DSTVis: Towards Better Visual Analysis of UAVs' Spatio-temporal Data

Qiang Lu, Fengxin Chen, Liangliang Ni, Ye Yu

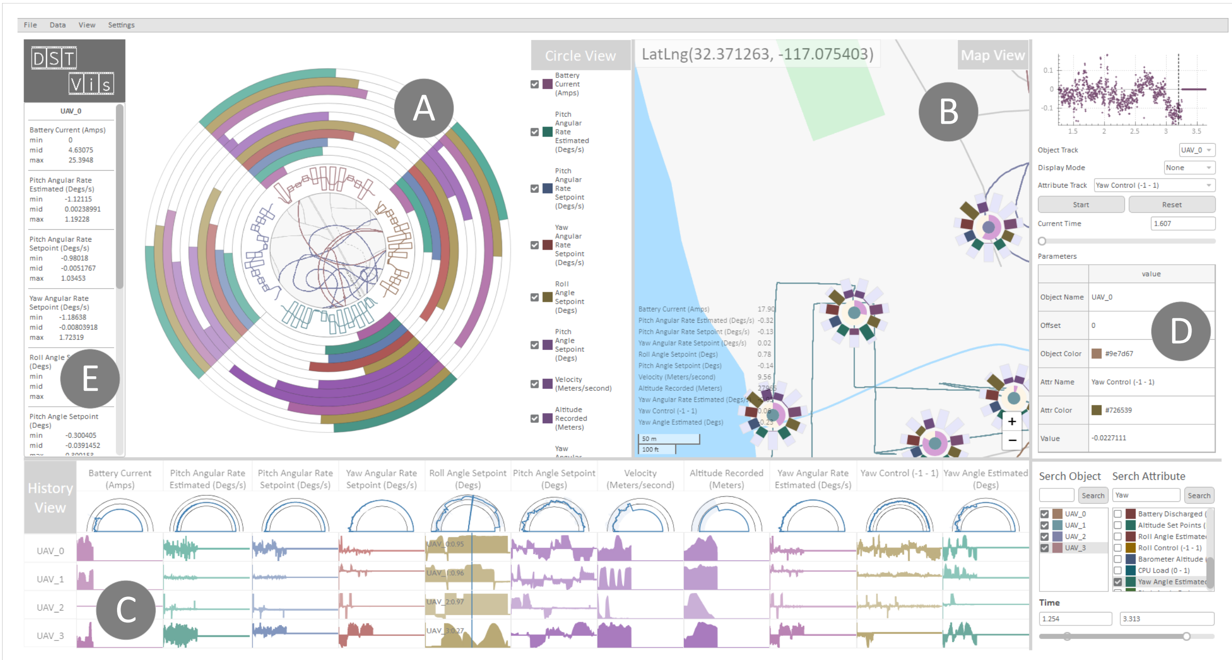


Fig. 1. DSTVis system. (A) Ring diagram for displaying and comparing real-time target multiple attributes and trajectory states. (B) highlighting target spatio-temporal properties at individual time points. (C) for retrospective comparison and analysis of historical data discrepancies. (D) Control the linkage between the three views. (E) Display Circle View parameters.

Abstract—Unmanned Air Vehicles (UAVs) for civilian have been widely applied in many fields. With the development of technology, the multi-factors spatio-temporal data associated with the UAV swarm has become massive and complex. Understanding these data and characteristics is of great significance to the development and operation of the UAVs. However, compared with other types, the multi-factors spatio-temporal data indicate the complexity and difficulty in analysis and visualization. Motivated by existing flight control platforms, we propose a visual analytics system called DSTVis. To develop DSTVis, we address three major challenges 1) real-time data visualization; 2) flight track monitoring; and 3) historical data analysis. For the first challenge, we assemble a set of effective visualizations called Circle View that can adequately display the large amount of data generated by the drone during flight. For the second challenge, we adopt a combination of time and space to support location navigation and data retrieval of single moment. For the last challenge, here a linkable control in the form of a matrix is used to analyze the performance of the target in the past time. We report on the design, implementation, and evaluation of the system and effectiveness of our system.

Index Terms—Visual analysis, spatio-temporal data, UAVs, interaction design

1 Introduction

As the market demand expands, civilian drone technology [37] has also made significant progress. According to the specific needs and working standards of various industries, UAV operation can reduce the cost of manpower and material resources [4, 9, 10, 35, 36], improve efficiency and quality, reduce environmental pollution and save time, which can replace the traditional operation mode of various industries. With the mutual integration of information technology such as big data, cloud computing, mobile Internet, and drone technology, drones have an increasingly good development prospect.

As far as we know, in the field of UAVs, less attention has been paid to the visualization of multi-dimensional Spatio-temporal data queries. After receiving the flight parameters, most ground stations present these as key-value pairs. It is tough to retrieve drone information despite vast amounts of data. In addition, the boring numbers are complex for the operator to form a subjective impression of the flight status of the UAV, which is not conducive to the operator's control of the UAV flight. In some cases, although visualization tools are used to analyze flight data, users usually need to switch between several additional tools to conduct a comprehensive analysis, which brings new problems to the users and is detrimental to visualization.

