

An Exploration Framework to Identify and Track Movement of Cloud Systems

Harish Doraiswamy, *Member, IEEE*, Vijay Natarajan, *Member, IEEE*, and Ravi S. Nanjundiah

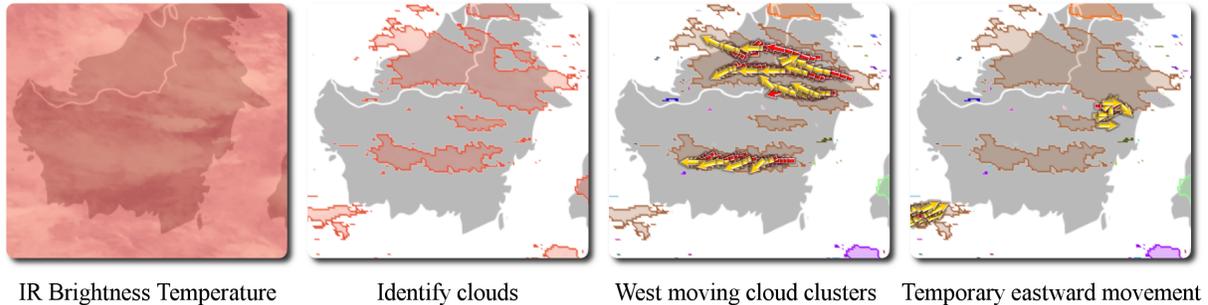


Fig. 1. Studying cloud systems at different scales. Our framework is used to study the movement of the equatorial Madden Julian Oscillation (MJO) over the Indian Ocean. Given the IR brightness temperatures over the island of Borneo, we first identify the set of clouds. Users can select clouds of interest and track their movement. Smaller scale cloud systems embedded in a MJO move in a westward direction. The manifestation of a convectively coupled kelvin wave results in a temporary eastward movement of parts of the cloud cluster. Such movement can be easily obtained using the querying ability of our framework. The rightmost figure shows a subset of the clouds that move eastward for at least 90 minutes. The temporary movement is indicated by the fact that the movement reverts to its original westward direction after a short duration. Our framework also helps quantify the overall eastward propagation of the MJO.

Abstract— We describe a framework to explore and visualize the movement of cloud systems. Using techniques from computational topology and computer vision, our framework allows the user to study this movement at various scales in space and time. Such movements could have large temporal and spatial scales such as the Madden Julian Oscillation (MJO), which has a spatial scale ranging from 1000 km to 10000 km and time of oscillation of around 40 days. Embedded within these larger scale oscillations are a hierarchy of cloud clusters which could have smaller spatial and temporal scales such as the Nakazawa cloud clusters. These smaller cloud clusters, while being part of the equatorial MJO, sometimes move at speeds different from the larger scale and in a direction opposite to that of the MJO envelope. Hitherto, one could only speculate about such movements by selectively analysing data and a priori knowledge of such systems. Our framework automatically delineates such cloud clusters and does not depend on the prior experience of the user to define cloud clusters. Analysis using our framework also shows that most tropical systems such as cyclones also contain multi-scale interactions between clouds and cloud systems. We show the effectiveness of our framework to track organized cloud system during one such rainfall event which happened at Mumbai, India in July 2005 and for cyclone Aila which occurred in Bay of Bengal during May 2009.

Index Terms—Cloud clusters, tracking, computational topology, split tree, weather and climate simulations.

1 INTRODUCTION

Clouds play a very important role in tropical circulation. They are a source of heating through latent heat release. They also interact and modify radiation by reflecting and absorbing radiation. Cloud systems, sometimes known as Mesoscale Convective Systems (MCS), consist of a collection of clouds of varying heights. These include tall growing and mature Cumulonimbus(Cb) clouds surrounded by a stratus (lower

clouds) [19]. The tall clouds are the mature cumulus cells and could be as high as 15 km, which is the typical tropopause height in the tropics. The cloud systems could also move, grow and decay. These MCS in turn could be embedded in larger systems such as a tropical cyclone. Considerable amount of rain can occur in association with such MCS.

Movements of cloud systems could have large temporal and spatial scales such as the Madden Julian Oscillation (MJO), which has a spatial scale ranging from 1000 km to 10000 km and time of oscillation of around 40 days. Embedded within these larger scale oscillations are a hierarchy of cloud clusters which could have smaller spatial and temporal scales such as the Nakazawa cloud clusters [34]. These smaller cloud clusters, while being part of the equatorial MJO, sometimes move at speeds different from the larger scale and in a direction opposite to that of the MJO envelope.

The complex movement of clouds within a cloud system can be used to identify if a merger is likely between clouds. A merged cloud could indicate more unstable weather. The merger and movement of such cloud clusters can also lead to high intensity rainfall events during the monsoon season. Techniques to track and visualize such cloud system interactions could therefore be effective for nowcasting, where using near-real time satellite data, merger and movement of cloud systems could be used for forecasting high intensity rainfall events.

• Harish Doraiswamy is with Department of Computer Science and Engineering at Polytechnic Institute of New York University. E-mail: hdoraisw@poly.edu.

• Vijay Natarajan is with Department of Computer Science and Automation and Supercomputer Education and Research Centre at Indian Institute of Science, Bangalore. E-mail: vijayn@csa.iisc.ernet.in.

• Ravi S. Nanjundiah is with Centre for Atmospheric and Oceanic Sciences and Divecha Centre for Climate Change at Indian Institute of Science, Bangalore. E-mail: ravi@caos.iisc.ernet.in.

Manuscript received 31 March 2013; accepted 1 August 2013; posted online 13 October 2013; mailed on 4 October 2013.

For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org.

Advent of better technology has affected in increasing the availability of high resolution data (spatial resolution ~ 4 km and temporal resolution ~ 30 mins). Therefore, the amount of data that is to be processed when considering even a short time period such as a single day is large. A query-driven visualization framework [20, 49, 50] will therefore be useful for performing analysis and visualization on these datasets, which are both large and complex.

1.1 Problem Statement

Given the infrared (IR) brightness temperatures over a particular region, clouds present in the atmosphere over that region are captured as sub-level sets of a given temperature threshold. These clouds are then tracked over time in order to study various phenomena. Existing methods that track movement of cloud clusters fail to capture local movement within such a system. Also, when using existing methods, in addition to the threshold required for identifying clouds, both the identification and tracking require the users to input additional parameters. The movement obtained is dependant on the value of these parameters. Eliminating these additional parameters will not only simplify the operations performed by the user, but will also help provide consistent results that can be reproduced. Tracking clouds at different thresholds convey different information about the phenomena being studied. It is therefore important to be able to efficiently track movement of clouds at various thresholds.

In addition to the computational effort required to process the large amount of data available, manual analysis of such data over long time periods without a priori knowledge of the system is not practical. This necessitates the development of tools to query and explore cloud movements using this data.

1.2 Contributions

We use techniques from computational topology and computer vision to design a framework that allows users to interactively visualize and explore the movement of cloud systems at various scales in space and time. Our framework has the following capabilities:

1. Compute and explore clouds at different thresholds. This is accomplished using the split tree of the input at each time step.
2. Compute a good threshold range to identify clouds of interest. This is accomplished using the persistence diagram.
3. Visualize cloud movement at different level of detail – from its local intra cloud-cluster movement to long term inter-cloud system interactions. Local movement is tracked using an optical flow computation. The local movement together with the computed clouds is then used to compute a cloud motion graph, which tracks the interaction between clouds.
4. Support queries on both the properties of individual clouds, as well on their motion patterns.
5. Support queries on entire cloud systems.

A precipitation distribution captures the amount of rainfall over that region at a given time. Since IR brightness temperature and precipitation are inversely related, by reversing the values of the precipitation of a region, we show that the same techniques used for IR brightness temperature works for this input as well. The production of such data includes additional information from microwave channels, which is absent in IR brightness temperatures, and hence a combination of these two datasets could lead to better tracking of multiple scale cloud systems.

Finally, we show the utility of the developed framework through the following use case scenarios:

- Study and track the equatorial MJO over the Indian ocean.
- Study events that lead to intense rainfall over Mumbai, India, in July 2005.
- Study the movement of clouds during the period that the tropical cyclone Aila made landfall.

2 RELATED WORK

We describe the related work in two categories – those related to feature tracking and visualization, and those related to the study of cloud systems.

