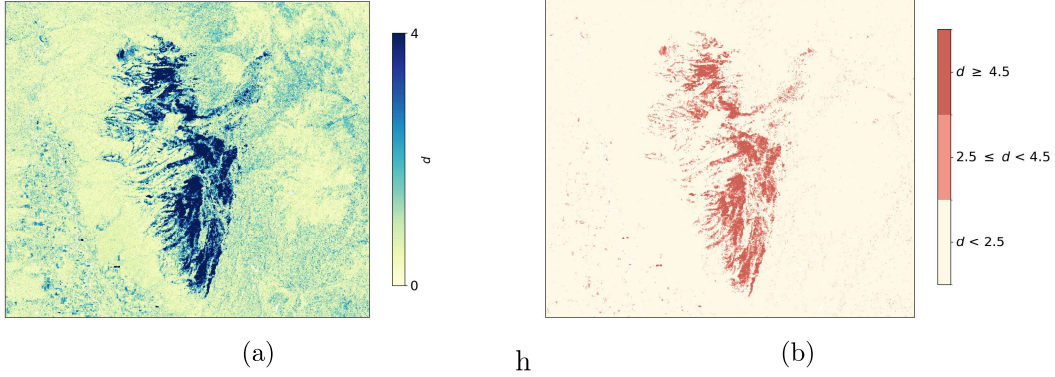


Figure 5: Above, the transformer metric and its associated disturbance map derived from RTC-S1 data to show the burn perimeter of the Park Fire in Butte, CA occurring in July 2024 [47]. The post-image used to compute the metric was acquired immediately after the fire. The two images illustrate above how disturbance is determined from the metric. The left panel (a) is the metric over the scene where the dynamic range goes from 0 to 4, i.e. we see where the distance from the estimated mean varies up to 4 standard deviations. The right panel (b) shows how the thresholds from Eqn. 6 generate a disturbance map.

the application of our model contains the well-known *receptive field* phenomena [46]. We mitigate this effect by applying the model to non-contiguous areas, i.e. applying the model to the chips with stride 7 and averaging the output over all such estimates. This is done in our open-source library [43]. The stride has important computation aspects in that if we have a stride of n and reduce by a factor k , then the computation time and memory goes up by a factor of k^2 . While there are moving window artifacts, the selection of the stride within the patch size of the window ensures that some of the model’s patch artifacts and the inference artifacts are mitigated.

2.5 Disturbance Delineation

In this section, we discuss how disturbance delineations are derived from our transformer metric. As noted in the previous sections, the metric is computed on a burst time-series with consistent acquisition geometry, that is for fixed orbit-track. Thus, the delineation in this section is focused on deriving a disturbance map in the scenario of a fixed spatial burst time-series. After developing this machinery, we will be able to describe how such disturbance maps can be aggregated across *all* bursts, even those on different orbit tracks (see Section 2.7).

A disturbance delineation is a discrete-valued image C_{ij} indicating where and how likely a disturbance occurred. Typically, a value of 0 indicates “no disturbance” and non-zero values indicate disturbance and/or a category of disturbance. In the simplest case, the

process of delineating a disturbance from an arbitrary metric m is the boolean operation:

$$C_{ij} := \begin{cases} 1 & \text{if } d_{ij} > t \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

where $t \in \mathbb{R}_{\geq 0}$. Here, a value of 1 indicates the pixel was disturbed and 0, un-disturbed. If the arbitrary metric m captures how likely a disturbance is in a given pixel with higher values indicating higher likelihood, then the process of delineation with a threshold t is the identification that disturbances are above a certain likelihood. Frequently, a threshold t is selected on an ad-hoc basis for a given time-series: an analyst inspects the per-pixel histogram of the metric m_{ij} and selects a threshold based on the percentile p so that $(100 - p)\%$ of pixels are marked as change. Given that disturbance is sparse in space and time, this is not an option for an operational product like DIST-S1. Indeed, it is not true that each new post-image should have a pre-determined *quantity* of change, if any at all. Fortunately, our metric has a probabilistic interpretation derived from a baseline of images and we can select a threshold that captures deviation relative to previously observed spatiotemporal variability with consistency in the probabilistic setting. Indeed, our thresholds too have a probabilistic interpretation: the higher the threshold, the greater the deviation a pixel must be from its baseline to be labeled as disturbed. We will demonstrate the wide applicability of the selected thresholds in Section 3. For now, we simply describe the definition of the relevant thresholds. Let d_{ij} be the transformer metric and C_{ij} a disturbance image with 0 indicating no disturbance, 1 indicating *moderate confidence* disturbance and 2 indicating *high confidence* disturbance.

$$C_{ij} := \begin{cases} 2 & \text{if } d_{ij} > 4.5 \\ 1 & \text{if } 4.5 \geq d_{ij} > 2.5 \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

These thresholds fashion the following probabilistic interpretation:

- A *moderate confidence* disturbance is between 2.5 and 4.5 deviations from the estimated mean in at least one of the polarizations.
- A *high confidence* disturbance is at least 4.5 deviations from the estimated means in at least one of the polarizations.

Rephrased, the higher the confidence of a disturbance, the greater the pixel labeled disturbed deviates from the estimated baseline distribution. These thresholds are meant to be fixed operationally as the transformer metric captures the spatiotemporal variability across land cover types and different environments in a systematic, probabilistically structured way. The training data was selected to be globally representative and across different phenological periods. As we shall see in Section 2.5.1, the thresholds suggested in Eqn. 6 provide helpful delineations of disturbance events such as floods, landslides, and wildfires across numerous environments.

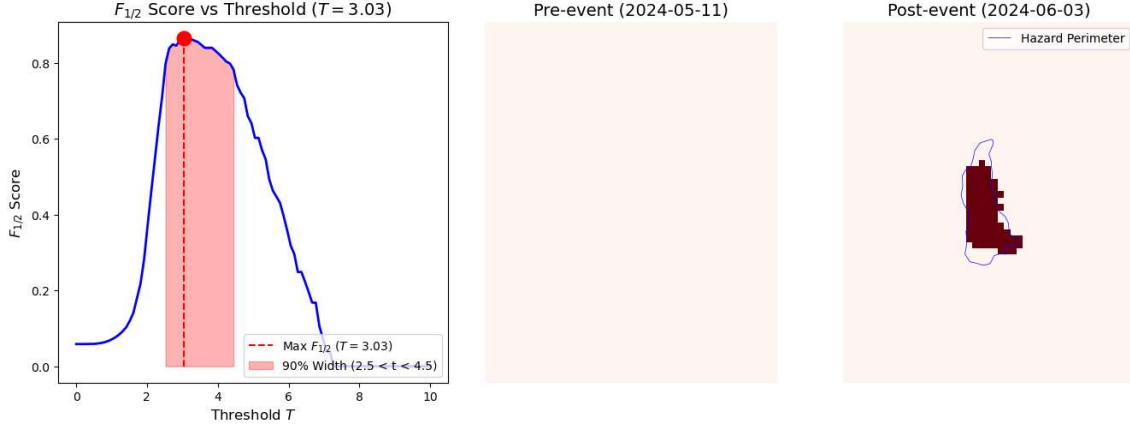


Figure 6: In the center and right panel we see the before and after metric imagery thresholded by the optimal threshold. The blue area in the right panel is the hand-drawn landslide extent derived from Planet Imagery (Credit: Alexander Handwerger.) The leftmost panel shows the results of optimizing the $F_{1/2}$ score for our disturbance metric and the 90% width. The optimal threshold T with respect to $F_{1/2}$ is 3.03.

We will supply the metric d_{ij} as a layer to users to ensure that more nuanced thresholding for a given scene or event can be performed, if desired. For example, if there is a significant rain event over a vegetated area, many of the pixels will appear disturbed and thresholds can be increased for such a given scene. It is worth noting that the confirmation process, which accumulates this metric over time to confirm changes over time ensures false positives are removed. This is discussed in more detail in Section 2.7.

2.5.1 Additional Disturbance Threshold Calibration

The process of *calibration* is the formal tuning of the algorithm and its parameters to provide optimal performance. Although our thresholds have a probabilistic interpretation, we did additional analysis to provide an analytical confirmation of these threshold choices. We evaluate accuracy using *calibration data*, that is, data that is independently generated. We curated hazard events delineated from high resolution optical datasets that were generated by Copernicus’s Emergency Management Service, National Interagency Fire Center’s Database, or handdrawn maps from planet imagery. See our `dist-s1-events` repository [48] for information about the provenance of these datasets. We consider a 1-parameter family of binary classifiers of the form:

$$\text{disturbed}_t := (d_{ij} > t) \quad (7)$$

where $t > 0$ for our metric. For this calibration exercise, we consider an image coming from a time prior to the hazard where the metric should have small values and a post-image

after the hazard whose disturbance delineation should be similar to the external calibration data. We can consider a threshold t that maximizes the $F_{1/2}$ score for all the available event datasets (see B). In Figure 6, we show a result for the Papua New Guinea Landslide in 2024. The $F_{1/2}$ score ensures that we prioritize precision over recall, because our baseline curation strategy means we will collect all disturbances in a calendar year. Specifically, there will be false positives from the point of view of our calibration exercise (i.e. areas that are not part of the hazard but are legitimate disturbances occurring over the last calendar year. Indeed, a hazard occurs over a much shorter time period: a landslide can happen in minutes, a flood over a few days, and a fire can spread over days, weeks, or even months. Moreover, the optically derived ground truth will also have different sensitivities than SAR and therefore there will be disturbances that SAR is not systematically able to detect and vice versa. For example, sparse short vegetation that gets burned in a wildfire is typically invisible to C-band SAR. Nonetheless, we can consider this calibration activity over 9 events to obtain a more formal view of the threshold selection. To obtain an upper and lower threshold, we consider the 90% width around the optimal threshold and for each upper and lower value we take a weighted average where the weight is determined by the optimal $F_{1/2}$ score. We see the results by event in Figure 7. Some events like Tuscany flood have very low optimal $F_{1/2}$ because the flood was not easily visible in the RTC-S1 pair. Ultimately, the results indicated a lower threshold of 2.48 and upper threshold of 4.28.