- *Qiang Lu and Ye Yu(corresponding author) are with the Key Laboratory of Knowledge Engineering with Big Data (Hefei University of Technology), Ministry of Education, Hefei University of Technology. E-mail: luqiang, yuye@hfut.edu.cn.*
- *Fengxin Chen and Liangliang Ni are with Hefei University of Technology. E-mail: 2021111103, 2018214911@mail.hfut.edu.cn.*

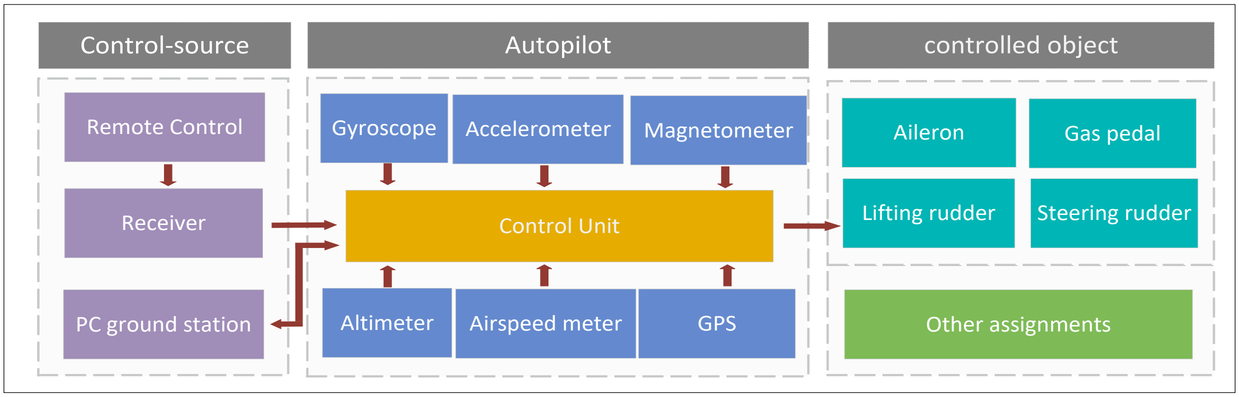


Fig. 2. Drone Control Center. The control unit is the most critical core equipment of the UAV. Its role is to transmit and process the control information from the remote control and ground station, synthesize and process the output information from sensors such as accelerometer, gyroscope, altimeter, airspeed meter, GPS module, etc., obtain flight data such as UAV position and attitude, and output the signal to control the UAV.

In addition, the retrieval of flight data is also a matter of concern in the actual use process. The multidimensional Spatio-temporal data visualization for UAVs should show the dynamic changes of flight data in real-time, which is the most fundamental requirement, but also should be able to record the data of the whole flight process locally or in the cloud. For example, a crew was using a drone for agricultural plant protection. However, the drone went off course during operation. This event could have been caused by many external or internal factors - wind speed and direction at high altitude, low battery power, and incorrectly set position information. Analyzing the drone's flight data over a while can the operator get a clear picture of what caused the event to occur.

Based on the limitations in the field of UAVs flight information, we propose a visual analysis system that allows operators to understand the dynamics of data during operation, the historical data of the UAVs, and analyze if there are any anomalies. Designing such a system poses three challenges:

Displaying massive amounts of UAVs' dynamics data. The information carried by the UAV flight process is related to its configuration. The control units of modern UAVs contain several sensors so that a large amount of spatio-temporal related data is generated during another UAV flight. The visual coding of this data should allow the user to understand how it changes over time. Like other fields, the visualization of multidimensional spatio-temporal data from UAVs suffers from excessive dimensionality and occlusion problems. Solving this problem requires designing a new type of visualization that enables it to display data from a temporal perspective.

Recording drone trajectories. It is important to encode the geospatial attributes of the data. Only in this way can we know the drone's coordinates when it generates data. Data is recorded in space and time, and where it occurs may be as important as when it occurs. It is also important to note that the field of drone flight is regulated in various countries, and the actual operation should also consider whether the drone route matches the preset one.

Retrace the historical flight information of the drone. The attributes should be displayed so that they can be easily compared in all dimensions (value, space, and time). The user should be able to select the attributes to compare using any of the visualizations provided. The amount of work involved in historical processing data is even more onerous for UAVs. The user must be able to adequately classify the individuals and attributes of the UAV fleet and be able to present historical information about the individuals to compare similarities and differences between them.

In this paper, we conclude five goals through field research. Based on these goals, we further developed DSTVis, a multidimensional Spatio-temporal data visualization and analysis system for UAVs, to solve the three challenges mentioned above. Specifically, for the first two challenges, we designed a linked dynamic graphical visualization for displaying dynamic information and trajectory information of UAVs in real-time. For the third challenge, we design a user interaction based on target categories that can be effectively applied to UAV Spatio-temporal datasets and helps to query and analyze UAVs' historical data. The main contributions of this paper can be summarized as follows.