2.1 Feature Tracking and Climate Visualization

Several efforts in the past use computational topology to identify and track features of a spatio-temporal input. Laney et al. [28] and later Bremer et al. [8] use the Morse decomposition to identify features of the input and track these features across time using the geometric properties of the features. Pascucci et al. [37] identify features using merge trees, and track burning cells during turbulent combustion by computing the overlap of the features. Widanagamaachchi et al. [48] extend this technique and design a framework to explore time-varying data. Kasten et al. [26] map critical points of the input scalar function across time steps and create a merge graph that is used to track unsteady flow fields. While these methods track the movement of features, they do not capture orthogonal movement that might be present within the feature.

Directly related to this work, Gambheer and Bhat [19] track clouds by considering the overlap of sub-level sets across time steps. More recently, Fiolleau and Roca [17] consider the time varying two dimensional input as a three-dimensional volume, and track clouds by tracking seed points within this volume. In addition to the threshold used to identify clouds, these techniques also require users to specify other parameters as input. Again, these techniques while capturing cloud movement, miss any different movement that is present within cloud systems.

There has been some recent work on developing visualization frameworks for the exploration of climate and weather data. Lundblad [29] design a software for visualizing weather and ship data. Ladstädter et al. [27] design a framework to explore the variables of climate model data. Santos et al. [42, 43] propose a work flow based, provenance enabled system, called UV-CDAT, that integrates climate data analysis libraries and visualization tools into a single application. Work on visualizing clouds focus on accurately rendering clouds [39]. To the best of our knowledge, there exists no work that allows for exploring and querying patterns of cloud motion.

2.2 Cloud Systems

Occurrence of an MJO is typically identified in a meteorological dataset by using the spectral analysis technique of Wheeler and Kiladis [47], which relates spatial frequency with spatial scale of the harmonic. A variable such as outgoing longwave radiation (OLR) is generally averaged over a large latitudinal extent such as 15°N - 15°S and space-time harmonic analysis conducted for the entire latitudinal belt. An MJO is identified if power exists at a global wave number of 3-5 and time-period corresponding to 40-50 days. Tracking of meridional oscillations on the intra-seasonal scale has been attempted by using continuous space-time wavelets and searching of significant energy on the spatial scale of 4000 km and a related peak in the 40-50 day scale [10, 38]. Neither of these bring out the rich structure of multi-scale interactions which our proposed framework attempts.

3 BACKGROUND

In this section, we briefly introduce the necessary background on techniques from computer vision and computational topology that form the mathematical and algorithmic basis of the proposed framework for visualizing and exploring cloud systems. Comprehensive discussions on these concepts is available in several textbooks [14, 21, 32, 45].

3.1 Scalar functions and Level sets

A *scalar function* is a function that maps points in a spatial domain to the set of real values \mathbb{R} . Scalar functions are used to represent temperature or precipitation data from satellite images or weather simulations. Figure 2(a) shows an example of a scalar function defined on a plane. The function is visualized using a colour map, where white denotes a zero function value and the colour progresses towards dark red as the function value increases.

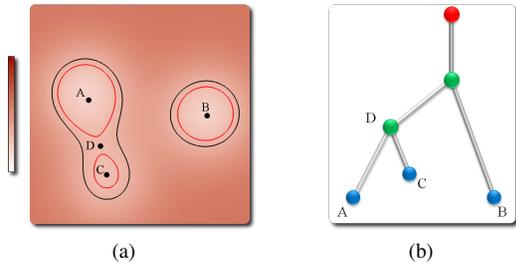


Fig. 2. Scalar function and split tree. **(a)** A scalar function with three minima labelled A, B, and C. It is visualized using the colour map shown on the left. A low function value corresponds to white colour, and the colour progresses towards dark red as the function value increases. Level sets at two function values are shown as red and black lines respectively. A sub-level set corresponds to the region within the level set and a super-level set corresponds to the region outside the level set. **(b)** The split tree tracks the components of the sub-level sets of a scalar function as the function value increases.

The scalar function is typically represented using a structured grid G , together with a piecewise bilinear function $f: G \rightarrow \mathbb{R}$. The function is defined on the vertices of the grid and the function value at a point within a cell of the grid is computed using bilinear interpolation.

The preimage $f^{-1}(a)$ of a real value a of a scalar function f is called a *level set*. It is the set of all points on the domain having function value equal to a . A *sub-level set* of a real value a is defined as the preimage of the interval $(-\infty, a]$. It is the set of all points having function value less than or equal to a . Similarly, the *super-level set* of a is the preimage of the interval $[a, +\infty)$. The closed red and black curves in Figure 2(a) denotes two level sets of the input function. The region inside these curves denote sub-level sets, while the region outside the curve denotes super-level sets.

3.2 Join tree and Split tree

Consider a sweep of the input function f in increasing order of function value. The nature of topological change to the sub-level sets of f when the sweep encounters a vertex determines the vertex type.

1. The vertex is *regular*: The topology of the sub-level sets do not change.
2. The vertex is a *minimum*: A new sub-level set component is created. The scalar function shown in Figure 2(a) has three minima A, B, and C.
3. The vertex is a *split saddle*: Two sub-level set components merge into one. This is equivalent to one of the components being destroyed. The point D in Figure 2(a) represents one such saddle where the sub-level set components created at minima A and C merge into one.

The *split tree* tracks the changes in the connectivity of sub-level sets of the input scalar function. Nodes of the split tree correspond to the set of minima and split saddles of f . Figure 2(b) shows the split tree corresponding to the input function shown in Figure 2(a).

The *join tree* of f is defined similarly, and tracks the connectivity of the super-level sets of f . Nodes of the join tree correspond to the set of maxima and join saddles of f . A *maximum* is a vertex where a new super-level set component is created during the sweep of the input in decreasing order of function value, and a *join saddle* is a vertex where two super-level set components merge into one.

A vertex that is not regular, such as a minimum or a saddle, is called *critical*. Efficient algorithms to compute join and split trees of an input scalar function can be found in [9, 11, 30, 36]. Regular vertices are often inserted into the join / split tree as degree-2 nodes to obtain an *augmented join tree / augmented split tree*. Clouds may be represented as sub- or super-level set components of an appropriate scalar function. The split and join tree help extract and analyse individual clouds.

3.3 Topological persistence

Consider the sweep of the input function f in increasing order of function value. As mentioned above, the topology of the sub-level sets change when this sweep encounters a critical point. A critical point is called a creator if a new component is created and a destroyer otherwise. It turns out that one can pair up each creator v_1 uniquely with a destroyer v_2 that destroys the component created at v_1 . The persistence value of v_1 and v_2 is defined as $f(v_2) - f(v_1)$, which is intuitively the lifetime of the feature created at v_1 , and is thus a measure of the importance of v_1 and v_2 . In this paper, we are only interested in the persistence of the set of minima or the set of maxima. For an appropriately chosen scalar function, the persistence is a measure of the size of the corresponding cloud. Given an input domain of size n , the persistence of such features can be computed efficiently in $O(n \log n + n\alpha(n))$ time using the union-find data structure, as compared to the cubic-time required by the algorithm to compute general topological persistence [13, 15]. The *persistence diagram* [12, 15] plots the features of the input function as set of points on a 2D plane, where the x - and y -coordinates of a feature corresponds to its birth and death time respectively.

3.4 Optical flow

Given a sequences of images over time, the *optical flow* is the *apparent motion* of the brightness patterns in the image [22, 23]. It is computed as a velocity field, which provides the velocity with which every pixel of an image moves. Studying the optical flow helps understand the motion of objects in the images. In the context of weather data, we aim to use it to understand the motion of features such as clouds. The basis of most optical flow computation algorithms is the assumption of *brightness constancy*, which states that when a pixel flows from one image to another, its intensity or colour does not change. Let $I(x, y, t)$ denote the intensity of the pixel (x, y) at time t . Let the pixel have flow $(u(x, y, t), v(x, y, t))$. Then, the brightness constancy can be written as

$$I(x, y, t) = I(x + u, y + v, t + 1).$$

Linearizing by applying a first-order Taylor expansion to the right hand side yields the approximation

$$I(x, y, t) = I(x, y, t) + u \frac{\delta I}{\delta x} + v \frac{\delta I}{\delta y} + 1 \frac{\delta I}{\delta t}.$$

This gives the *optical flow constraint* equation

$$u \frac{\delta I}{\delta x} + v \frac{\delta I}{\delta y} + 1 \frac{\delta I}{\delta t} = 0.$$

Both the brightness constancy equation and the optical flow constraint equation provide just one constraint on two unknowns for each pixel. Algorithms computing the optical flow introduce additional constraints in order to solve the above equations. We refer the reader to multiple surveys on the topic for a comprehensive discussion of algorithms for computing the optical flow [4, 5, 6, 18, 33, 35, 44, 46].