2.6 Aggregating Burst Disturbance into a Tiled Product

At this stage, we are ready to summarize our algorithm and understand how the various ingredients of the algorithm can be used to create a single tiled product. For a given MGRS tile and a new Sentinel-1 pass, we collect all the most recent RTC-S1 data from bursts overlapping the MGRS tile. For each burst product, we curate the reference baseline as in Section 2.2. Each burst will have a *different* time-series due to availability. For example, within a single MGRS tile, Sentinel-1 will turn off its imaging and therefore each RTC-S1 product will perform its own independent baseline curation. This flexibility allows us to map disturbances for all of the recent data regardless of spatiotemporal variability. After localizing all the necessary RTC-S1 data, we despeckle each image as noted in Section 2.1. We then compute the transformer metric for each burst time-series. We remark that this calculation burst-wise allows for two important operational considerations: (1) we are able to apply our transformer metric within memory and (2) the model trained on the RTC-S1 data product is being applied to this same product directly before any additional resampling. Once we have a metric for each burst from the most recent S1 pass, we aggregate the data into the MGRS tile, taking the average across overlapping pixels that are valid. This merged dataset is precisely **GEN-METRIC** (see Section 2.8). With our metric layer resampled to an MGRS tile, we delineate disturbance using the thresholds dictated in Section 2.5 to create the layer **GEN-DIST-STATUS-ACQ**. We apply an ocean mask that excludes all pixels that are at least 2 km from any land surface. To perform confirmation, we need the prior DIST-S1

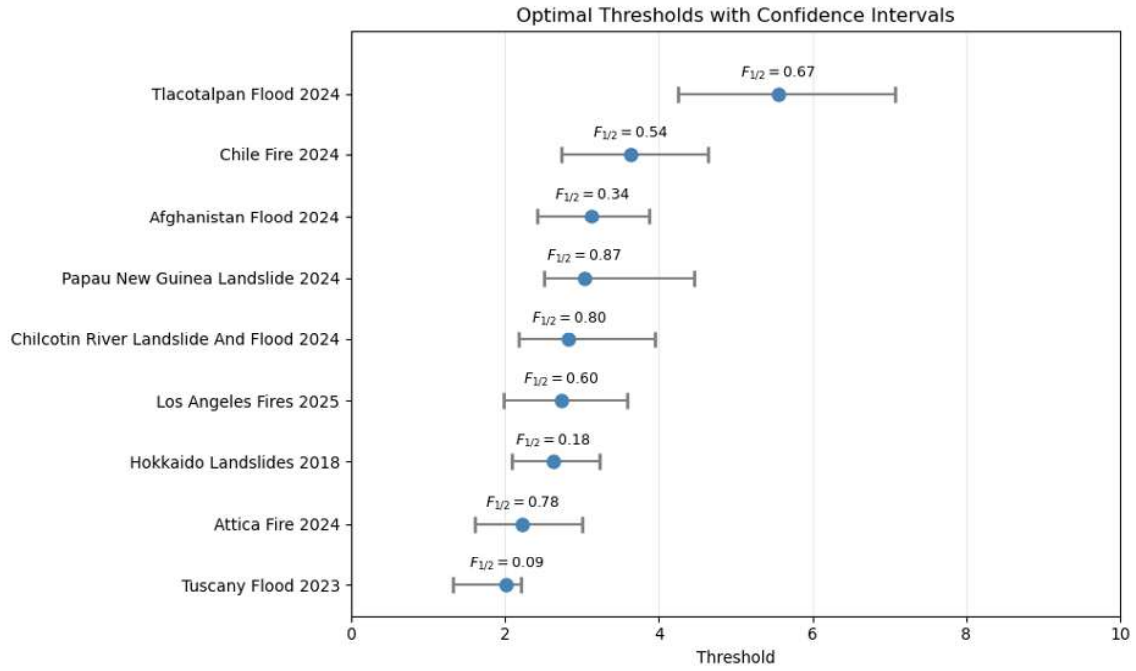


Figure 7: The results computing the 90% width of the $F_{1/2}$ score and the associated optimal $F_{1/2}$ score for 9 events considered. The Papua New Guinea event is shown in more detail in Figure 6.

product generated in this given tile and perform a confirmation process which we describe in the next section.

2.7 Disturbance Confirmation and Accumulation

The confirmation process takes the current disturbance delineation determined from our reference baseline and accumulates this change through time within the fixed spatial MGRS tile. It is worth noting that while our disturbance relies strictly on a baseline constructed from the same viewing geometry as the current acquisition, the confirmation process is done using the *MGRS tile time-series*. Thus, disturbances are accumulated and confirmed irrespective of the viewing geometry and we are able to increase the overall temporal sampling of the DIST-S1 product. The confirmation process for DIST-S1 is directly adapted from the DIST-HLS algorithm with minor adjustments to permit appropriate use of the Sentinel-1 temporal sampling and the immediate deviation measured by our metric. The simplest goal of the confirmation process is this: if a disturbance is detected in 3 consecutive observations, we want to *confirm* that disturbance. Pragmatically, we only want to look at the previous tiled product in a tile time-series to perform this confirmation. To accomplish this, we accumulate changes through various layers within the product. Most importantly, we utilize a confidence layer to track the accumulation of disturbance metric through the tile's time-series. We define the confirmation confidence C_{acc} at a fixed pixel by

$$C_{acc} = N_{\text{dist}}^2 \cdot \bar{d} \quad (8)$$

where N_{dist}^2 is the number of disturbed observations and $\bar{d}$ is the sample mean of the metric value across valid observations since a disturbance is first detected. This is precisely how confidence is defined in DIST-HLS. C_{acc} can be rewritten as a cumulative sum of the metric for disturbed pixels weighted by N_{dist} :

$$C_{acc} = |\{\text{Disturbances}\}| \sum_{i \in \{\text{Disturbances}\}} d^i \quad (9)$$

The weighting by $N_{\text{dist}} = |\{\text{Disturbances}\}|$ ensures the confidence is higher the more disturbed observations there are. To prevent a single, highly anomalous observation from dominating C_{acc} , we ensure that no metric contributes more than a value 10 to the cumulative sum, that is in the cumulative sum within C_{acc} we perform a ceiling operation:

$$\sum_{i \in \text{Disturbances}} \min(d^i, 10)$$

A disturbance is considered *confirmed* once C_{acc} exceeds this threshold t_C . This threshold t_C is selected so that three consecutive detections at the lowest disturbance threshold (2.5) triggers confirmation, i.e. $t_C = 3^2 \cdot 2.5 = 22.5$. To differentiate between high and low confidence disturbances, we simply track the *maximum* of the metric since the first detected

disturbance. If at any time the metric was above the high confidence alert threshold (4.5), the pixel is labeled as a high confidence confirmed pixel. We also have additional parameters that we set to ensure practical accumulation of changes and so that the confirmation logic can reset i.e. restarting the counter of N_{dist} and the related cumulative sum. If any of these parameters indicate that our cumulative confidence C_{acc} needs to be reset for a pixel that has already been confirmed, then we label that pixel as “finished”. We discuss these empirical parameters below:

- **Maximum elapsed time since last disturbance detection:** If the time since the last detected disturbance exceeds 30 days, then we restart cumulative calculation of C_{acc} .
- **Low Density Disturbance:** We track the per-pixel percentage of the disturbances with respect to the number of valid observations. If this density falls below 50%, we restart the cumulative calculation of C_{acc} .
- **Absolute cumulative no-disturbance observations:** If the total number of non-disturbed observations in the series exceeds 10, then we restart the cumulative calculation of C_{acc} .

For example, if in a given pixel, we detect a disturbance, and then record no disturbance in the following two observations, then the *low density disturbance* criterion will ensure we restart our calculation of C_{acc} . However, if a disturbance is visible in one viewing geometry but not another (say alternating disturbance and no-disturbance observations), then the low density disturbance criterion will allow us to confirm the pixel’s disturbance. The resulting disturbance state is captured in the GEN-DIST-STATUS field, which distinguishes between provisional, confirmed, and finished disturbances for both high and low-confidence detections. The full descriptions of the status labels for this layer are captured in table 3 in Section 2.8. An example series of products illustrating this confirmation process is shown in the Figure 8 showing the growth of the Palisades Fire in January 2025. We see how the fire grows and is confirmed and that an atmospheric river in February yields numerous disturbance alerts across the scene but is not confirmed. Table 2 shows an example disturbance series for calculating the accumulating confidence C_{acc} .

Metric at time i (d^i)	2.1	2.4	3.6	2.4	3.9	5.9	3.7	2.7	2.0
Cumulative Observed Disturbances (N_{dist})	0	0	1	1	2	3	4	5	0
Cumulative Confidence (C_{acc})	0	0	3.6	3.0	13.2	35.6	62.4	92.5	0
Disturbance Status	0	0	1	2	2	6	6	6	8

Table 2: Confirmation process operation across a single pixel’s time series (left to right). Confirmation confidence C_{acc} given by Eqn. 8. Disturbance status follows the code assignments in Table 3.