- Design and implementation of an integrated visual analysis tool that includes real-time data display and historical data retrieval, and the system can effectively reduce visual confusion and improve the efficiency of information representation.
- Reveal the differences of data among different UAVs by user interaction, which helps further analysis and research.
- Evaluate DSTVis and our abstractions with summative results and confirm the usefulness of our system through case study and user study.

2 Relate Work

In the following subsections, we discuss the preliminary work that guided our research. We focus on the three most relevant topics: time-series data visualization, trajectory visualization, and user interaction interfaces.

The multi-coordinated view visualization approach is based on a reduced dimensional approach. The time axis is the mainline, and all time-related attributes are quantified. There will be multiple individual views to display multidimensional information, and the user could link these views together for coordinated analysis. [19]. Dortmont et al. [32] display time series and event series by sharing a horizontal time axis, where the event series is represented as a series of graphical symbols. The event sequences are simple line graphs. When loading the dataset, the user initially sees the events in the individual event sequences and a graph for each available time series. Takanori et al. [7] treat time-dependent multivariate data as a whole. They use dimensionality reduction first and visualization later for time-series related data. Zhang et al. [40] analyze the data analysis protocol by visualization instrument module to display the flight data directly. Existing studies have less application for multi-UAV timing data. We propose a novel visualization view that helps analysts get accurate real-time information about UAVs.

For the spatial data level of UAVs, we focus more attention on trajectories. To reduce the visual clutter of large-scale trajectories, researchers use aggregation methods, such as K-means clustering [29], to group trajectories together. Syver et al. [31] use vessel glyphs to represent fishing boats and analyze the status of fishing boats in the form of trajectories displayed according to zoom levels. Andrienko et al. [3] classify trajectory visualization techniques into three main categories: direct visualization, aggregation visualization, and feature visualization, such as in the form of displaying trajectories on a timeline,

Table 1. Task Analysis

Category	Design	Task Overall Goals
Data Visualization	T1: To show the various attributes of the real-time data.	G1 - G3
	T2: To display the temporal distribution of data.	G1 - G3
	T3: To support displaying all data during a run.	G4, G5
Categorization & Comparison	T4: To display the frequency of target occurrence in the range time.	G3, G4
	T5: To compare the properties of different targets at the same time.	G4, G5
Data Abstraction	T6: To support understanding the peak and trough values of targets in the range time.	G1, G4
Interactive Feature Specification	T7: To support selecting the attributes to be displayed.	G1 - G5
	T8: To support selecting the target to be displayed.	G1 - G5
	T9: To support adjusting the display visual effect.	G1 - G5

using trajectories as animations to represent the movement of objects, or displaying spatial and temporal information together in a Spatio-temporal information cube. These three types of techniques have been applied to sparse Spatio-temporal data visualization. These methods have been of great help to people in their research.

Multivariate visualization is the commonly used method for visualizing multidimensional Spatio-temporal data. It focuses more on encoding spatial, temporal, and other attributes simultaneously in a more compact way in one view. Vanessa et al. [27] present a set of design guidelines about geo-temporal visualization techniques for communicating correlation. Chen et al. [33] devised a Spatio-temporal visual representation of team formation changes that allows analysts to visually analyze the evolution of the formation and track the spatial flow of players within the formation over time. They further developed a visual analysis system, ForVizor, that enables users to track Spatio-temporal changes in information. Kim et al. [15] and Li et al. [17] provided trend visualization models to present spatial changes on a two-dimensional map. Juraj et al. [26] connect time and space in a scaled Spatio-temporal cube based on partial domain aggregation and model the cube, thus achieving fast identification of Spatio-temporal patterns at multiple scales. Guo et al. [8] investigated the simplicial spacetime meshing scheme. Zhang et al. [39] developed a multivariate temporal structure that can express temporal information at a location in multidimensional space. We have done an in-depth study at this level and proposed a trajectory visualization method that can carry multidimensional attributes.

There is a generous design space for visualization techniques representing multi-attribute data [13], and direct user manipulation of reduced-dimensional model parameters is a typical interaction method in terms of introducing interactions and dynamic displays. Sabando et al. [28] allow sorting and filtering to display detailed information about compounds. Alexandra Lee et al. [16] design time ranges where the user can define any size or order. Kim et al. [14] correctly interpret, define and change axes in a user-driven manner. Users can define and modify axes by dragging data items to either side of the x- or y-axis. The system will calculate linear combinations of data attributes based on the x- or y-axis and bind them to the axes. Zong et al. [42] designed user-click colour and filter applications to highlight selected points interactively. Dylan et al. [5] used a column view where users can view details of each attribute as needed. We are inspired by sportfolio [38] to have a new visualization view for multidimensional Spatio-temporal data.

3 Background And Task Analysis

In this section, we summarize a set of analysis tasks that guide the system design based on the initial intention of the system design.