4 TRACKING CLOUD MOVEMENT

Clouds are masses of condensed water vapor floating in the atmosphere. They are spatial features that evolve over time. This section describes methods for efficient identification of clouds and tracking of cloud movement based on techniques presented in the previous section.

4.1 Identifying Clouds

Two scalar functions commonly used by climate scientists to study clouds are *infrared (IR) brightness temperature* and *precipitation*, respectively. The function is sampled on a grid representing the geographic region of interest. We now describe how a cloud is identified using these two functions. Clouds are extracted from both types of data using level set analysis.

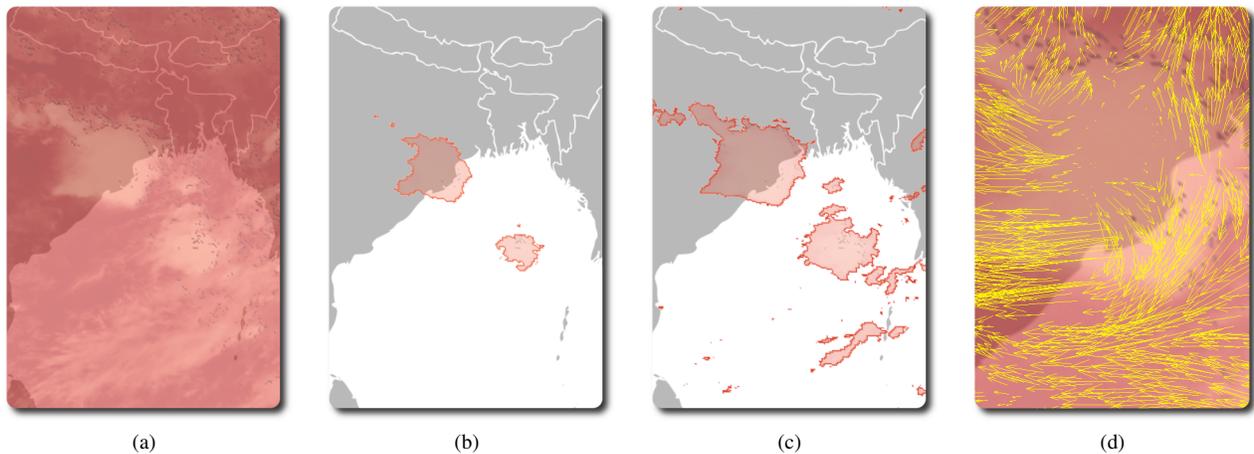


Fig. 3. Identifying and tracking clouds using IR brightness temperature data. **(a)** The input scalar function represents the IR brightness temperature over the east coast of India (8° to 20° latitude, 80° to 95° longitude) at 15:00 hours UTC on 1 June, 2006. **(b)** The set of clouds computed as a sub-level set at threshold 205K. **(c)** The set of clouds computed as a sub-level set at threshold 235K. **(d)** The vector field obtained using the optical flow algorithm. The flow around the large cloud on the left in (b) is shown.

4.1.1 IR brightness temperature

The IR brightness temperature data that we use in this work is the CPC-merged dataset [1]. This dataset has been prepared by merging pixel resolution IR brightness data from GOES/8/10, METEOSAT-7/5 and GMS geostationary satellites at a spatial resolution of 4km and temporal resolutions of 30 minutes. The data is available from February 2000 to present. This is a unique dataset that allows study of tropical systems such as Cyclone and Monsoons at multiple scales from the local cloud scale to the planetary scale with a single comprehensive dataset.

Given the IR brightness temperature distribution, the set of clouds over the corresponding geographical region is identified as the collection of sub-level set components for a given temperature threshold t . Commonly used thresholds are approximately in the temperature range $t \in [192K, 235K]$. Figure 3(a) shows the IR brightness temperature over the east coast of India at 15:00 hours UTC on June 1, 2006. Figures 3(b) and 3(c) show the clouds during that period computed at thresholds $t = 205K$ and $t = 235K$, respectively.

The threshold values usually depend on whether the clouds of interest are tall or short. Colder temperatures ($\leq 210K$) indicate tall clouds with cloudtops at 7.5 km or higher and are associated with *Cumulonimbus* (Cb) cells, while higher temperatures indicate short clouds that are associated with *Stratus* clouds [31]. However, there is no single good value for t . We propose the use of the persistence diagram to identify the threshold of interest.

A cloud is represented by the smallest minimum contained in the corresponding sub-level set component. We generate the persistence diagram of the set of all minima of the IR brightness temperature. The persistence of a cloud is represented in the persistence diagram as the height of the corresponding point above the $x = y$ line. Clouds of interest often correspond to high persistent features. The persistence diagram helps locate highly persistent minima whose temperature lie within a larger feasible interval of values. The search space for the threshold is now reduced to the smaller temperature range containing these minima. Figure 4 shows the persistence diagram for the input shown in Figure 3(a). The most prominent clouds correspond to those formed in the threshold range [200K, 210K]. This is also validated by the fact that there are no new significant clouds in Figure 3(c), which was generated using a higher threshold of 235K.

4.1.2 Precipitation

While IR brightness temperature is a continuous function, precipitation is more discrete (zero or non-zero) in nature. Additionally using only IR brightness temperature could have problems in separating high level cirrus from tall Cb clouds. However, the use of precipitation data in combination with the IR brightness temperatures could differenti-

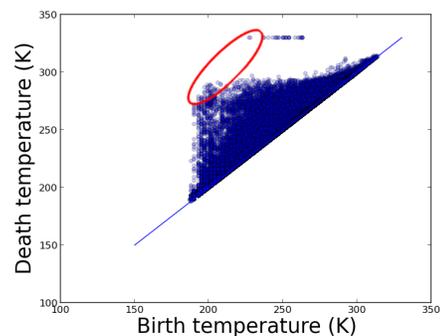


Fig. 4. Using persistence diagram of the input function shown in Figure 3(a) to identify a good threshold range to compute clouds. Note that the main concentration of high persistence clouds correspond to the threshold range [200K, 210K]. Also, we do not see any new significant clouds when a threshold of 235K is used to identify clouds (Figure 3(c)).

ate between tall Cb clouds which precipitate, and cirrus clouds (in the form of an anvil to tall Cb clouds) which could have low IR temperature but would not precipitate.

We use the 3B42 dataset [2] in our experiments, which combines data from TRMM precipitation radar, TRMM microwave image and infrared scanner merged with IR brightness from geostationary satellites. It has a spatial resolution of 0.25 degrees (approximately corresponding to 25 km in the deep tropics). The temporal resolution is 3 hours. This dataset spans December 1997 - present and is one of the widely used datasets for studying tropical convection at various temporal and spatial scales ranging from comparison with modelling studies at monthly scale to obtaining diurnal signatures of rainfall. More details about the creation of this dataset can be found in [24, 25].

When using precipitation as input, a higher value of precipitation corresponds to tall clouds since Cb cells are likely to produce more rain than stratus. Given the precipitation over a geographical region of interest, the set of clouds is computed as the collection of super-level sets for a given precipitation threshold p . Figure 5(a) shows the precipitation of the same region over the east coast of India as shown in Figure 3. Figures 5(b) and 5(c) show the set of clouds obtained using two different thresholds. We again use the persistence diagram of the set of maxima of the precipitation to identify an appropriate threshold and compute the clouds of interest. Unless otherwise specified, the examples in the remainder of this paper use IR brightness temperature as their input scalar function.