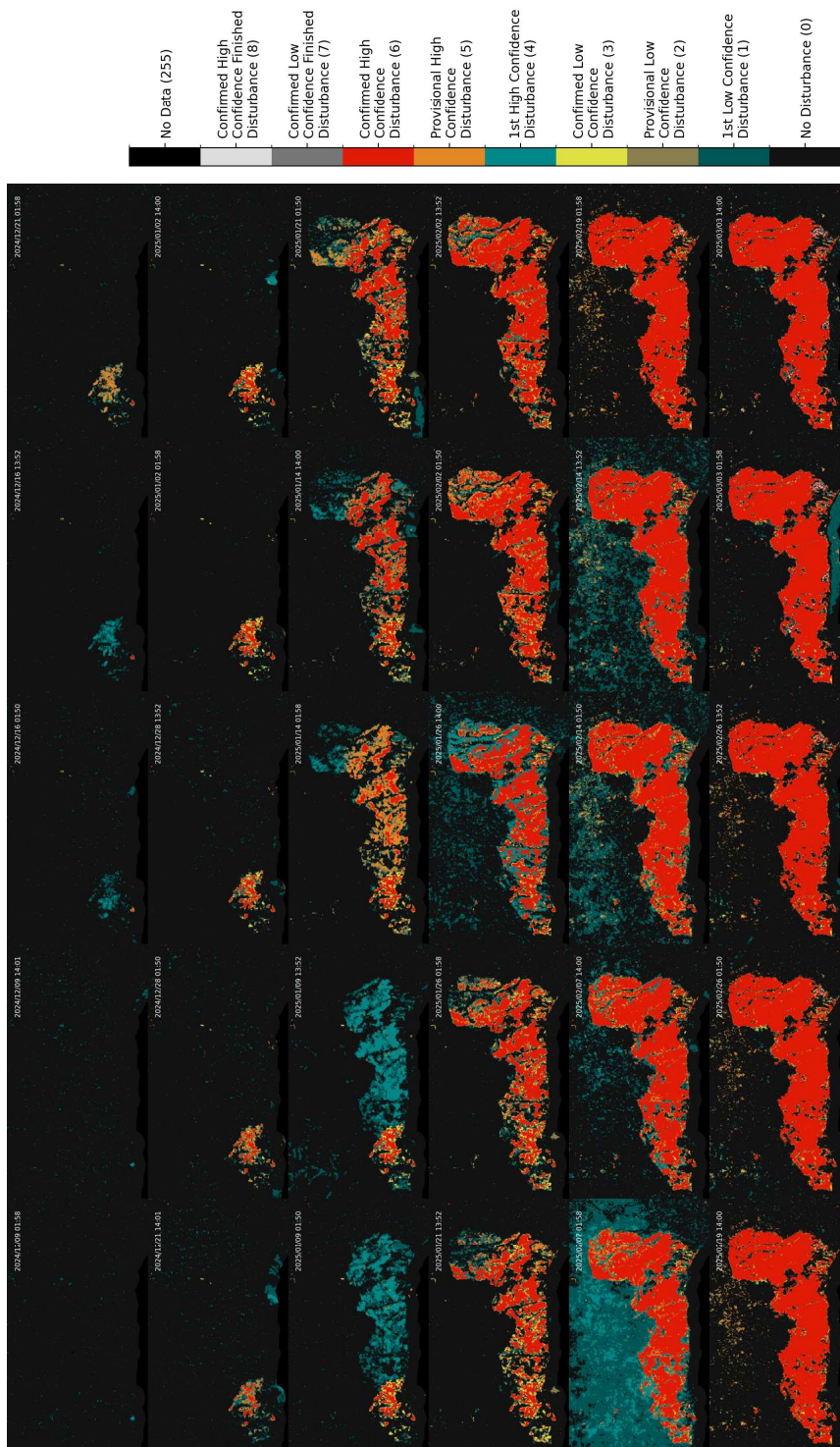


Figure 8: Time series of the GEN-DIST-STATUS layers of the DIST-S1 disturbance alerts covering the 2025 Palisades Fire (subset of MGRS tile 11SLT) for a 3-month period (December 2024 – March 2025). The Franklin River is visible in the December panels and the Palisades Fire in January. The atmospheric river in early February yields provisional alerts that are ultimately not confirmed.

2.8 The DIST-S1 Product: Formatting and Organization

In this section, we provide an overview of the DIST-S1 product’s output layers including some interpretation. Our DIST-S1 product will be distributed with respect to the Military Grid Reference System (MGRS) as shown in Figure 10 with 30 meter resolution and size 3600×3600 . Each tile spans approximately $109.8 \text{ km} \times 109.8 \text{ km}$. In Figure 10, the extents are $6^\circ \times 8^\circ$ tiles, which each contain hundreds of such tiles. A DIST-S1 product over a given MGRS tile will be coregistered with the DIST-HLS products to ensure analysis can be *harmonized*. Indeed, much of the DIST-S1’s algorithm design was adapted directly from DIST-HLS. Further, to make this alignment explicit, labels, color maps, and interpretation will be the same wherever possible.

A new DIST-S1 product over a given MGRS tile is *generated* with each new pass of Sentinel-1 data over that tile. In fact, each DIST-S1 product can be uniquely identified by its MGRS tile and the date and time of the latest Sentinel-1 pass over that tile. In Table 3, we show the various layers that constitute a DIST-S1 product and how disturbance and accumulated disturbances are captured within these layers. We show a sample product in Figure 9.

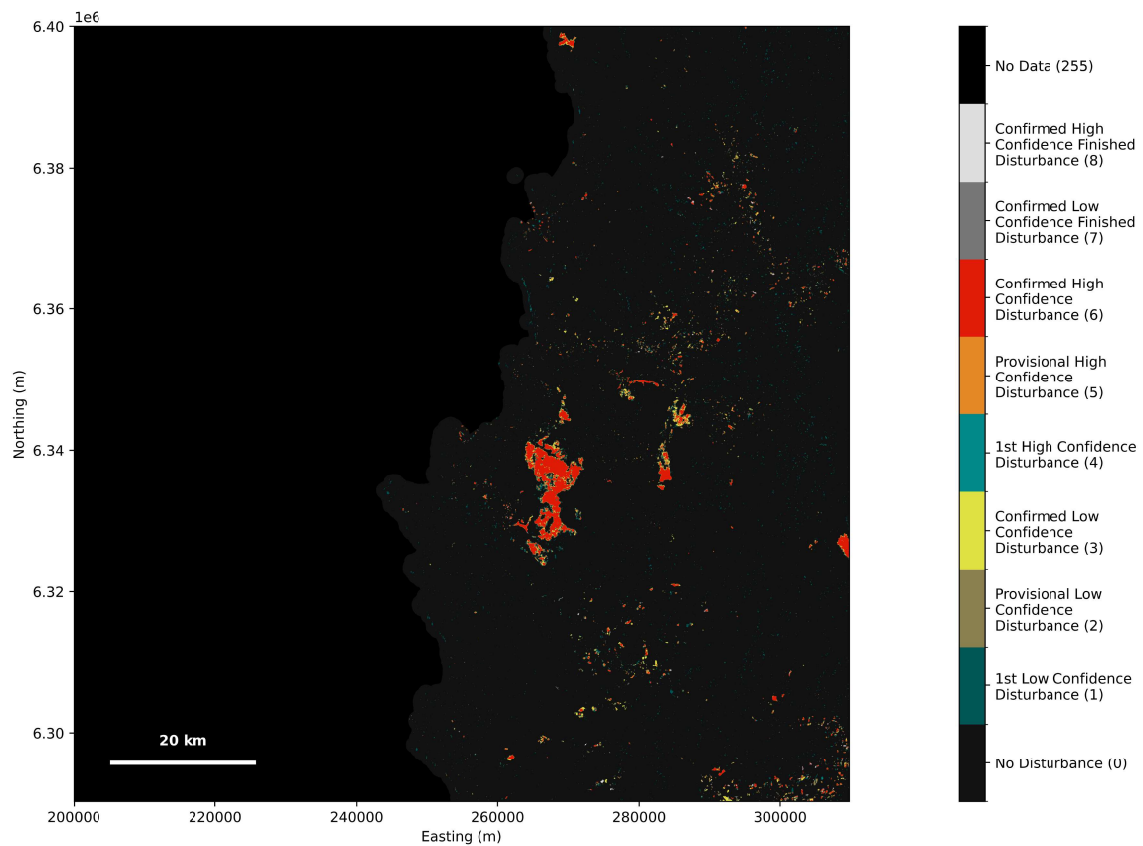


Figure 9: Sample DIST-S1 status layer showing confirmed disturbances capturing the Chile wildfire extent in February 2024. The product itself is derived from tile 19HBD and from the Sentinel-1 pass on April 2, 2024, 2 months after the start of the fire.

Table 3: The DIST-S1 layers: their names, descriptions, datatypes, and possible values.

DIST-S1 ALERT Raster Layer	Description	Layer name	Data type	Layer values
<i>Disturbance Status</i>	Status of generic SAR disturbance measured via the transformer metric. The discrete value in this layer indicates <i>moderate</i> and <i>high</i> disturbance labels, indicating the confidence of disturbance. Additionally, there are confirmation labels: <i>first</i> (1 observation), <i>provisional</i> (2+ consecutive observations) and <i>confirmed</i> (cumulative confirmation confidence above threshold - threshold set to confirm after 3 consecutive disturbance detections). The magnitude of confirmed observations is determined by the highest observed change over consecutive observations.	GEN-DIST-STATUS	UInt8	0: No Disturbance 1: First Moderate Disturbance 2: Provisional Moderate Disturbance 3: Confirmed Moderate Disturbance 4: First High Disturbance 5: Provisional High Disturbance 6: Confirmed High Disturbance 7: Confirmed Low Finished 8: Confirmed High Finished 255: No Data
<i>Disturbance Status Lat- est</i>	Indication of generic SAR disturbance measured via the transformer metric from just the latest pass of S1 data	GEN-DIST-STATUS-ACQ	UInt8	Same as above, but only uses [0,1,4]

Table 3 – continued from previous page

DIST-S1 ALERT Raster Layer	Description	Layer name	Data type	Layer values
<i>Current Metric Anomaly</i>	The transformer metric computed from the current acquisition and the historical baseline images as defined by the current window strategy. Layer can be interpreted as the confidence a pixel is disturbed.	GEN-METRIC	Float32	$[0, \infty)$. Smaller values indicate less disturbance and higher values indicate more.
<i>Maximum Metric Anomaly</i>	The maximum value of the transformer metric that has occurred over a disturbance series. Used by the DIST-HLS confirmation process.	GEN-METRIC-MAX	Float32	$[0, \infty)$. Smaller values indicate less disturbance and higher values indicate more.
<i>Generic Distur- bance Confidence</i>	Mean metric value since initial disturbance detection times the number of metric values above a threshold squared, until the anniversary date is reached, or a fixed number of consecutive non-disturbances are observed. Value converted to integer form. Same definition as used by the DIST-HLS confirmation process.	GEN-DIST-CONF	Int16	-1 : No Data 0 : No Disturbance >0 : Disturbance Confidence
<i>Date of First Dis- turbance</i>	The day of first detection of the current disturbance series specified as the number of days since December 31, 2020.	GEN-DIST-DATE	Int16	-1 : No Data 0 : No Disturbance >0 : Day of first disturbance

Table 3 – continued from previous page

DIST-S1 ALERT Raster Layer	Description	Layer name	Data type	Layer values
<i>Number of Disturbances</i>	The number of disturbance detections within an on-going disturbance series. Maximum of 254.	GEN-DIST-COUNT	UInt8	0 : No Disturbance [1-254] : Number of Disturbance Detections 255 : No Data
<i>Percentage of Disturbances</i>	The percentage of disturbances within the current disturbance series.	GEN-DIST-PERC	Float32	[0, 1.0]. Smaller values indicate fewer disturbances and higher values indicate more.
<i>Disturbance Duration</i>	Number of days of on-going disturbance detection starting from the first day of detection (inclusive) and increasing up to one year (366 days).	GEN-DIST-DUR	Int16	-1 : No Data 0 : No Disturbance [1-366] : Number of days from first to most recent disturbance detection
<i>Date of Last Observation Assessed</i>	The day of the last observation assessed for disturbance for land pixels specified as the number of days since December 31, 2020.	GEN-LAST-DATE	Int16	-1 : No Data ≥ 0 : Last Day Assessed

3 Remarks on Validation

This section provides a high-level overview of the selected calibration of the DIST-S1 algorithm.