3.1 Background

Modern drones are used in various scenarios mostly in the form of drone swarms. UAV swarms based on GPS positioning are composed of flight control (Fig. 2), onboard computers and communication modules in hardware, cluster communication, data interoperability and mutual control or other functions in software using communication links [1, 2, 21], etc. Their sophisticated design not only brings complexity to the hardware structure but also poses many challenges to the software design. During the operation of the UAV cluster, the ground control station will get much information related to the flight of the UAV cluster from the airborne equipment, such as position, altitude, speed, heading, etc. These are all data that are associated with time and space. Our design needs to visualize the data from three levels - real-time data, historical data, and target trajectory to meet users' needs.

3.2 Task Abstraction

We conducted a comprehensive literature review of 30 papers collected in the field of data visualization, the field of UAVs, an interview with an expert in the field who studies multidimensional spatio-temporal data visualization, and an exchange with two UAV flight enthusiasts. We compiled a list of analysis tasks based on the actual requirements and the original design intention.

G1: To understand real-time latitude, longitude, altitude, speed, and heading information of multiple groups of UAVs. Being able to observe and learn the dynamic information of UAVs in real time is a fundamental requirement for users to operate at the gimbal, and it is crucial for users to report the real-time status of multiple UAVs in a clear and intuitive manner.

G2: Understanding the spatial and temporal distribution of multiple UAVs in the current environment. Field research studies have shown that it is important to present the dynamics of latitude and longitude numerically, and it is also important to present the changes in geographic location in the temporal dimension on a 2D map, and it is more helpful for the operator to better control the UAVs.

G3: Understanding detailed information about a single target at a single moment in time. In the preliminary interviews, users wanted to be able to observe not only the dynamics of clusters of drones, but also to be informed of specific properties of specific targets.

G4: Understand historical data for multiple groups of targets over a range of time. Being able to retrace the historical data of a target facilitates the analysis of whether the target has behaved abnormally in the past time. The large amount of real-time data can affect visual observations, and analyzing historical data can somewhat mitigate the negative impact it causes.

G5: Compare the flight status of different UAVs. Observing the similarities and differences in flight states between different UAVs can be linked to whether they produce anomalous behavior.

4 SYSTEM ARCHITECTURE

Our development starts with the analysis of the collected data set. To demonstrate the validity of our platform, we choose typical examples from real-world application scenarios. We are using data from flight logs generated by UAVs using the PX4 autopilot flight control software [6, 23], an advanced autopilot system matching the hardware of aircraft such as the pixhawk, developed by researchers at ETH Zurich. Its predecessor is APM and improves on APM by using a 32-bit ARM processor. In addition to UAVs, PX4 drives can be used for UAV carts, ships, and all kinds of autopilot-like devices. We then divided the functional requirements of the visualization view based on our

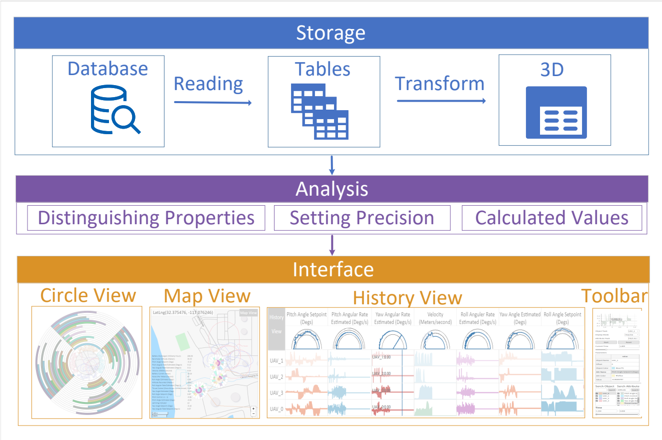


Fig. 3. The DSTVis process consists of three main components: a data manager, a real-time data visualization viewer, and a historical data visualization viewer. The raw data is pre-processed and stored in an application-specific data index for use by data-driven sources. The real-time data visualization browser is designed to use real-time data to display multiple targets, and their attribute values detected at the current moment and to depict the trajectory state on a two-dimensional map. The Historical Data Visualization Viewer performs historical data retrieval and comparative analysis.

design goals and design tasks and implemented the design requirements through complete data abstraction and system architecture. The visualizations and datasets used in this paper are available through https://github.com/VIMLab-hfut/DSTVis.