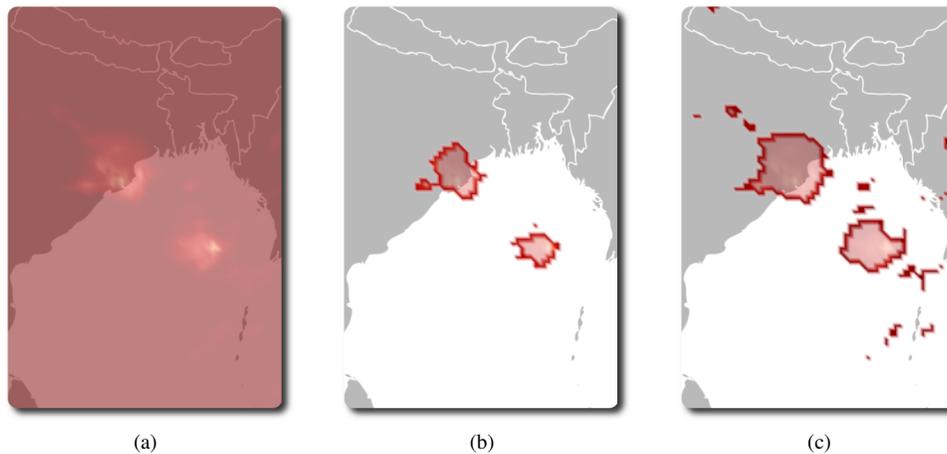


Fig. 5. Identifying clouds using precipitation data. **(a)** The input scalar function represents the precipitation over the east coast of India. Note that the colour map is now reversed. Zero precipitation corresponds to dark red, and the colour tends to white with increasing function value. **(b)** The set of clouds computed as a super-level set at a precipitation threshold=4. **(c)** The set of clouds computed as a super-level set at a precipitation threshold=1.

4.2 Tracking cloud movement

Climate scientists are often interested in studying the temporal properties of a cloud system such as its origin, how it interacts with other clouds, and its movement. Given the set of clouds at two consecutive time steps, we track the movement of clouds using the optical flow between the pair of IR brightness temperature images. In particular, our software uses an implementation of the Farneback algorithm [16] provided by OpenCV [7]. Figure 3(d) shows the velocity field for a time-step obtained using this algorithm. The advantage of using optical flow is two-fold: 1) by not explicitly representing each cloud with a single point, it captures intra-cloud system movements that are missed by existing methods, and 2) it results in a time-varying vector field that captures the smooth local motion of the clouds. As we show later in Section 6, the local motion can help visualize the cloud movement and, more importantly, it can also be used as a powerful tool for querying and identifying interesting patterns in cloud movements.

The velocity field obtained as the output of the optical flow algorithm is used to construct a *cloud motion graph*, that maps clouds from one time step to clouds in the next time step. This graph captures the movement and interaction between clouds over time. It essentially captures when clouds originate, how they merge and split with other clouds, and when they cease to exist. We use this graph to obtain a higher level representation of the cloud system.

5 VISUALIZATION OF CLOUD SYSTEMS

We now describe our software framework which allows users to explore and visualize the movement of cloud systems. For ease of explanation, we assume the input function is specified as IR brightness temperatures. Note that the same techniques described below will work for a precipitation input by reversing the function values. Also note that in order to use this system, users do not require knowledge of the underlying representation.

5.1 Exploring the data

Given the input data for different time steps, we first compute the set of clouds at each time step. We then compute the optical flow between consecutive time steps, and use the obtained velocity field to construct the cloud motion graph. The different data structures present in our framework can be used to support various queries over the given input. We now describe the different ways in which users can explore the input data.

5.1.1 Explore different thresholds

As mentioned in Section 4.1, there is no universally correct threshold to be used. Depending on the goal, climate scientists use different

thresholds to identify clouds. Without a priori knowledge of the system being studied, a trial and error method to obtain the right threshold becomes very cumbersome. While using the persistence diagram helps narrow this search space, computing the clouds at different possible thresholds is still computationally expensive. Moreover, since the search space is continuous, there are theoretically infinite possible thresholds that can be used.

The split tree (Section 3.2) captures threshold ranges in which there is no change to the structure of a cloud. If we order the critical points in increasing order of function value, then there is no topological change in the structure of clouds between the function values corresponding to two consecutive critical points. This provides a way to discretize the search space. Also, by storing the augmented split tree it is easy to recover the clouds at a threshold $t \pm \epsilon$ given the set of clouds at t by moving up / down the arcs of the split tree.

We therefore first compute the set of split trees for the given input, and use this as a data structure to identify clouds. In addition to providing a set of discrete function intervals that the users might be interested in, the split tree also allows us to efficiently compute clouds with changing threshold values.

5.1.2 Query the input

Both the velocity field obtained from the optical flow algorithm, and the cloud motion graph encode information about the motion of cloud systems. This enables us to support queries on the input, which greatly eases the effort of users to find patterns in cloud movements. Our framework supports queries on individual clouds as well as on cloud systems. The different queries supported by our system with respect to individual clouds are:

1. Query clouds based on direction of movement. Users are often interested in finding clouds that move in certain directions. Using the optical flow, we allow users to search for clouds that move in the four principal directions, i.e. north, south, east, and west.
2. Query clouds based on its longevity. This allows users to identify clouds that are alive for a given duration. The lifetime of a cloud is measured similar to persistence, based on its mergers and splits over time.
3. Query clouds based on its size. The user can view only those clouds that have a given size. While we currently use the area of the cloud as a measure of its size, it can be easily extended to support other measures of size as well.

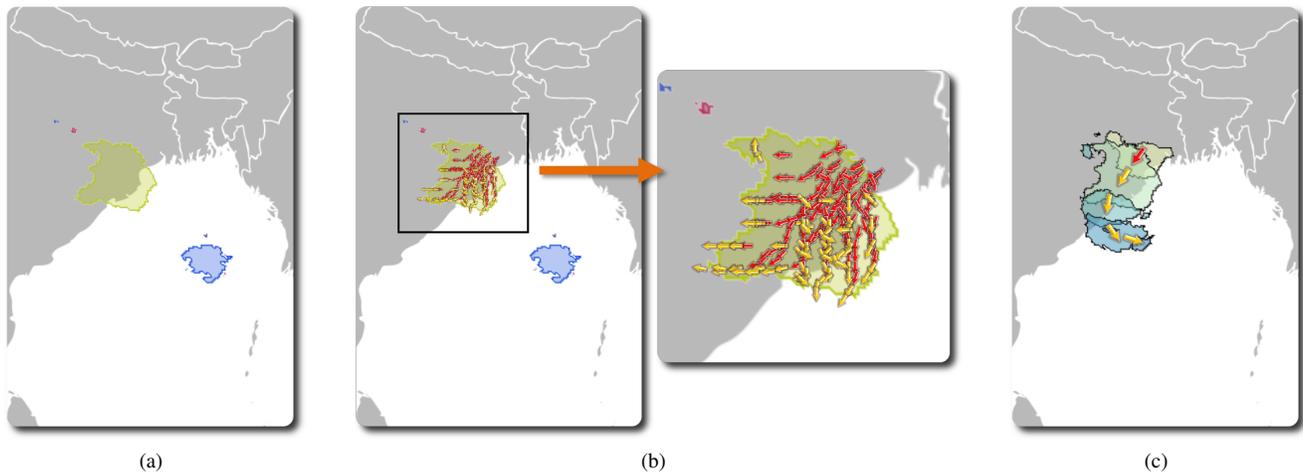


Fig. 6. Visualization of cloud movement. **(a)** The set of clouds at a particular time step. Clouds having the same colour correspond to those that have either interacted earlier, or will interact with each other in the future. **(b)** The different trajectories show the local movement of a selected cloud. Red arrows depicts the past trajectory, while the yellow arrows depicts the future trajectory. **(c)** The long term movement of the large green cloud present in (a). The position of the cloud at discrete time steps is visualized by varying the colour from yellow to blue over time. The approximate trajectory of this movement is also shown.

4. Query clouds based on its velocity. Knowing the rate at which clouds are moving, wind patterns can be estimated (generally known as Cloud Motion Vectors).

We also support queries based on movement, lifetime, size and velocity of cloud systems, which are represented as connected components of the cloud motion graph. In addition to this, we also support the queries on cloud systems based on their behaviour over time. An example of such query is when a user wants to identify cloud systems which is involved in a lot of merge / split activity between clouds. If two cloud systems merge, they are likely to form bigger systems which could result in higher rainfall / more unstable weather. Conversely splitting and moving away could indicate weakening of the system.