Many of the parameters are calibrated empirically. For example, we selected confirmation parameters based on desired outcome of the confirmation process given our knowledge about Sentinel-1 temporal sampling. The related process of *validation* is comparing the output product of the tuned algorithm and comparing to an independently generated dataset to assess the accuracy of the generated product. The validation of these products will be covered in a forthcoming journal article similar to the article that covers the validation of DIST-HLS [50]. We will call the independently labeled data *validation data* or

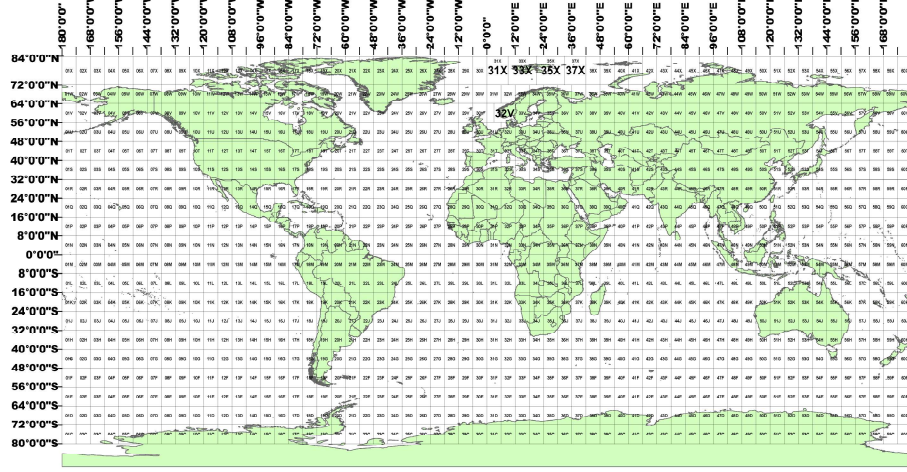


Figure 10: MGRS tiling system from [49]. Shown are $6^\circ \times 8^\circ$ aggregated tile extents for visualization; there are hundreds of MGRS tiles subdividing each of these aggregated tiles.

calibration data. We are frequently required to assess the performance of the algorithm curating optically-derived disturbances as optical datasets can be more easily understood by an analyst. However, calibrating and validating the DIST-S1 SAR-derived outputs against optically-derived data presents important obstacles: different sensor sensitivities yield different disturbance delineations. Additionally, the interannual baseline curation of the DIST-S1 product makes it harder to compare against hazard datasets where often the delineations are based on an image before and after the hazard. In contrast, the DIST-S1 product invariably accumulates all disturbances over the entire calendar year. We therefore balance empirical observations of the DIST-S1 product parameters with formal calibration activities. Indeed, as we saw in Section 2.5, there are probabilistic interpretations of the disturbance thresholds.

3.1 Requirements for DIST-S1 Accuracy

The OPERA project DIST-S1 requirement is stated below:

The disturbance (DIST-S1) products, derived from RTC-S1 data, shall identify pixels having at least 50% vegetative structure loss per RTC pixel (approximately 30m in size) between a current single-date RTC observation as compared to a set of historical RTC observations. The Disturbance Alert (DIST-ALERT-S1) product shall measure said disturbances with 80% accuracy per validation pixel for at least 80% of all validation products considered.

A validation approach developed by the Global Land Analysis and Discovery (GLAD) group at the University of Maryland (UMD) for their DIST-HLS product was expanded and adapted to evaluating the performance of the DIST-S1 algorithm. This evaluation has shown that the operational DIST-S1 algorithm meets the above total accuracy requirement over a stratified set of test sites as measured against human generated reference results. The data, results, and their analysis will be discussed in a forthcoming journal article. The OPERA project has ensured these products are passing.

4 Data Access

Data Access Input Data

- The primary input data to generate the DIST-S1 products, are the RTC products which are available through the National Aeronautics and Space Administration (NASA) Alaska Satellite Facility (ASF)-Distributed Active Archive Center (DAAC).

Data Access Output Data:

- DIST-S1 products, will be made available through the National Aeronautics and Space Administration (NASA) Alaska Satellite Facility (ASF)-Distributed Active Archive Center (DAAC).

Data Access Related URL: <https://search.earthdata.nasa.gov/search/granules?p=C4090131664-ASF>

5 Acronyms

γ^0	Gamma-naught (the scale of the OPERA RTC-S1) products
ADT	Algorithm Development Team
AUC	Area under curve metric
ALOS	Advanced Land Observing Satellite
ASF	Alaska Satellite Facility
CuSum	Cumulative Sum Change Detection
DAAC	Distributed Active Archive Center
dB	decibels
DEM	Digital Elevation Model
DIST-S1	OPERA's Sentinel-1 Disturbance product
DIST-HLS	OPERA's HLS Disturbance product
ESA	European Space Agency
F_β	β -weighted harmonic mean of precision and recall. See Appendix B
GeoTiff	Geographic Tagged Image File Format

GPU	Graphical Processing Unit
HDX	Humanitarian Data eXchange
InSAR	Interferometric SAR
IW	Interferometric Wide (S1-A/B mode)
JAXA	Japan Aerospace Exploration Agency
JPL	Jet Propulsion Laboratory
KML	Keyhole Markup Language
L1	Level-1 product
L2	Level-2 product
NaN	Not a Number
NASA	National Aeronautics and Space Administration
NISAR	NASA-ISRO SAR
NLL	Negative Log Likelihood
OPERA	Operational Product for End-users from Remote sensing Analysis
RADD	RAdar for Detecting Deforestation
RNN	Recurrent Neural Net
RTC-S1	Radiometric and Terrain Corrected Sentinel-1 Product
RV	Random Variables
S1	Sentinel-1
S1-A/B	Sentinel-1 A/B
SAR	Synthetic Aperture Radar
ScanSAR	Scanning Synthetic Aperture Radar
SDS	Science Data System
SLC	Single-Look Complex
UNOSAT	United Nations Satellite Centre
ViT	Vision Transformer
VH	Vertical-Horizontal Polarization
VHR	Very High (spatial) Resolution
VV	Vertical-Vertical Polarization

6 Acknowledgements

This research was conducted at the Jet Propulsion Laboratory, California Institute of Technology, under contract with the National Aeronautics and Space Administration (80NM0018D0004). The original Copernicus Sentinel data used for this paper have been provided by the European Space Agency. The work to develop the deep learning model was originally done by Harris Hardiman-Mostow under JIFRESSE summer internship under supervision from Charlie Marshak and Alexander Handwerger. Diego Martinez helped organize and clean up the research code into `dist-s1-model` [35]. Diego Martinez and Jake Lee helped optimize the model for production use including reducing the total number of parameters and

reducing the stride used in inference.

References

- [1] J. Reiche, A. Mullissa, B. Slagter, Y. Gou, N.-E. Tsendbazar, C. Odongo-Braun, A. Vollrath, M. J. Weisse, F. Stolle, A. Pickens, *et al.*, “Forest disturbance alerts for the congo basin using sentinel-1,” *Environmental Research Letters*, vol. 16, no. 2, p. 024005, 2021.
- [2] M. Watanabe, C. N. Koyama, M. Hayashi, I. Nagatani, T. Tadono, and M. Shimada, “Refined algorithm for forest early warning system with alos-2/palsar-2 scansar data in tropical forest regions,” *Remote Sensing of Environment*, vol. 265, p. 112643, 2021.
- [3] C. N. Koyama, M. Watanabe, M. Hayashi, T. Ogawa, and M. Shimada, “Mapping the spatial-temporal variability of tropical forests by alos-2 l-band sar big data analysis,” *Remote Sensing of Environment*, vol. 233, p. 111372, 2019.
- [4] M. Hansen, A. Pickens, and Z. Song, “Opera dist product - algorithm theoretical basis document.” https://lpdaac.usgs.gov/documents/1835/OPERA_DIST_ATBD_V1.pdf, 2024.
- [5] E. J. Rignot and J. J. Van Zyl, “Change detection techniques for ers-1 sar data,” *IEEE Transactions on Geoscience and Remote sensing*, vol. 31, no. 4, pp. 896–906, 1993.
- [6] J. Reiche, R. Verhoeven, J. Verbesselt, E. Hamunyela, N. Wielaard, and M. Herold, “Characterizing tropical forest cover loss using dense sentinel-1 data and active fire alerts,” *Remote Sensing*, vol. 10, no. 5, 2018.
- [7] J. Reiche, S. De Bruin, D. Hoekman, J. Verbesselt, and M. Herold, “A bayesian approach to combine landsat and alos palsar time series for near real-time deforestation detection,” *Remote Sensing*, vol. 7, no. 5, pp. 4973–4996, 2015.
- [8] H. Chen and Z. Shi, “A spatial-temporal attention-based method and a new dataset for remote sensing image change detection,” *Remote Sensing*, vol. 12, no. 10, 2020.
- [9] S. Ji, S. Wei, and M. Lu, “Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 1, pp. 574–586, 2019.
- [10] D. Bonafilia, B. Tellman, T. Anderson, and E. Issenberg, “Sen1floods11: a georeferenced dataset to train and test deep learning flood algorithms for sentinel-1,” in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 835–845, 2020.