4.1 Data Abstraction

Despite little academic attention, visualization techniques have been widely adopted for the development and operation of autonomous aerial vehicles [11, 20, 25]. Here we discuss only extant tools for post-flight analysis of autonomous UAVs. Among many popular systems for autonomous UAV development, the open-source PX4 [12] stands out. Based on the analysis of the UAV flight control system, we can be informed that the data generated by the existing civil UAV flight process may contain multiple attributes, such as longitude/latitude coordinates, pitch, roll, yaw, altitude, speed, and so on. The data is recorded in time and space, giving each datum geo-spatial temporal attribute. These attributes constitute a three-dimensional table. Using the data taxonomy presented by Munzner [24], we start by parsing the drone flight logs into a CSV file. The Tables are referred to in the PX4 documentation as "system messages". Each system message contains data related to a specific system within the UAV. For example, the system message "vehicle_ gps_ position" will contain the data recorded or used by the UAV's GPS. We abstract these tables and define a dataset D, which is the collection of all data generated during the drone's flight. All data in the flight log is temporal, making our first-dimension time. We define it as T, and the frequency of recording flight data varies with different flight control systems, so T is the set of data receiving moments during the entire flight process. The dataset has z items, corresponding to z moments, each with multiple targets and their attributes. The dataset D are described as:

$$D = \{d_1, d_2, d_3, \ldots, d_z\}$$
$$T = \{t_1, t_2, t_3, \ldots, t_z\}$$

D and T for a full single shot.

$$d_z = [\{a_{11}, a_{12}, \ldots, a_{1m}\}, \{ a_{21}, a_{22}, \ldots, a_{2m} \}, \{ a_{31}, a_{32}, \ldots, a_{3m} \}, \ldots, \{ a_{n1}, a_{n2}, \ldots, a_{nm} \}]$$

Each moment t_z corresponds to a data item d_z, and each data item contains n targets$\{Ob_1, Ob_2, Ob_3, \ldots, Ob_3\}$. We define each target as a set $a_1, a_2, ..., a_m$ containing m attributes. In this case, all data in the flight log are temporary and each message contains a timestamp. The frequency of recorded data varies depending on the system from which the messages come. In the same UAV, the gyroscope sensor will record data more frequently than the sensor collecting temperature information, which makes our time the first dimension i.e. T. The second dimension consists of the attributes contained in each message, messages with spatial information will contain data corresponding to spatial position (latitude, longitude) as well as relative position (x, y) and orientation (pitch, roll, yaw) D. d_z as the multivariate data at the current moment constitutes the real-time data, $a_1, a_2, ..., a_m$ will be displayed as multivariate attributes for a single target at a single moment in time, and D containing all the data will be used to retrace the historical data.

4.2 System Abstraction

DSTVis is a Qt-based application. There are three main modules: the database module, data service module, and visual analysis module. We take advantage of the high portability of qt to run our program on platforms such as Windows/Linux. Inspired by the above analysis tasks, we designed DSTVis to allow users to explore and compare targets at different scales from different perspectives. The system consists of a real-time data module and a trajectory visualization module historical data module. Each of the above modules corresponds to three main views [22].

- A circular view for displaying and comparing real-time target multiple attributes and trajectory states.
- Highlight the target attributes and locations at a certain moment.
- Used for retrospective comparison and analysis of historical data discrepancies.

The system consists of interactive information visualization tools. Circle View provides an overview of the exploration process by automatically capturing the visualization state, and the user can see how this data is encoded changes over time. Map View shows the drone's coordinates when it generates data. The map view shows the user coordinates of data when generated. It shows not only the spatial properties of the drone but also the temporal properties. Users in HistoryTracking can determine how changes to one or more attributes viewed within a specific time frame led to changes in other attributes. A large amount of historical data is displayed regularly. Comparative analysis can be done between historical data.

5 VISUALIZATION DESIGN

DSTVis is an open-source visualization tool for supporting multidimensional Spatio-temporal data. The tool uses a C++-based back-end program to parse the data. The front-end is developed and specified by d3.js. DSTVis aims to enable users to analyze multidimensional Spatio-temporal data efficiently to ensure that the current state and historical state are accessible for multidimensional spatio-temporal data. DSTVis consists of three parts: Circle View, Map View, and History View [18, 34, 41].

5.1 Circle View

Circle View is designed to help users clearly understand the attributes of the current drone, including speed, heading, distance, etc. (T1, T3), and also supports users in comparing the attributes between drones (T5), view peak, and trough values (T6).

Fan ring. In order to avoid serious line occlusion problems when general radar maps, parallel coordinate systems, etc. contain huge amounts of data, we show some attributes of the UAV in the form of a sector ring (T1). In the following, we refer to the (Fig. 4(a)) circle cut by the circle with the smallest radius and the circle with the largest radius as the "main sector ring" and the (Fig. 4(b)) circle cut by the adjacent circle as the "subsector ring". Subsector rings with the same central angle respectively represent different attributes of the same UAV, and different central angles of subsector rings indicate

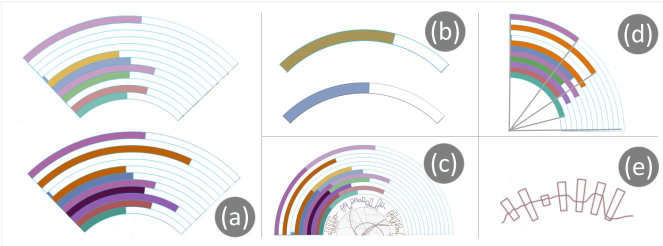


Fig. 4. A portion of the Circle View. (a) Main sector ring. (b) Subsector ring. (c) Merged main sector ring. (d) Describes how to fill the subsector ring. (e) Simplified version of the Box plot.