Our system also supports temporal queries, which the user can use to find time intervals that contain clouds or cloud systems having any of the above given properties.

In addition to the above queries on the movement of clouds, we also support queries to gather various statistics of the cloud system such as the area covered by the clouds and the frequency distribution of cloud heights. The latter is particularly of interest for climate scientists. The change in such a frequency distribution within a cloud system could indicate whether the cloud system is intensifying (having more taller clouds) or decaying (having less taller clouds). If multiple cloud systems with tall clouds are seen in a region, then more unsettled weather is highly likely over that region. One can also get an overview of the frequency distribution of cloud heights using the persistence diagram. Thus, such queries can be used as a now-casting tool.

5.2 Visualizing cloud movement

In addition to exploring the data, our framework also supports visualizing the interactions between clouds and their movement. We provide three ways in which this movement of clouds can be visualized – (a) by showing inter-cloud interactions, (b) by showing local movement of select clouds, and (c) by showing long term movement of select clouds. We now describe each of these visualizations in detail.

5.2.1 Cloud interactions

Users are interested in understanding how clouds interact with each other over time. In particular, they are interested in identifying clouds that belong to the same cloud system. Such interactions are captured by the connected components of the cloud motion graph, which represents the set of cloud systems. Let the edges of this graph be directed towards nodes having a later time step. Then, a node having more than one incoming edge would correspond to multiple clouds merging into

a single cloud, while a node having multiple outgoing edges would correspond to a cloud splitting to multiple clouds.

In order to show such interaction between clouds, we assign a unique colour to clouds that belong to a common component. Thus, when the user selects a time step, clouds having the same colour would correspond to those within a cloud system. Figure 6(a) shows clusters of clouds at a particular time step. Clusters are determined by the cloud interactions either in the past or in future.

5.2.2 Local cloud movement

The intra-cloud system movement can easily be overlooked when viewing an animation of the cloud system movements. This is especially true for potentially interesting short term movements which could identify interesting phenomena. Climate scientists are therefore interested in viewing such intra-cloud system movement of a cloud of interest as a static image. The velocity field obtained using the optical flow algorithm, which tracks the motion of individual pixels forming a cloud, may be processed to identify the intra-cloud system interactions.

We use the *streak lines* of the computed velocity field to visualize such movement. The user first selects a cloud of interest at a particular time step. We then sample points within this cloud depending on the required density specified by the user. Using the sampled points as seed points, we compute and display the streak lines. Climate scientists also want to differentiate between the history and future of these trajectories. We therefore colour code the streak lines depending on the time step of the selected cloud. Figure 6(b) shows the local movement of one such cloud. With respect to the time step of the selected cloud, the past movement of the cloud is represented using a red trajectory, while the future movement is represented using a yellow trajectory.

5.2.3 Long term cloud movement

Users are also interested in viewing the long term movement of clouds in a single frame, rather than keeping track of the clouds by either explicitly changing time steps, or through an animation of their movement. In addition to the long term direction of movement, they are also interested in tracking the size of the clouds over time, since this provides a sense of whether the cloud system is growing or decaying over time.

As mentioned in Section 4.2, this movement is stored in the cloud motion graph. When the user selects a cloud of interest, we trace the evolution of that cloud as a set of nodes in the cloud motion graph. A

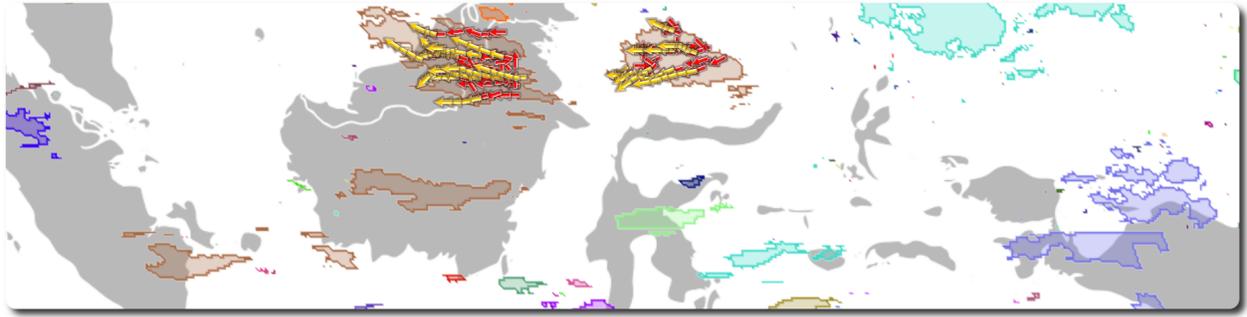


Fig. 7. Movement of Nakazawa cloud clusters over the Indonesian region during the period 2 January 2007 to 8 January 2007. The normal movement of the cloud cluster is to move in a westward direction. The local movement of two such clouds are shown.

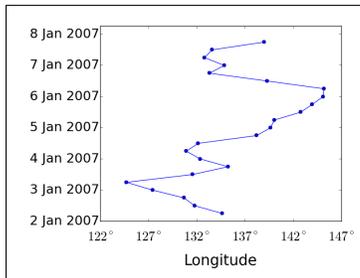


Fig. 8. The graph plots the longitude at different times over a period of 6 days. The plot implies a gradual eastward movement of the west moving cloud concentration, thus verifying the eastward propagations of the MJO.

uniform sample of the time steps are then selected. The clouds corresponding to the nodes in these time steps are then displayed so that the change of size over time can be visualized. In order to differentiate between the different time steps, the clouds are colour coded over time. Note that having a dense sampling of time steps could clutter the visualization. In addition to displaying the clouds, the general direction of movement is also displayed as a glyph. This direction is computed as the vector that transforms the center of the cloud system from one time step to the next. Figure 6(c) shows the long term movement of the green cloud in Figure 6(a).

6 USE CASE SCENARIOS

In this section, we demonstrate the utility of our framework through three use case scenarios.

6.1 Implementation and Experimental Setup

We first describe the implementation of our prototype software and the experimental setup. Experiments are performed on the CPC-merged dataset [1]. The data is available as a set of 62 MB binary files, where each file stores the data for the two 30 minute intervals within the corresponding hour. We first crop the data based on the given time period and the spatial range of interest. We then pre-process this input in order to support interactive exploration and visualization. Pre-processing includes computing and storing the split trees for the different time periods. We also compute the birth temperature, death temperature, and persistence of various clouds when computing the split tree. The optical flow associated with each time step is also stored while pre-processing the input.

Our software then uses the stored split trees and optical flow to compute the set of clouds and the cloud motion graph using the threshold specified by the user. The software allows the user to view the movement of clouds as an animation. Alternatively, the user can view a specific time instant. The software also allows the selection of clouds of interest in order to view either the local or long term movement of the clouds. The user can also specify the density of sampling and the density of time intervals that are used to generate the resultant visualizations. The visualizations are generated using the pre-computed data

structures, which are small compared to the entire input data. Also, given a time step of interest, we load only the data corresponding to neighbouring time steps into memory. This helps reduce the memory overhead and hence the software can handle very large time periods.

In our current implementation, querying of cloud systems is supported via a command interface. Querying is currently performed using a brute-force search. Due to this, queries over a large time range (typically greater than two weeks) requires time in the order of minutes to finish. Adding specialized data structures would potentially improve the querying time. In all the reported experiments, no filter corresponding to the size or lifetime of the clouds were applied.

6.2 Analysis of movement of the equatorial MJO over Indian Ocean

Nakazawa [34] used 3-hourly IR data and showed that within the intra-seasonal Madden Julian Oscillation there exists super cloud clusters of spatial size of thousands of kilometres moving eastward at $10-15 \text{ ms}^{-1}$ and had a time scale of less than 10 days. Embedded inside these were cloud clusters of 100 kms and time scale 1-2 days moving westwards. We could verify the existence of such cloud clusters over the Indonesian region using our framework. Figure 7 shows the local movement of a couple of selected clouds of the MJO on 2 January 2007, which shows a general westward movement of the cloud clusters.

The eastern movement of the super cloud clusters was speculated to be due to the successive cloud cluster formation east of an older one, leading to an overall eastward propagation. Since the actual clouds move westwards, identifying the eastward propagation of the westward moving clouds would normally require the user to manually analyse cloud movements over a longer time period. Also, such analysis would only provide qualitative verification of the phenomena. Using our query framework it is possible to obtain a quantitative verification of such phenomena as follows.