- [11] C. Rambour, N. Audebert, E. Koeniguer, B. Le Saux, M. Crucianu, and M. Datcu, “Sen12-flood: a sar and multispectral dataset for flood detection,” 2020.
- [12] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, “An image is worth 16x16 words: Transformers for image recognition at scale,” *CoRR*, vol. abs/2010.11929, 2020.
- [13] OPERA ADT Team, “**dist-s1** – GitHub repository.” <https://github.com/OPERA-Cal-Val/dist-s1>, 2024. [Online; accessed 16-Oct-2026].
- [14] T. Goldstein and S. Osher, “The split bregman method for l1-regularized problems,” *SIAM journal on imaging sciences*, vol. 2, no. 2, pp. 323–343, 2009.
- [15] S. Van der Walt, J. L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J. D. Warner, N. Yager, E. Goullart, and T. Yu, “scikit-image: image processing in python,” *PeerJ*, vol. 2, p. e453, 2014.
- [16] J. M. Bioucas-Dias and M. A. T. Figueiredo, “Total variation restoration of speckled images using a split-bregman algorithm,” in *2009 16th IEEE International Conference on Image Processing (ICIP)*, pp. 3717–3720, 2009.
- [17] C.-A. Deledalle, L. Denis, S. Tabti, and F. Tupin, “Mulog, or how to apply gaussian denoisers to multi-channel sar speckle reduction?,” *IEEE Transactions on Image Processing*, vol. 26, no. 9, pp. 4389–4403, 2017.
- [18] G. H. X. Shiroma, M. Lavalley, and S. M. Buckley, “An area-based projection algorithm for sar radiometric terrain correction and geocoding,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–23, 2022.
- [19] OPERA ADT/PST Team, “**dist-s1-enumerator** – GitHub repository.” <https://github.com/OPERA-Cal-Val/dist-s1-enumerator>, 2024. [Online; accessed 16-Oct-2026].
- [20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Advances in Neural Information Processing Systems* (I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, eds.), vol. 30, Curran Associates, Inc., 2017.
- [21] C. M. Bishop and H. Bishop, *Deep learning: Foundations and concepts*. Springer Nature, 2023.
- [22] D. Small, “Flattening gamma: Radiometric terrain correction for sar imagery,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 8, pp. 3081–3093, 2011.

- [23] O. L. Stephenson, T. Köhne, E. Zhan, B. E. Cahill, S.-H. Yun, Z. E. Ross, and M. Simons, “Deep Learning-Based Damage Mapping with InSAR Coherence Time Series,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–17, 2021.
- [24] OPERA Advanced Data Technologies, “burst_db – GitHub repository.” https://github.com/opera-adt/burst_db, 2024. [Online; accessed 16-Oct-2024].
- [25] K. P. Murphy, *Machine learning: a probabilistic perspective*. MIT press, 2012.
- [26] Wikipedia contributors, “2024 Enga landslide – Wikipedia, The Free Encyclopedia.” https://en.wikipedia.org/wiki/2024_Enga_landslide, 2024. [Online; accessed 16-Oct-2024].
- [27] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, “Masked autoencoders are scalable vision learners,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16000–16009, June 2022.
- [28] C. J. Reed, R. Gupta, S. Li, S. Brockman, C. Funk, B. Clipp, K. Keutzer, S. Candido, M. Uyttendaele, and T. Darrell, “Scale-mae: A scale-aware masked autoencoder for multiscale geospatial representation learning,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4088–4099, 2023.
- [29] C. Feichtenhofer, h. fan, Y. Li, and K. He, “Masked autoencoders as spatiotemporal learners,” in *Advances in Neural Information Processing Systems* (S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, eds.), vol. 35, pp. 35946–35958, Curran Associates, Inc., 2022.
- [30] J. Jakubik, S. Roy, C. Phillips, P. Fraccaro, D. Godwin, B. Zadrozny, D. Szwarcman, C. Gomes, G. Nyirjesy, B. Edwards, *et al.*, “Foundation Models for Generalist Geospatial Artificial Intelligence,” *arXiv preprint arXiv:2310.18660*, 2023.
- [31] G. Tseng, I. Zvonkov, M. Purohit, D. Rolnick, and H. Kerner, “Lightweight, Pre-trained Transformers for Remote Sensing Timeseries,” *arXiv preprint arXiv:2304.14065*, 2023.
- [32] A. Lacoste, N. Lehmann, P. Rodriguez, E. Sherwin, H. Kerner, B. Lütjens, J. Irvin, D. Dao, H. Alemohammad, A. Drouin, *et al.*, “Geo-bench: Toward foundation models for earth monitoring,” *Advances in Neural Information Processing Systems*, vol. 36, 2024.
- [33] Clay Foundation and Development Seed LLC., “Model repository.” <https://github.com/Clay-foundation/model>, 2024. Accessed: 2024-10-11.
- [34] OPERA, “Glad ocean mask.” <https://console.cloud.google.com/storage/browser/earthenginepartners-hansen/OceanMask>, 2026. Accessed: 2026-03-16.

- [35] OPERA ADT/PST Team, “`dist-s1-model` – GitHub repository.” <https://github.com/OPERA-Cal-Val/dist-s1-model>, 2024. [Online; accessed 16-Oct-2026].
- [36] C. F. Brown, S. P. Brumby, B. Guzder-Williams, T. Birch, S. B. Hyde, J. Mazzariello, W. Czerwinski, V. J. Pasquarella, R. Haertel, S. Ilyushchenko, K. Schwehr, M. Weisse, F. Stolle, C. Hanson, O. Guinan, R. Moore, and A. M. Tait, “Dynamic world, near real-time global 10 m land use land cover mapping,” *Scientific Data*, vol. 9, no. 1, p. 251, 2022.
- [37] L. Mou, L. Bruzzone, and X. X. Zhu, “Learning spectral-spatial-temporal features via a recurrent convolutional neural network for change detection in multispectral imagery,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 2, pp. 924–935, 2018.
- [38] Z. Zheng, A. Ma, L. Zhang, and Y. Zhong, “Change is everywhere: Single-temporal supervised object change detection in remote sensing imagery,” *CoRR*, vol. abs/2108.07002, 2021.
- [39] C. Feichtenhofer, Y. Li, K. He, *et al.*, “Masked autoencoders as spatiotemporal learners,” *Advances in neural information processing systems*, vol. 35, pp. 35946–35958, 2022.
- [40] J. Irvin, L. Tao, J. Zhou, Y. Ma, L. Nashold, B. Liu, and A. Y. Ng, “Usat: A unified self-supervised encoder for multi-sensor satellite imagery,” *arXiv preprint arXiv:2312.02199*, vol. 1, no. 2, p. 3, 2023.
- [41] M. J. Allen, F. Dorr, J. A. G. Mejia, L. Martinez-Ferrer, A. Jungbluth, F. Kalaitzis, and R. Ramos-Pollan, “M3leo: A multi-modal, multi-label earth observation dataset integrating interferometric sar and rgb data,” 2024.
- [42] H. Hardiman-Mostow, C. Marshak, and A. L. Handwerger, “Deep Self-Supervised Disturbance Mapping With the OPERA Sentinel-1 Radiometric Terrain Corrected SAR Backscatter Product,” *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 18, pp. 10254–10271, 2025.
- [43] C. Marshak, R. West, T. Oliver-Cabrera, J. Jung, and H. Hardiman-Mostow, “dist-metrics.” <https://github.com/distmetrics>, 2024. Accessed: 2024-11-01.
- [44] OPERA Cal/Val Team, “Dist validation repository.” <https://github.com/OPERA-Cal-Val/DIST-Validation/tree/main>, 2024. Accessed: 2024-10-11.
- [45] M. G. Bato, D. Bekaert, B. Chapman, M. Hansen, J. W. Jones, J. W. Kim, and Others, “OPERA Product Validation Document v1.5.” <https://t.ly/CrUq>, 2023.

- [46] W. Luo, Y. Li, R. Urtasun, and R. Zemel, “Understanding the effective receptive field in deep convolutional neural networks,” *Advances in neural information processing systems*, vol. 29, 2016.
- [47] Wikipedia contributors, “Park Fire – Wikipedia, The Free Encyclopedia.” https://en.wikipedia.org/wiki/Park_Fire, 2024. [Online; accessed 16-Oct-2024].
- [48] OPERA ADT/PST Team, “dist-s1-events – GitHub repository.” <https://github.com/OPERA-Cal-Val/dist-s1-events>, 2024. [Online; accessed 16-Oct-2026].
- [49] NASA HLS Team, “HLS Tiling System Description.” <https://hls.gsfc.nasa.gov/products-description/tiling-system/>, 2024. Accessed: 2024-10-15.
- [50] A. H. Pickens, M. C. Hansen, Z. Song, A. Poulson, A. Komarova, A. Baggett, T. Kerr, A. Mikus, C. Ortiz Dominguez, A. Tyukavina, *et al.*, “Rapid monitoring of global land change,” *Nature Communications*, vol. 16, no. 1, p. 8948, 2025.
- [51] C.-A. Deledalle, L. Denis, and F. Tupin, “How to compare noisy patches? patch similarity beyond gaussian noise,” *International Journal of Computer Vision*, vol. 99, no. 1, pp. 86–102, 2012.
- [52] M. J. Canty, *Image analysis, classification and change detection in remote sensing: with algorithms for Python*. Crc Press, 2019.
- [53] NISAR Science Team, *NASA-ISRO SAR (NISAR) Mission Science Users’ Handbook*. NASA Jet Propulsion Laboratory, 2018. 261 pp.
- [54] M. Shimada, T. Itoh, T. Motooka, M. Watanabe, T. Shiraishi, R. Thapa, and R. Lucas, “New global forest/non-forest maps from alos palsar data (2007–2010),” *Remote Sensing of Environment*, vol. 155, pp. 13–31, 2014.
- [55] E. Page, “Cumulative sum charts,” *Technometrics*, vol. 3, no. 1, pp. 1–9, 1961.
- [56] NISAR Science Algorithms Team, “Forest disturbance product generation.” https://gitlab.com/nisar-science-algorithms/ecosystems/forest_disturbance/ProductGeneration, 2022. Accessed: 2024-10-11.
- [57] S. Seitz, “Probabilistic CuSum for Change Point Detection.” <https://sarem-seitz.com/posts/probabilistic-cusum-for-change-point-detection.html>, 2022. Accessed: 2024-10-11.
- [58] Wikipedia contributors, “2024 Attica wildfires – Wikipedia, The Free Encyclopedia.” https://en.wikipedia.org/wiki/2024_Attica_wildfires, 2024. [Online; accessed 16-Oct-2024].

- [59] Copernicus Emergency Management Service, “Copernicus EMS Rapid Mapping.” <https://rapidmapping.emergency.copernicus.eu/>, 2024. [Online; accessed 16-Oct-2024].

A Classical Disturbance Metrics

Metric Short-name	Description
<i>Transformer*</i>	Disturbance metric using per-pixel Mahalanobis distance of the post-image relative to normal RV with parameters estimated from deep-learning model. See Figure 4 for details.
<i>Log Ratio</i>	The decrease of the VH imagery in the log-domain when compared to the per-pixel median of pre-images. See Eqn. 10.
<i>Mahalanobis VH</i>	Mahalanobis metric computed from post-image and empirical normal RV determined by sample statistics as in Eq. 3.
<i>Max Mahalanobis</i>	Mahalanobis metric computed for VH and VV post-images separately identical to <i>Mahalanobis VH</i> above. Then per-pixel max across the polarizations.
<i>Mahalanobis 2d</i>	The 2-dimensional Mahalanobis metric defined in Eq. 11, where the mean and covariance are estimated via sample statistics.
<i>CuSum</i>	The maximum of a probabilistic CuSum metrics computed for each polarization as defined in Eqn. 17-18.