different UAVs. Different subsector rings with the same minimum and maximum radius indicate different attributes. Different colours on the sector ring correspond to different attributes, and for different UAVs, we use different depths of the same colour. The colour filling of the subsector ring is related to the attribute value. The maximum value that can be represented by a fully filled subsector ring is adaptive according to the maximum value of the drone attribute, while the unfilled state of the subsector ring is adaptive according to the minimum value of the drone attribute. In the case where the current value c is less than Max, the degree of filling of the colour to the subsector ring, etc. can be calculated as $c/(Max-Min)*100\%$. If the current value is equal to Max, the degree of filling of the sector ring is 100%. When encoding the values, we associate the maximum value represented by the maximum area of a subsector ring with the angle of the central point of the circle occupied by that subsector ring. Assuming that $c<Max$ and also that this subsector ring has a circular centroid $alpha$, then the value of the corresponding circular centroid of the filled area is $(c-Min)/Max*\alpha$ (Fig. 4(d)).

Comparison. We compare the attribute values between targets by merging the main sector rings. The superposition of attributes between two targets is achieved by rotating the position of the primary sector ring to the other main sector ring. The superimposed view is a stacked histogram in polar coordinates. In this way, we have two options for comparing the attributes of the UAVs. One, we can move the mouse over the stacked chart, and the view will automatically display the attribute information of all targets contained in the current stacked bar chart. Second, we can compare the size of the centroids of the two sub-sector rings representing the target attributes. This allows us to analyze whether the flight status of the UAV produces anomalies. We have tried to use the same main sector ring to represent a certain attribute and each sub-sector ring on the main sector ring to represent a different drone so that we can compare the size of the value of each drone in this attribute by the size of the rounded corners corresponding to the coloured area above. However, it turns out that the area of the coloured areas of the subsector rings close to the outer ring may be much larger than that of the subsector rings close to the inner ring, even though their values correspond to smaller circular angles. Its visual bias affects our judgment of the values. As a result, we now have a design and display that clearly conveys information.

Peaks and valleys. To better understand the peaks and valleys of an attribute (T8), we use a simplified Box plot to represent the variability of the same parent data within a specific time end. The simplified Box plot removes the upper and lower quartiles, which consists of a rectangle and a line segment dividing the rectangle in order to focus on showing the maximum and minimum values of an attribute over a specific period, and the number axis used to show the median shows the mean in our design. Different UAVs correspond to boxes of different colours. In addition, to better compare the average, a Bessel curve is used to connect the points where the medians are located. In addition, we show the parameters of the simplified Box plot selected by the mouse on the left side of the view.

Most of the UAV data are multidimensional time-series data for which various visualization methods have been proposed, such as dimensionality reduction, scatter plot matrix, parallel coordinate plots (PCP) and graphical symbols. The most serious difficulty in designing this type of view is data masking, and we aim to show the details more clearly. We have solved the problem of target and attribute dimensions in the form of a fan ring in the design. In addition, the user can understand and analyze the unique values with a simplified Box plot, with the parameter panel on the left side as a reference.

5.2 Map View

Map view uses leaflet's open-source map to mark the geographic location of the drone. There are two options here: choose to show the path from the start moment to the current moment and scroll through the updates; choose the path for the entire flight. Users clicking on the path in Map View will form a style showing multiple attributes of the drone at the clicked location.

Trajectory. The view is designed with two modes, mode-1 and mode-2. Mode-1 shows the path generated by the drone from the start moment to the current moment, and the geographic location of the current moment will be represented by a point (Fig. 5(b)). Thus, when mode-1 is selected, the point will keep evolving the path as time keeps advancing. Our control panel sets up a progress bar to control the time change. Clicking the "Start" button will move the time backwards from the initial moment, clicking the "Pause" button will pause, and clicking the "Reset" button will return to the initial moment. In addition, we set different colours for the paths of different drones, and the colour set corresponds to the colour of the main sector ring representing the drone in Circle View. Mode-2 shows the path of the whole flight process. When Mode-2 is selected, the geographic location of the current moment we distinguished with a bolded dot. As in Mode-1, the position of this bolded dot changes over time when the time starts to change.

Details. The map in the Circle View (Fig. 5(a)) is used to overview all the trajectories, and when we click the map in the Circle view, it will automatically locate the corresponding position in Map View. We set the corresponding style on the path of the Map View to know the information about the attributes contained in the drone at a specific moment (Fig. 5(d), (e)). We can observe that the style consists of three parts, one is the outermost bar graph, each rectangle of the bar graph represents an attribute of the drone, and the degree of filling of the rectangle symbolizes the size of the attribute's value. The second is the circle connected to the bar graph, which serves as a circular progress bar. It represents the relationship between the current moment and the total duration. For example, if the selected period contains data for n time points and t $\in \{t_1, t_2, t_3, ..., t_n\}$, the circle will be divided into n copies of the same size sector ring, and the selected coordinate point belongs to the t_z moment, the circle will fill z copies clockwise from the axis where $\pi/2$ is located. It can be seen that the rings belonging to the exact moment of the UAV data are filled to the same extent. In this case, we can distinguish whether different drones belong to the exact moment by the degree of filling of the circles used. In addition, as we click on a certain point on the map, the geographical location of the point (including latitude and longitude) is displayed at the top left of the view, and other attributes of the point are displayed at the bottom left of the view. Mouse-hovering over the style will have the same effect.