The idea behind the following experiment is to study the concentration of west moving clouds over a long time interval. We first query for the set of west moving cloud systems at regular intervals in time. We used an interval range of 6 hours in our experiment. We then use the geometric centre of all the clouds of the obtained cloud system to approximate the concentration of the west moving clouds.

Figure 8 plots the longitude of these centres at different times. Note that initially, the concentration of west moving clouds also moves westward. Then, as a new set of west moving clouds are formed to the east of existing cloud clusters, the cloud concentration begins to move eastward until the new cloud clusters are fully formed. As these newly formed clouds moves westward, the cloud concentration again moves to the west. The above process keeps repeating with every new cloud cluster that is formed to the east, which implies a gradual eastward motion of the MJO. The plot also indicates the time scale of the west moving cloud clusters, the time between the formation of one cloud cluster and the next, to be around 1-2 days.

While the expected movement of clouds within these cloud clusters is in a westward direction, it is also possible that there are exceptions, wherein there is an occasional short term eastward movement due to a convectively coupled kelvin wave [40]. Manually searching for such

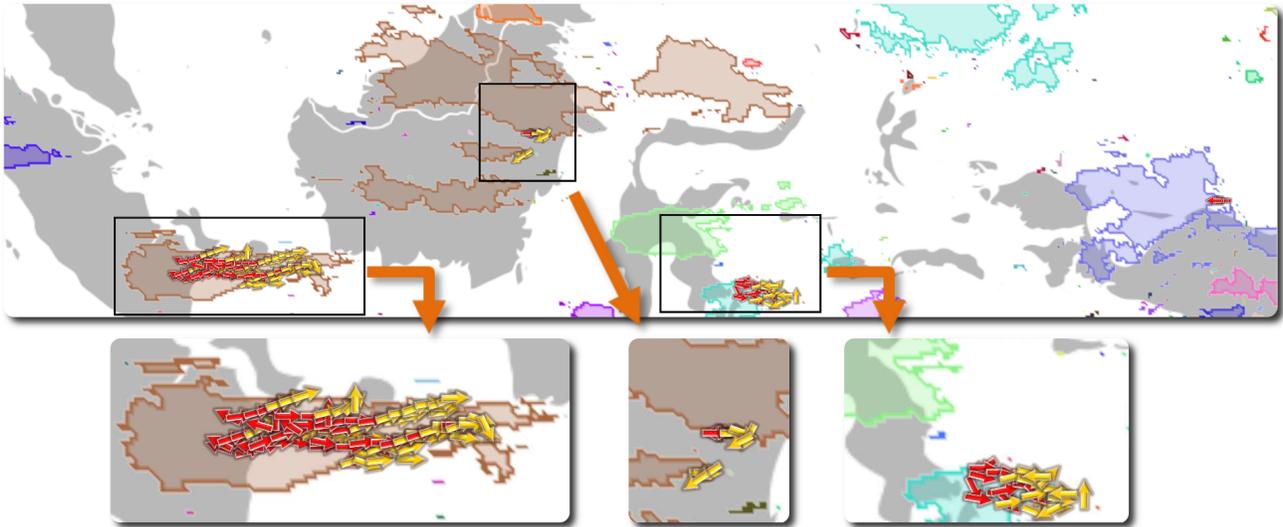


Fig. 9. Parts of the cloud cluster move in an eastward direction as opposed to the normal westward direction. Querying for such anomalies results in 4 such movements on 2 January. Note that the eastward movement is short term due to a convectively coupled kelvin. The clouds then resume their normal westward course.

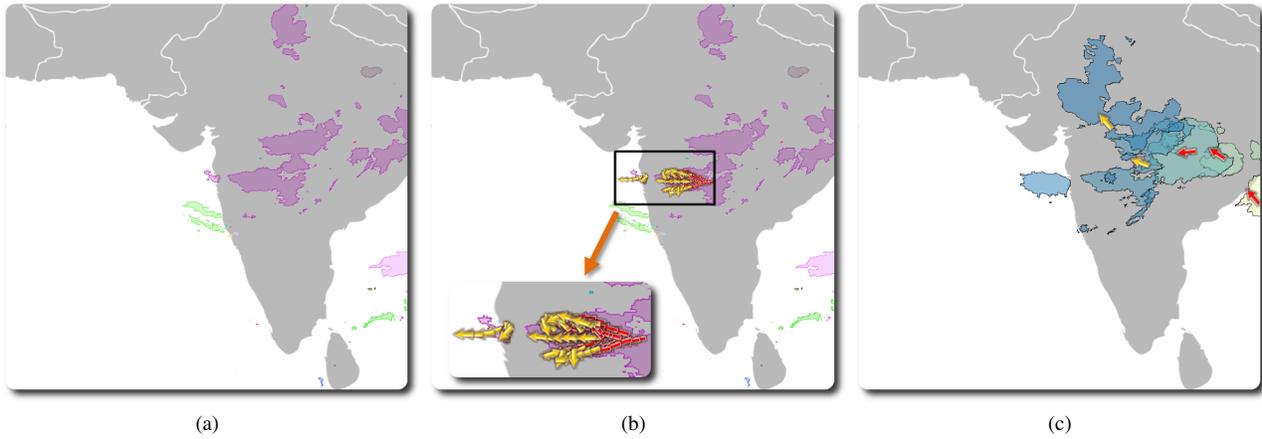


Fig. 10. Movement of clouds during the 2005 Mumbai rainfall. **(a)** Clouds at the beginning of the rainfall phenomena. **(b)** Local movement of the two clouds that was responsible for this event. **(c)** Long term movement of the large cloud shows that it moved in from east of West Bengal. Also, this cloud splits into two clouds, one merging with the smaller cloud, while the other moves in the north west direction.

possibilities is difficult. It is also possible that only parts of a large cloud moves eastward. In addition to the size of such east moving clouds being small, the duration of this eastward movement is short. It is therefore possible to miss such movement through manual analysis. Using our framework, we query the input for clouds that move eastward. The result of this query is shown in Figure 9, which shows the eastern moving clouds for one time step.

Our analysis indicates the asymmetry of the embedded cloud clusters. Westward moving clusters are more and of longer duration while eastward moving clusters are of shorter duration and fewer. Further it indicates that MJO are a multi-scale phenomena with each scale having a characteristic signature, that is, a larger scale envelope that gives the eastward moving MJO and shorter scale cloud clusters that move westward within the envelope. These short term westward moving cloud clusters in turn contain smaller clouds that move eastward. Our analysis also indicates that the movement of envelope is associated with an eastward movement of the position of genesis of westward moving cloud clusters. This insight has been easily obtained from our present analysis.

6.3 2005 Mumbai Rainfall

The heavy rainfall event at Mumbai (Bombay), India was of unprecedented intensity. It occurred on 26-27 July 2005. It was a highly lo-

calized event with Santa Cruz (Mumbai Airport) receiving 94.4 cm of rainfall within 24 hours. In contrast, Colaba in South Mumbai received a scant 7.3 cm during the same period. Most of the 94 cms fell within a few hours. If we examine the large-scale features we note that prior to this event, monsoon was in an inactive phase from 19-22 July 2005. On 23 July a low pressure area formed over Northern Bay of Bengal, intensified into a well-marked low, and moved westward [41].

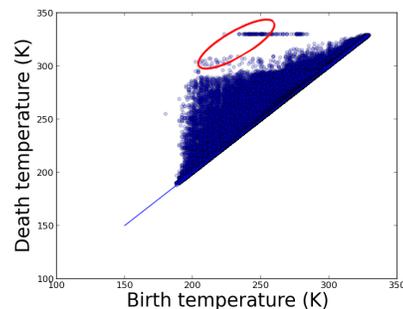


Fig. 11. Persistence diagram for Mumbai rainfall data. Note the presence of high persistence clouds formed at a higher threshold range.