Table 5: Short-names of DIST-S1 metric and other conventional metrics in this Appendix. The (*) is the transformer metric used for the DIST-S1 algorithm.

The DIST-S1 algorithm will provide a probabilistic view of *generic* disturbances at the near global scale from RTC-S1 time-series. However, in order to demonstrate the algorithms efficacy, it is important to compare the DIST-S1 algorithm to conventional disturbance algorithms, which is what we do here. For our discussion of conventional algorithms, we consider signal-based, metric-based algorithms. We provide nomenclature (i.e. short-names) for the DIST-S1 metric and the conventional metrics we will discuss here. All the metrics presented here are implemented in our open-source library `distmetrics` [43]. By no means are the conventional metrics here a comprehensive list, and additional change metrics can be found in [51] and [52] for example. Our selection is motivated by NISAR’s

plan to use the *CuSum* algorithm for their annual disturbance product over specific sites [53].

As noted in the text, we call the DIST-S1 algorithm's metric simply the *transformer* in reference to the transformer architecture used to estimate μ and σ (more details follow in Section 2.4). We have summarized how this metric is computed from inputs in Figure 4. We will define the *log-ratio* metric as the metric which tracks the *decrease* in a VH time-series. Define L per pixel (dropping spatial indices):

$$L := \log(I_{n+1}^{VH}) - \log\left(\text{med}_{k=1,\dots,n} I_k^{VH}\right)$$

This is simply the dB difference of the post-image with the per-pixel median of the reference pre-images. The log-ratio metric is thus defined:

$$\ell_{ij} := \begin{cases} -L_{ij} & \text{if } L_{ij} < 0 \\ 0 & \text{otherwise} \end{cases} \quad (10)$$

First notice, we artificially set the metric to be 0 if the median is smaller than the value of the post-image at any given pixel. Next, note the value inspected is the negative of the log ratio discussed in the introduction in Eq. 1. This slightly modified metric aligns with our interpretation of disturbance metrics in that larger values indicate larger disturbance. Here, a decrease in cross-polarized backscatter is indicative of vegetation or forest loss [54].

We will call the metric derived using the empirical sample estimates from Eq. 3 via the Mahalanobis distance as the *1D Mahalanobis Metric* for a given polarization. We will also consider the maximum of the 1D Mahalanobis metric for each polarization as the *Max Mahalanobis* metric. We can also estimate the full covariance matrix Σ from the 2D pre-images and compute the Mahalanobis distance as:

$$d_{2d} := \sqrt{(I_{n+1} - \hat{\mu})^T \hat{\Sigma}^{-1} (I_{n+1} - \hat{\mu})} \quad (11)$$

where $\hat{\mu}$ is the 2-dimensional vector of VV and VH means computed from the pre-images and similarly $\hat{\Sigma}$ is the 2×2 covariance matrix computed from this data.

NISAR will utilize the so-called Cumulative Sum (CuSum) change detection method to detect forest loss [53]. The CuSum algorithm has a slightly different perspective on how change is detected. The algorithm was first proposed in [55] and considers the cumulative sum of residuals of the images from a drift parameter, frequently set to be the mean of the time-series. Let

$$r_n = \sum_{k=1}^n (I_k - \hat{\mu}_n) \quad (12)$$

where $\hat{\mu}_n$ is the empirical mean computed from the first n pre-images. As noted in [55], $\hat{\mu}_n$ can be replaced with any "drift" term that approximates the distribution. In the NISAR

ATBD [56], the average of the per-pixel maximum and minimum is used as this drift term. Typically, this metric can serve as an $\mathbb{R}$ -valued metric such that when its value sufficiently deviates from 0 can indicate change in given polarization VV or VH polarized imagery. However, since VH and VV have different dynamic ranges (i.e. VH has smaller backscatter returns than VV), it is helpful to translate this single-channel metric into a probabilistic setting following [57] so the two polarization can be compared more easily.

Let R_n be the random variable:

$$R_n := \sum_{k=1}^{k=n} (X_k - \hat{\mu}_n) / \hat{\sigma}_n.$$

Assume that the assumptions of Central Limit Theorem hold, then as $t \rightarrow \infty$, this random variable converges to a normal distribution with $N(0, \sqrt{n})$. To obtain a *standard* normal distribution, we define the random variable S_n as:

$$S_n := \sum_{k=1}^{k=n} \frac{X_k - \hat{\mu}_n}{\hat{\sigma}_n \sqrt{n}} \sim N(0, 1) \quad (13)$$

What's nice about this modeling choice is that we only assume that the X_t are independent and identically distributed. Similarly, we can compute the following value from the time-series:

$$T_n = \sum_{k=1}^{k=n} \frac{I_k - \hat{\mu}_n}{\hat{\sigma}_n \sqrt{n}} \quad (14)$$

We highlight that S_n is a random variable whereas T_n is a real number. We can therefore define $q(T_n) := P(|S_n| > |T_n|)$. Following the same computation from Eqn. 2, we can find that:

$$q(T_n) := P(|S_n| > |T_n|) \quad (15)$$

$$= 2(1 - F(|T_n|)) \quad (16)$$

This allows us to take the cumulative residuals and translate them into a probability, and compare the probabilities these residuals deviate from the mean in each polarization. We will define the metric for a polarization p as

$$c^p := 1 - q(T_n^p). \quad (17)$$

We again have the interpretation that larger values indicate more disturbance, though c must lie in the closed interval $[0, 1]$. We define the *CuSum* metric as

$$c := \max_{p \in \{VV, VH\}} c^p \quad (18)$$

It is worth noting that the in NISAR ATBD [53, 56], change detection is performed after a full year of data is collected and so the metric here differs slightly. We have adapted the CuSum method directly so that it can determine change with each new acquisition using the cumulative residuals. This CuSum metric combines both polarizations so that it indicates change in either polarization.

It is well known that these conventional methods do not account for trend nor seasonality [56]. As such, time-series adjustments are performed. For example, RADD applies a trigonometric fit applied to the tropical forest data [6, 7] in order to remove seasonality. NISAR currently applies is a de-trending step for CuSum [56] on a per-time-series basis.

B Accuracy Measures

We quickly review standard, relevant accuracy measures that we will use to discuss our accuracy assessments for calibration and validation of the DIST-S1 disturbance delineation. A statistical learning textbook can serve as an additional reference e.g. [25]. For any binary classifier (i.e. a classifier delineating “disturbed” and “undisturbed” areas) and associated validation data of the same label, we can describe True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN). The first word in the compound descriptor refers to the agreement of the classification relative to the validation data (or ground-truth data). The second word describes the label given to the input data by the classifier. For example, False Positives are classified as positive by the classifier but disagree with the validation data. We provide a table for some of the standard abbreviations used in the section in Table 6.

Accuracy Metrics	Abbreviation
True Positive	<i>TP</i>
False Positive	<i>FP</i>
True Negative	<i>TN</i>
False Negative	<i>FN</i>
F_1 Score	F_1
Jaccard Score	Jacc
Accuracy	Acc

Table 6: Common Accuracy Metrics and Their Abbreviations

We can now define *precision* and *recall*. The former measures how well the classifier performs with respect to all labels given by the classifier. In this discussion, the most important class is “disturbance.” The precision is defined as

$$\text{Precision} := \frac{TP}{TP + FP}$$

The Recall metric captures how well the classifier performs relative to a given ground-truth class label; again, we focus on the “disturbance” label. The recall is defined as:

$$\text{Recall} := \frac{TP}{TP + FN}.$$

The F_1 score is the harmonic mean between precision and recall balancing the accuracy of a given label by the classifier and the retrieval of the ground truth for disturbance.

$$F_1 := \frac{2}{\text{Precision}^{-1} + \text{Recall}^{-1}}.$$

More generally:

$$F_\beta := \frac{1 + \beta^2}{\text{Precision}^{-1} + \beta^2 (\text{Recall}^{-1})}, \quad (19)$$

which weighs recall by a factor of β over precision. We will use $\beta = 1/2$ to calibrate thresholds in Section 2.5.1. We also evaluate our disturbance delineation using the Jaccard similarity coefficient to measure the similarity between the DIST-S1 disturbance delineation and the validation data. The Jaccard similarity is defined as the size of the intersection divided by the size of the union:

$$\text{Jacc}(A, B) := \frac{|A \cap B|}{|A \cup B|}$$

which can be rewritten for a binary classifier as:

$$\text{Jacc} := \frac{TP}{TP + FP + FN}.$$

The total accuracy of the classifier is given:

$$\text{Acc} := \frac{TP + TN}{TP + TN + FP + FN} \quad (20)$$

All the metrics in this section take values in the interval $[0, 1]$. It’s important to note that disturbance is a sparse occurrence in space and time. That means the set of “undisturbed pixels” is a much, much larger set than “disturbed pixels.” We often call classes with largely different sizes *imbalanced*. Thus, total-accuracy is not as informative a metric as the others presented because a classifier that simply ignores all disturbance will likely have high total accuracy. For imbalanced classes, it is preferable to analyze precision, recall, Jacc, and F_1 for disturbance [25]. For the remainder of this document, all metrics are used to describe the “disturbed” label determined by the DIST-S1 algorithm. Specifically, our classifier’s “positive” label is an estimate of a disturbed pixel.