Displaying drone trajectories is a very important thing for drone control. In practical applications, it is necessary for the user to observe information about a specific drone at a specific moment. It will cause a significant loss of system resources if a lot of data is displayed on the view. We use a zoomed-in format, where the original map is only used to display the track information, and the auxiliary view floats above the track points to view the spatio-temporal attributes of a particular drone at a certain moment, which satisfies the user's needs and reduces the system overhead.

5.3 History View

The history view uses area charts, stacked line charts and semicircular radar charts for historical data analysis and comparison (T3, T4). Here is an $m*(n+1)$ table, with m representing the number of currently selected attributes and n representing the number of currently selected

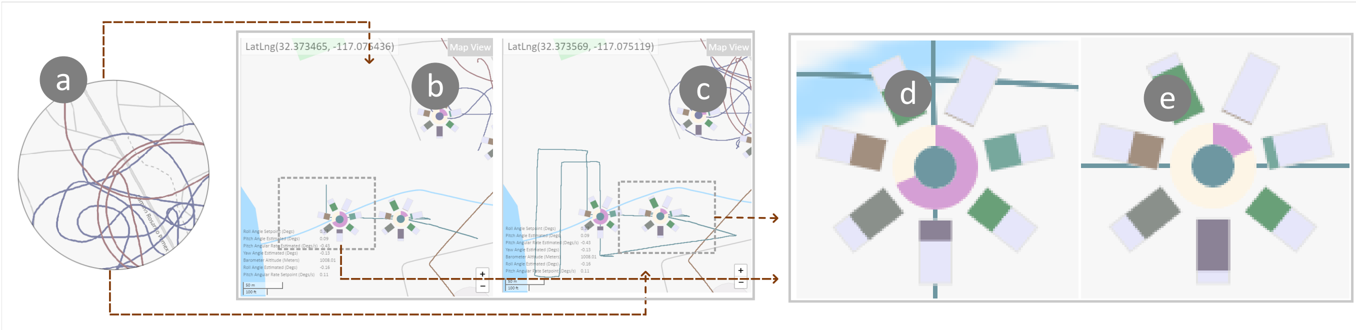
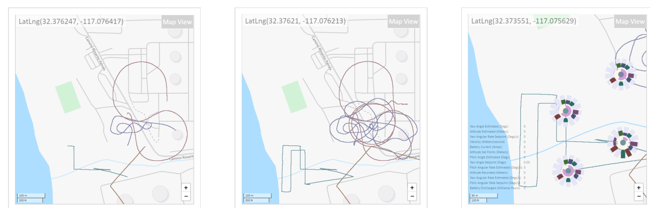


Fig. 5. The linked view of Circle View and Map View. (a) The central view of Circle View. (b) The representation style of the path and a point on the path of mode-1. (c) The representation style of the path and a point on the path of mode-2. (d) The filling method of the circle when has experienced a shorter duration. (e) The filling method of the circle when has experienced a longer duration.

targets. Thus, each table row represents a target, and each column from column 2 onwards represents an attribute. The first column of the table (Fig. a) uses a semicircular radar plot with a polar coordinate system. The angle indicates the moment, and the radius indicates the average of all targets for the current attribute. The area map is generated based on the period selected by the toolbar. Starting from the second column, each table item corresponds to the magnitude of the value of an attribute of the current target for the selected period. We use progressive colours to indicate different targets under the same attribute, with different colours for different attributes. In addition, the display of attribute values between targets can be merged by dragging the target names. The merged view is a stacked line graph, and we can know the size of the attributes of different targets at the exact moment by the change of the line graph.



(a) Mode-1. (b) Mode-2. (c) Style.

Fig. 6. (a) and (b) are different modes of changing Map View. (c) is the style of the surfacing of the click track points.

5.4 User Interaction

The following interactions are included in the user interaction system.

Parameter settings. First, for the timing chart on the system control panel, we can choose to display a specific property of a sure drone in the drop-down menu of the control panel. In addition, we can set the form of the timing chart, including line chart, area chart, scatter chart, etc. Secondly, when we control the number of attributes displayed by the system, we can search for the corresponding attribute in the "Search" button, automatically locating the multi-select box for that attribute, which solves the problem of difficult selection due to the number of attributes. The map in Map View can also be zoomed in and out with the mouse wheel. Users can also adjust the frequency of track point changes in Map View by the menu bar. It is an innovative work.