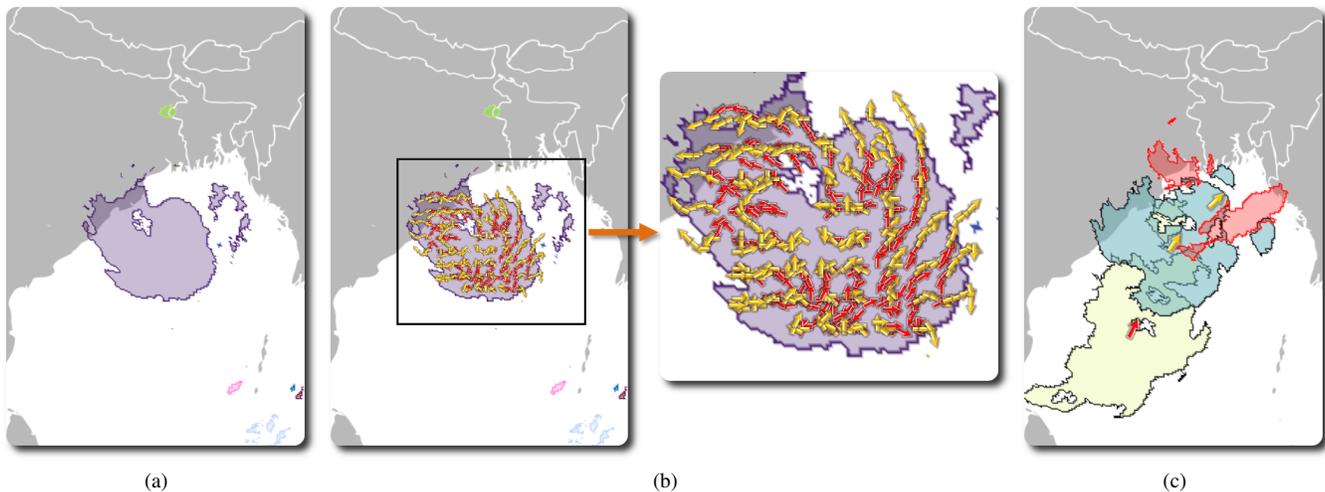


Fig. 12. Movement of cyclone Aila. **(a)** Clouds of the cyclone before landfall. **(b)** Cloud corresponding to the eye shows a spiral local movement. **(c)** Long term movement of the cloud corresponding to the eye of the cyclone. Note that after landfall, the clouds corresponding to the eye splits into two (highlighted in red), one moving in the north-eastern direction, while the other in the north-western direction.

Figure 10(a) shows the set of clouds during the start of the rainfall phenomenon over Mumbai. Note that there is a large cloud just east of Mumbai, while there is a small cloud directly over Mumbai. The local movement of these clouds (Figure 10(b)) show that the large cloud is moving from the west towards Mumbai. On the other hand, the smaller cloud was formed over Mumbai, and it moves west towards the Arabian sea. Looking at the long term movement in Figure 10(c) shows that the larger cloud has travelled across the breadth of India after originating in the region east of West Bengal. This cloud then splits into two, one part moving in the north west direction, while the other merges with the small cloud that has formed over Mumbai.

Figure 11 shows the persistence diagram corresponding to this input. Note that the presence of high persistence clouds that were created in the threshold range $[215K, 225K]$ suggests using a high threshold. We therefore use a threshold of $220K$. The validity of the range suggested was also verified using a lower threshold of $210K$. In this experiment, we couldn't capture the merger between the large west moving cloud and small cloud that was born over Mumbai.

The use of higher threshold to capture high rainfall around Mumbai could be related to the fact that quite a bit of rainfall are orography related. That is, when moist flow faces an obstruction such as the Western Ghats, copious rainfall can occur even with relatively shorter clouds. Mahani et al. [31] have also commented on the problems related close to regions of high orography.

The breaking of the MCS with a part going north-westward and another part moving towards Mumbai has not been documented in the previous study of Sahaney et al [41]. They used conventional methods of manually tracking low pressure systems (Figure 1 in their paper).

6.4 Cyclone Aila

One generally reads only about the movement of the entire cyclonic system. In this use case, we show the rich structure of cloud movements within a cyclone whose motion could be very different from that of the larger envelope, which in this case is the cyclone. Figure 12 shows the movement of clouds when cyclone Aila made landfall on May 2009. Figure 12(b) shows the local movement of the cloud corresponding to the eye of the cyclone, which indicates a spiral pattern. The long term movement, shown in Figure 12(c) of this cloud system corresponds to a movement towards West Bengal. Note that the cloud splits after landfall.

This again shows the occurrence of multi-scale interactions, the smaller cloud systems embedded in the larger scale cyclone with differing life cycles and direction of movement. While the splitting of the system is mentioned in the IMD report [3], no supporting satellite pictures are shown. Our framework easily tracks this splitting of

the system. Such a split in the cloud, which results in the split clouds moving away from each other, indicates that the system is likely to weaken. Such observations from real time data could again be used for now-casting.

7 CONCLUSIONS

In this paper, we proposed a framework to identify and explore cloud systems. It uses techniques from computational topology to efficiently identify clouds at different thresholds. The motion of clouds are then tracked using optical flow. Our framework allows for various queries on cloud systems that helps to easily and effectively analyse cloud systems without any knowledge of the underlying techniques.

Our framework clearly brings out that most tropical convective phenomena such as the Madden Julian Oscillation and tropical cyclone are multi-scale phenomena with smaller scale cloud systems embedded in the larger envelope. The movement of the envelope could be very different from that of the individual cloud systems. For MJO, our method clearly brings out that the envelope moves eastward due to the eastward movement of the genesis region of westward moving cloud clusters. Multiple cloud systems over a region could indicate severe weather while splitting of a convective system and their drifting away could indicate weakening of the larger system. These indicate that our framework has potential as a now-casting tool. We intend to release this framework for use by meteorologists and climate scientists to facilitate their study of cloud systems.

Currently, the cloud motion graph is used only for supporting queries. In future, we intend to use it as a user interface to interact with the visualization system. Additional visualizations of the calculated statistics such as distribution of cloud sizes could help guide users to explore the data. While queries on the input are processed quickly for smaller time intervals, this is not true when the data is available over large time periods. It will be interesting to consider indexing data structures to support interactive querying over large data sets. A web based interface for the framework that accesses real-time data will help complement existing now-casting tools.

ACKNOWLEDGMENTS

This work was partially supported in part by the National Science Foundation (CNS-1229185, CNS-1153503, IIS-0905385, IIS-1142013, and AGS 0835821), the Department of Energy, Office of Science Biological and Environmental Research, the Department of Science and Technology, India, under Grant SR/S3/EECE/0086/2012 and the DST Center for Mathematical Biology, IISc, under Grant SR/S4/MS:419/07. Ravi S. Nanjundiah thanks MoES/CTCZ for their support.