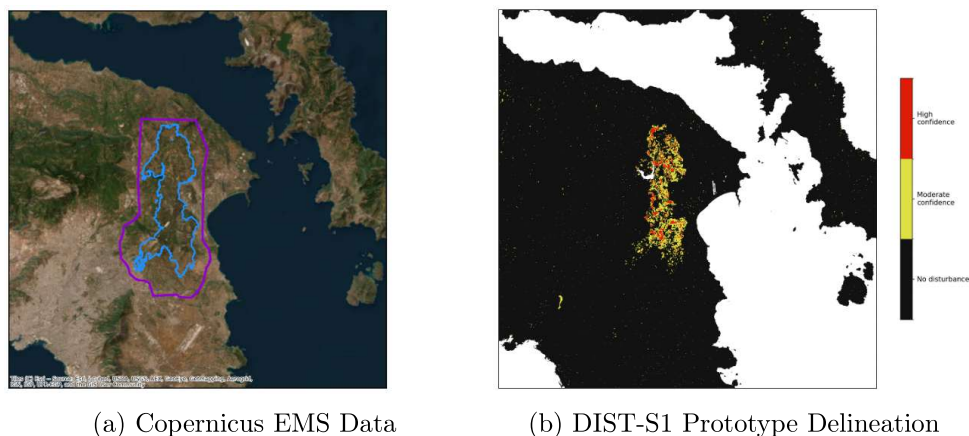


Figure 11: Above is an illustration of disturbance event data showing the burn perimeter of the wildfires impacting the Attica region of Greece during August 2024 [58]. In (a), data from Copernicus Emergency Management Services (EMS) [59]. In dark blue, the Area-of-Interest (AOI) showing the total area that was mapped with VHR optical data. In light blue, the burn perimeter is identified. In (b), the DIST-S1 prototype is shown. The DIST-S1 product was generated using an acquisition from August 26th, 2024 (approximately 2 weeks after the fires began) along with the 10 most recent pre-images prior to the event start (August 11th, 2024).

C Comparison of the Transformer Metric to CuSum

This appendix contains visual comparisons with the event data early in the design of the DIST-S1 product. It showcases how the transformer metric outperforms the classical CuSum metric introduced in Appendix A for a variety of events. We will show precision-recall (PR) curves and provide the area under the curve (AUC) value for these curves. Comparing these curves allows us to see overall how the transformer metric compares to the CuSum algorithm generally. The closer the curves are to (1,1) (e.g. Figure 13), the better its performance relative to the optically-derived validation dataset. Similarly, the higher the AUC, the more robust the disturbance metric is at detecting the ground truth disturbance. In Figure 12, we show pre-event and post-event change delineations for the fire in Attica Greece.

These are referenced, but not essential to the text. We have the companion figures to the Chile Wildfire to those occurring in Attica region of Greece (see Figures 12-13).

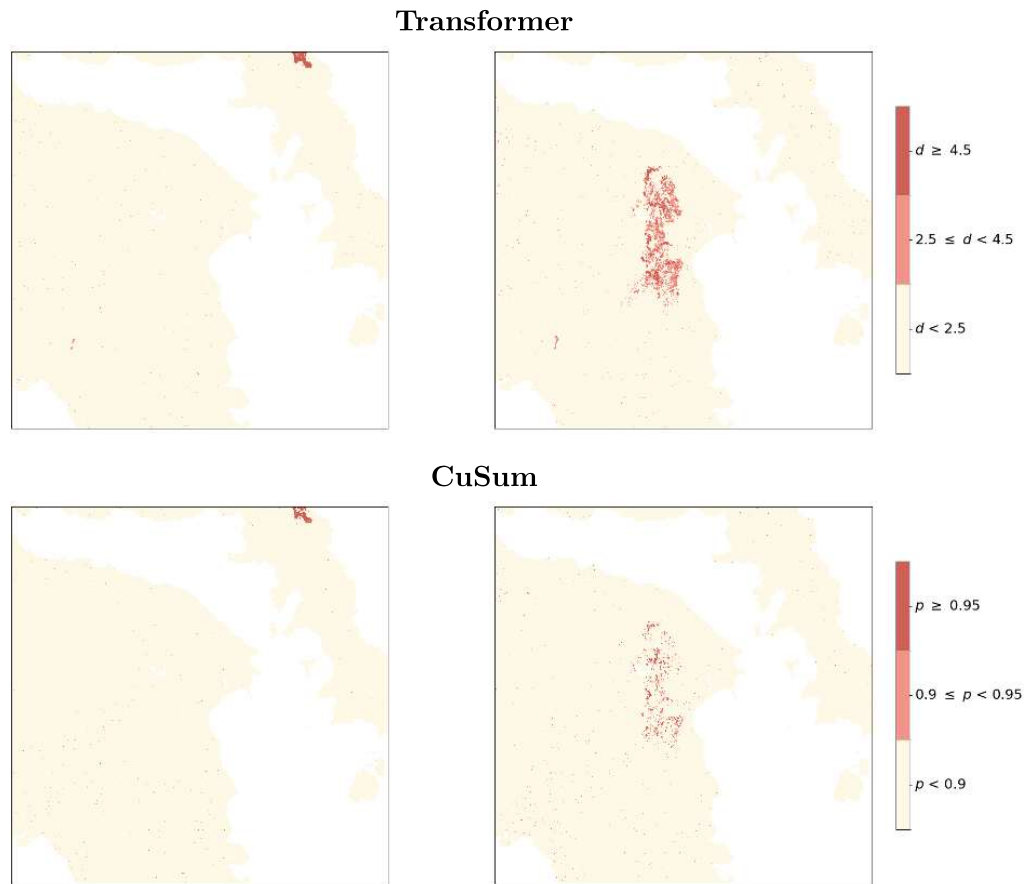


Figure 12: A comparison of CuSum and Transformer Metric over fire in Attica region of Greece. The pre-event image (left column) is on August 2nd, 2024. The post-event image is on August 26th, 2024. The fires are said to have started sometime between August 11-14 [58].

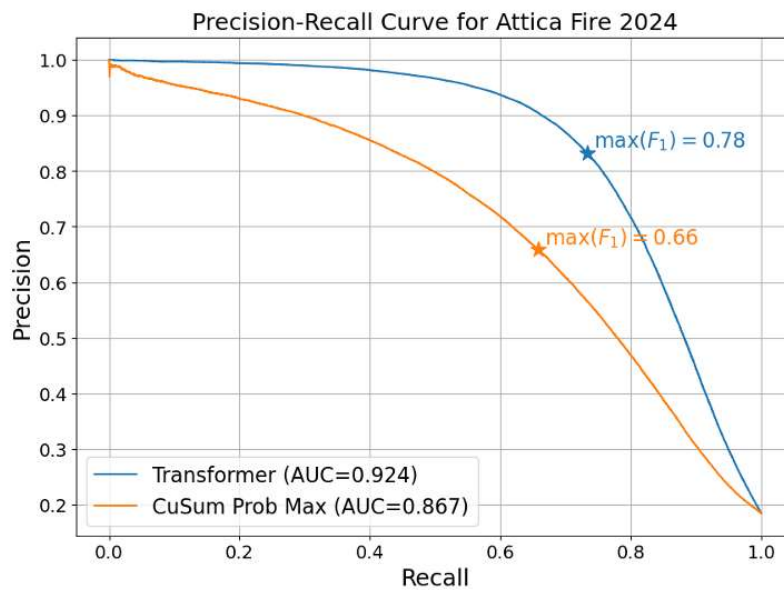


Figure 13: Precision recall curve by parametrizing binary classifiers with threshold $t > 0$ from Eqn. 7. The curves represent the precision and recall over each threshold across the Attica Fire (see Figure 11).

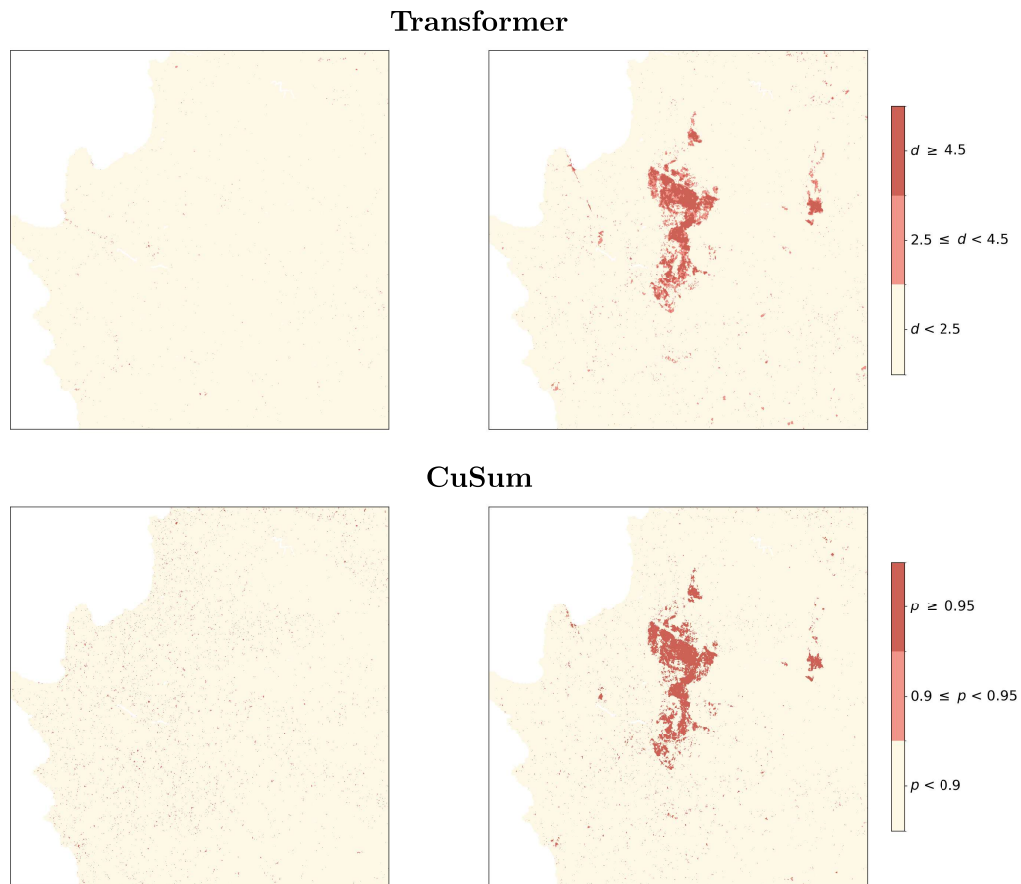


Figure 14: A comparison of CuSum and Transformer Metric over the Chile Fire in the Valparaíso region in February 2024. The pre-event images in the left column are January 12th, 2024 and the post-event images in the right column are on February 17th. The fire start occurred approximately on February 4th.

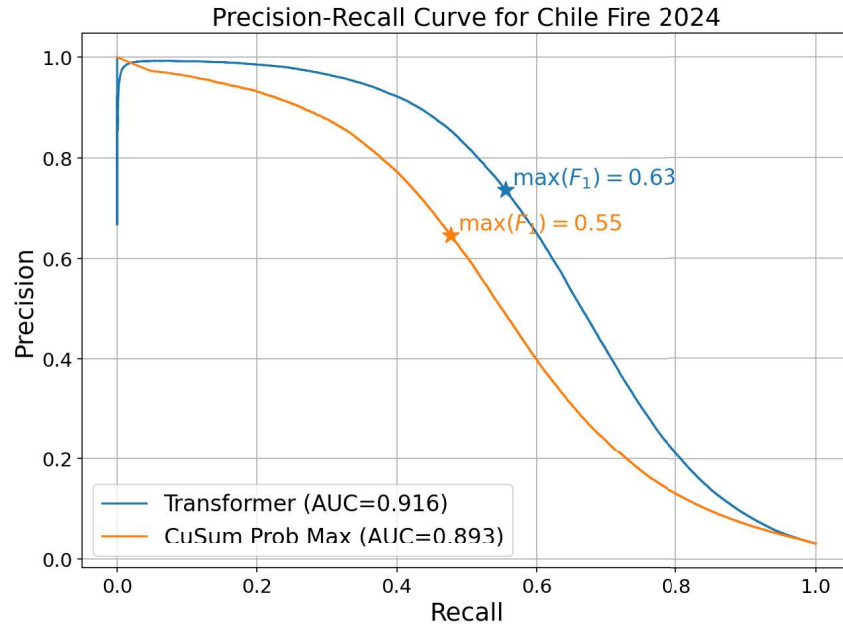


Figure 15: Precision recall curve by parametrizing binary classifiers with threshold $t > 0$ from Eqn. 7. The curves represent the precision and recall over each threshold across the possible values. The curves are from the wildfire occurring in the Valparaíso region of Chile in February 2024.

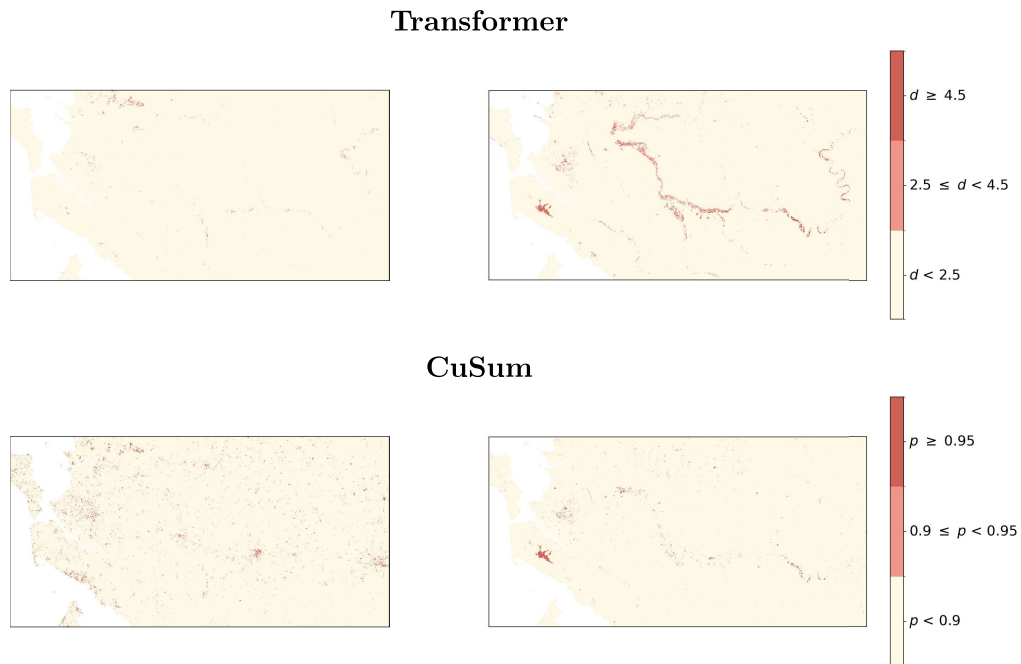


Figure 16: A comparison of CuSum and Transformer Metric over the South West France region on December of 2023. The pre-event images in the left column are November 24th, 2023 and the post-event images in the right column are on December 18th. The Flood occurred approximately on December 14th.

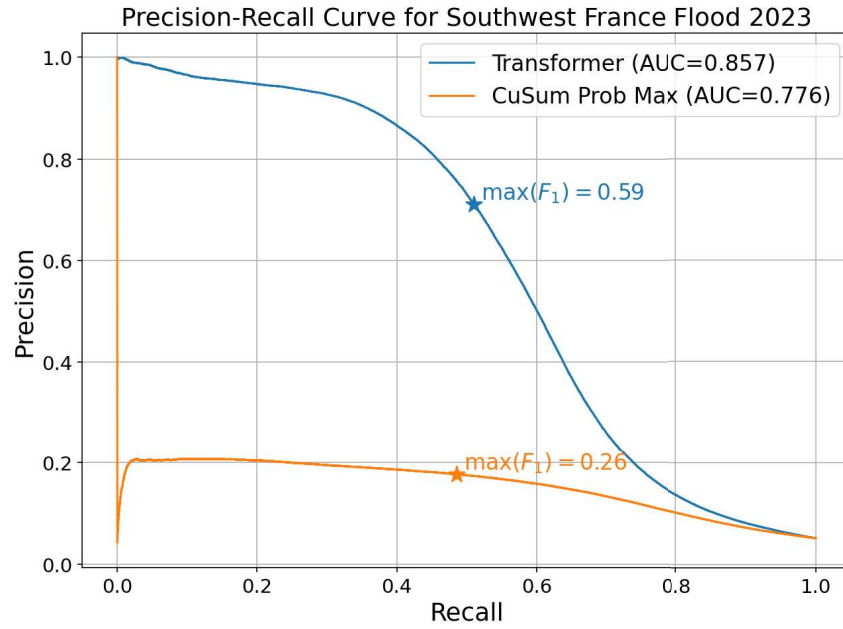


Figure 17: Precision recall curve by parametrizing binary classifiers with threshold $t > 0$ from Eqn. 7. The curves represent the precision and recall over each threshold across the possible values. The curves are from the flood occurring in the South West region of France in December 2023.

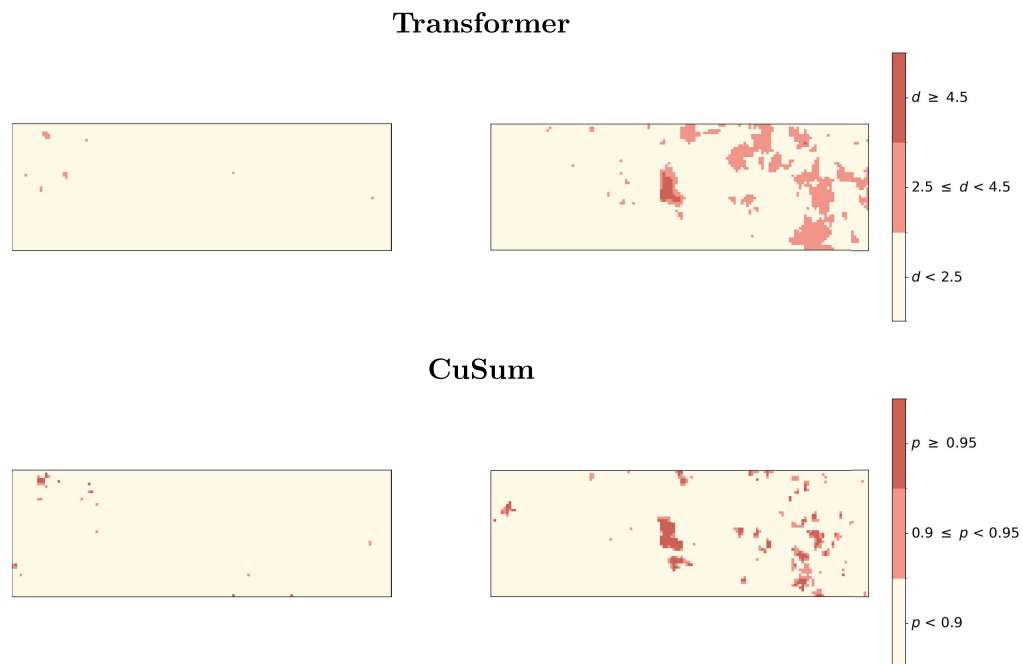


Figure 18: A comparison of CuSum and Transformer Metric over Papua New Guinea region on May of 2024. The pre-event images in the left column are on May 22nd, 2024 and the post-event images in the right column are on June 3rd. The landslide occurred approximately on May 24th.

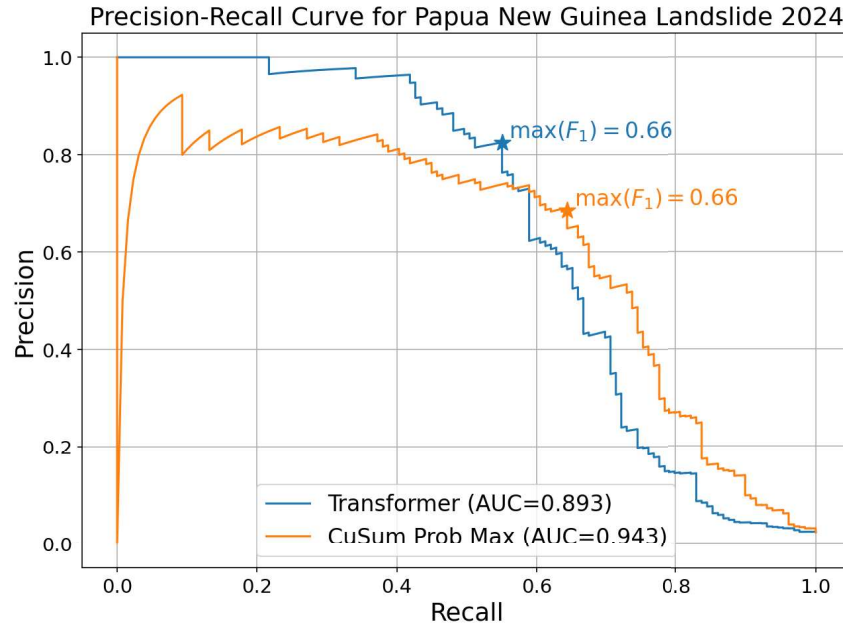


Figure 19: Precision recall curve by parametrizing binary classifiers with threshold $t > 0$ from Eqn. 7. The curves represent the precision and recall over each threshold across the possible values. The curves are from the landslide that occurred in Papua New Guinea on May 2024.