REFERENCES

- [1] CPC-merged IR brightness temperature data. http://mirador.gsfc.nasa.gov/collections/MERG_001.shtml.
- [2] Nasa trmm repository. <http://trmm.gsfc.nasa.gov/>.
- [3] Severe cyclonic storm, aila: A preliminary report. Technical report, India Meteorological Department, 2009.
- [4] J. Aggarwal and N. Nandhakumar. On the computation of motion from sequences of images—a review. *Proceedings of the IEEE*, 76(8):917–935, 1988.
- [5] S. Baker, D. Scharstein, J. Lewis, S. Roth, M. Black, and R. Szeliski. A database and evaluation methodology for optical flow. *Intl. of Computer Vision*, 92:1–31, 2011.
- [6] J. L. Barron, D. J. Fleet, and S. S. Beauchemin. Performance of optical flow techniques. *Intl. Journal of Computer Vision*, 12:43–77, 1994.
- [7] G. Bradski. The OpenCV Library. *Dr. Dobbs' Journal of Software Tools*, 2000.
- [8] P.-T. Bremer, G. Weber, V. Pascucci, M. Day, and J. Bell. Analyzing and tracking burning structures in lean premixed hydrogen flames. *IEEE Transactions on Visualization and Computer Graphics*, 16(2):248–260, Mar. 2010.
- [9] H. Carr, J. Snoeyink, and U. Axen. Computing contour trees in all dimensions. *Comput. Geom. Theory Appl.*, 24(2):75–94, 2003.
- [10] A. Chakraborty and R. S. Nanjundiah. Spacetime scales of northward propagation of convection during boreal summer. *Mon. Wea. Rev.*, 140:38573866, 2012.
- [11] Y.-J. Chiang, T. Lenz, X. Lu, and G. Rote. Simple and optimal output-sensitive construction of contour trees using monotone paths. *Comput. Geom. Theory Appl.*, 30(2):165–195, 2005.
- [12] D. Cohen-Steiner, H. Edelsbrunner, and J. Harer. Stability of persistence diagrams. In *Proc. Symp. Comput. Geom.*, pages 263–271, 2005.
- [13] H. Edelsbrunner and J. Harer. Persistent homology — a survey. In J. E. Goodman, J. Pach, and R. Pollack, editors, *Surveys on Discrete and Computational Geometry. Twenty Years Later*, pages 257–282. Amer. Math. Soc., Providence, Rhode Island, 2008. Contemporary Mathematics 453.
- [14] H. Edelsbrunner and J. Harer. *Computational Topology: An Introduction*. Amer. Math. Soc., Providence, Rhode Island, 2009.
- [15] H. Edelsbrunner, D. Letscher, and A. Zomorodian. Topological persistence and simplification. *Disc. Comput. Geom.*, 28(4):511–533, 2002.
- [16] G. Farneback. Two-frame motion estimation based on polynomial expansion. In *Proc. Scandinavian conf. Image analysis*, pages 363–370. Springer-Verlag, 2003.
- [17] T. Fiolleau and R. Roca. An algorithm for the detection and tracking of tropical mesoscale convective systems using infrared images from geostationary satellite. *IEEE Transactions on Geoscience and Remote Sensing*, PP(99):1–14, 2013.
- [18] D. Fleet and Y. Weiss. Optical flow estimation. In N. Paragios, Y. Chen, and O. Faugeras, editors, *Handbook of Mathematical Models in Computer Vision*, pages 237–257. Springer US, 2006.
- [19] A. V. Gambheer and G. S. Bhat. Life cycle characteristics of deep cloud systems over the indian region using insat-1b pixel data. *Mon. Wea. Rev.*, 128:4071–4083, 2000.
- [20] L. J. Gosink, J. C. Anderson, E. Wes Bethel, and K. I. Joy. Variable interactions in query-driven visualization. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1400–1407, 2007.
- [21] A. Hatcher. *Algebraic Topology*. Cambridge U. Press, New York, 2002.
- [22] B. K. P. Horn. *Robot Vision*. MIT Press, 1986.
- [23] B. K. P. Horn and B. G. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.
- [24] G. Huffman. Estimates of root-mean-square random error for finite samples of estimated precipitation. *J. Appl. Meteor.*, pages 1191–1201, 1997.
- [25] G. Huffman, R. Adler, B. Rudolph, U. Schneider, and P. Keehn. Global precipitation estimates based on a technique for combining satellite-based estimates, rain gauge analysis, and nwp model precipitation information. *J. Clim.*, 8:1284–1295, 1995.
- [26] J. Kasten, I. Hotz, B. Noack, and H.-C. Hege. Vortex merge graphs in two-dimensional unsteady flow fields. In M. Meyer and T. Weinkauff, editors, *EuroVis - Short Papers*, pages 1–5, Vienna, Austria, 2012. Eurographics Association.
- [27] F. Ladstädter, A. K. Steiner, B. C. Lackner, B. Pirscher, and G. Kirchengast. Exploration of climate data using interactive visualization. *J. Atmos. Oceanic Technol.*, 27:667679, 2010.
- [28] D. Laney, P. T. Bremer, A. Mascarenhas, P. Miller, and V. Pascucci. Understanding the structure of the turbulent mixing layer in hydrodynamic instabilities. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):1053–1060, Sept. 2006.
- [29] P. Lundblad, O. Eurenium, and T. Heldring. Interactive visualization of weather and ship data. In *Intl. Conf. on Information Visualisation*, pages 379–386, 2009.
- [30] S. Maadasamy, H. Doraiswamy, and V. Natarajan. A hybrid parallel algorithm for computing and tracking level set topology. In *Proc. Intl. Conf. High Performance Computing*, pages 12.1–12.10, 2012.
- [31] S. E. Mahani, X. Gao, S. Sorooshian, and B. Imam. Estimating cloud top height and spatial displacement from scan-synchronous GOES images using simplified IR-based stereoscopic analysis. *Journal of Geophysical Research*, 105:15597–15608, June 2000.
- [32] J. Milnor. *Morse Theory*. Princeton Univ. Press, New Jersey, 1963.
- [33] A. Mitiche and P. Bouthemy. Computation and analysis of image motion: A synopsis of current problems and methods. *Intl. Journal of Computer Vision*, 19:29–55, 1996.
- [34] T. Nakazawa. Tropical super clusters within intraseasonal variations over the western pacific. *J. Meteor. Soc. Japan*, 66:823839, 1988.
- [35] M. Otte and H.-H. Nagel. Optical flow estimation: Advances and comparisons. In J.-O. Eklundh, editor, *Computer Vision ECCV '94*, volume 800 of *Lecture Notes in Computer Science*, pages 49–60. 1994.
- [36] V. Pascucci and K. Cole-McLaughlin. Parallel computation of the topology of level sets. *Algorithmica*, 38(1):249–268, 2003.
- [37] V. Pascucci, G. Weber, J. Tierny, P.-T. Bremer, M. Day, and J. Bell. Interactive exploration and analysis of large-scale simulations using topology-based data segmentation. *IEEE Transactions on Visualization and Computer Graphics*, 17(9):1307–1324, 2011.
- [38] A. Piyush Shanker and R. S. Nanjundiah. Morlet wavelet analysis of tropical convection over space and time: Study of poleward propagations of intertropical convergence zone (ITCZ). *Geophys. Res. Lett.*, 31(2):L02103, 2004.
- [39] K. Riley, D. Ebert, C. Hansen, and J. Levit. Visually accurate multi-field weather visualization. In *IEEE Visualization*, pages 279–286, 2003.
- [40] P. Roundy. Analysis of convectively coupled kelvin waves in the indian ocean mjo. *Journal of the Atmospheric Sciences*, 1342:1342–1359, 2007.
- [41] S. Sahany, V. Venugopal, and R. S. Nanjundiah. The 26 july 2005 heavy rainfall event over mumbai: numerical modeling aspects. *Meteorology and Atmospheric Physics*, 109(3):115–128, 2010.
- [42] E. Santos, D. Koop, T. Maxwell, C. Doutriaux, T. Ellqvist, G. Potter, J. Freire, D. Williams, and C. T. Silva. Designing a provenance-based climate data analysis application. In *Proc. Intl. Conf. on Provenance and Annotation of Data and Processes*, pages 214–219, 2012.
- [43] E. Santos, J. Poco, Y. Wei, S. Liu, B. Cook, D. Williams, and C. Silva. Uv-cdat: Analyzing climate datasets from a user's perspective. *Computing in Science Engineering*, 15(1):94–103, 2013.
- [44] C. Stiller, J. Konrad, and R. Bosch. Estimating motion in image sequences - a tutorial on modeling and computation of 2d motion. *IEEE Signal Processing Magazine*, 16:70–91, 1999.
- [45] R. Szeliski. *Computer Vision: Algorithms and Applications*. Springer, 2010.
- [46] J. Weickert, A. Bruhn, T. Brox, and N. Papenberg. A survey on variational optic flow methods for small displacements. In O. Scherzer, editor, *Mathematical Models for Registration and Applications to Medical Imaging*, volume 10 of *Mathematics in industry*, pages 103–136. Springer Berlin Heidelberg, 2006.
- [47] M. Wheeler and G. N. Kiladis. Convectively coupled equatorial waves: Analysis of clouds and temperature in the wavenumberfrequency domain. *J. Atmos. Sci.*, 56:374399, 1999.
- [48] W. Widanagamaachchi, C. Christensen, P.-T. Bremer, and V. Pascucci. Interactive exploration of large-scale time-varying data using dynamic tracking graphs. In *IEEE Symp. on Large Data Analysis and Visualization*, pages 9–17, 2012.
- [49] K. Wu, W. Koenigler, J. Chen, and A. Shoshani. Using bitmap index for interactive exploration of large datasets. In *Proc. Intl. Conf. on Scientific and Statistical Database Management*, pages 65–74, 2003.
- [50] K. Wu, E. Otoo, and A. Shoshani. On the performance of bitmap indices for high cardinality attributes. In *Proc. Intl. Conf. on Very large data bases*, pages 24–35, 2004.