Mouse interaction. As required by task T5, we designed a mouse interaction method that allows attribute values to be merged into the same round corner by dragging the main sector ring and stacking it with the main sector rings of other round corners, with the option to split the merged main sector ring (T9). Hovering the mouse over the sector ring displays information about the current attribute. After the user clicks to select a track point in the circle view, the details of the coordinate point and the coordinate points in its region can be displayed in the track view. When the user hovers the mouse over the radar plot, the attribute value for the moment represented by the angle where the mouse is located is highlighted. The specific parameter is displayed on the left side. Likewise, the area map in the same column will highlight the value for that moment.

6 Evaluation

6.1 Case Study

We use the flight logs of open source projects [30] as to the source of our real dataset. To demonstrate the generality of our system, we have analyzed the ulg format file and used the analysis method in Section 4.1 to expand the existing data set with data expansion. In addition, we filtered 29 flight attributes related to flight control by data cleaning. For example, attributions with spatial information will contain data for a spatial position (lat, lon) as well as relative position (x, y) and orientation (pitch, roll, yaw). Finally, we obtained flight data for four targets over a flight period. In this experiment, we will focus our analysis on this dataset.

To determine if the DSTVis system can be used to analyze and answer UAV flight questions, we searched GitHub and the UAV Exchange forum for keywords. We reviewed related problems and suggestions encountered by users. One of the users described in PX4-Autopilot, "We were flying a mission with our coaxial octocopter, v1.10.1. I have been looking the reason at the logs but did not found anything yet". We can find that it is essential to analyze the path changes during the flight of the UAV and to see the spatial information carried by the points on the path. DSTVis provides an interactive operation to view the changes in the target UAV's motion trajectory over time. As shown in Figure. 6(a), the target UAV changes as a point throughout the flight from the start moment to 2.039 s. Figure. 6(b) represents the changes of the target UAV in the form of generated trajectories over the period from the start moment to 2.039 s.

We know from the existing literature that one of the suitable methods for visualizing UAV flight data is to encode the flight trajectory. It can make the semantic information of the flight trajectory in the view richer by colouring the trajectory in the view in time or space. However, encoding the paths with colours is mainly done to distinguish the targets to which the trajectory belongs or by reflecting the change in the magnitude of the target's value in some attributes utilizing gradient lines. We even found that this colour coding approach can effectively encode attributes that are not directly related to the trajectory, such as some binary data, categorical data (e.g., flight patterns), or divergent data (e.g., actuator positions). Although this approach is potent, few attributes are observable at the exact moment in the user-observed view. In our design approach, you can see that both samples have a style that shows all spatial properties of the current position anywhere on the path (Fig. 6(c)). Of course, we can set the number of attributes displayed via the toolbar. We can observe from the sample that the flight speed of UAV3 is 22.643 m/s at 2.039 s. This implementation is beneficial for us to see the spatial and temporal information at any moment of the

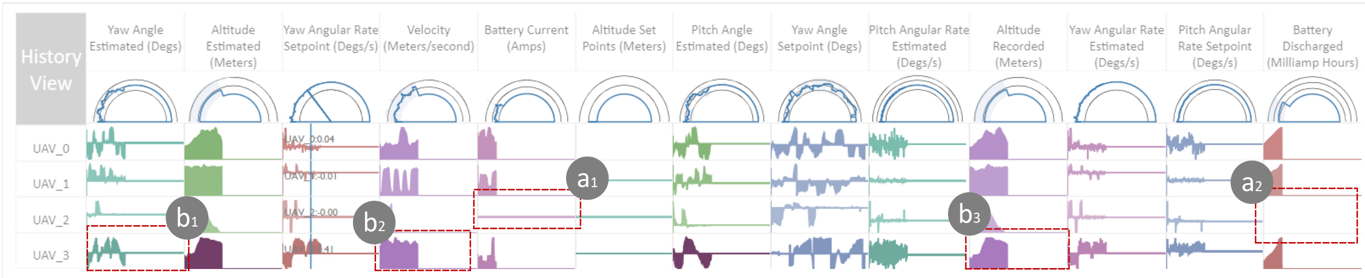


Fig. 7. The area map is arranged by means of a matrix, where each element of the matrix represents a certain attribute of a certain UAV. The figure contains a total of four drones, UAV0, UAV1, UAV2 and UAV3, as well as a selection of 14 of these attributes to display. a_1, a_2 are the current battery and battery discharged of UAV2. b_1, b_2, b_3 are the altitude estimated, velocity and altitude recorded of UAV3.

Fig. 8. Circle View. (a) Control the Circle View displays 27 attributes. (b) Mouse-hovering over the view of UAV0. (c) Mouse-hovering over the view of UAV3.

flight. The dynamic path generation method based on time change can trace the flight state of UAVs more clearly. In addition, we have linked the flight attributes to variable views (among them, views that include area, line and scatter plots, etc.), in which the variable view can show the temporal changes of a given drone and its attributes, helping users to see the attributes they want to see in the space or period they are viewing. In response to the question raised by the previous forum user, we analyzed and guessed that the logger was wrong and the disarm occurred after the landing state.

We can learn from the graph that the pitch angle of UAV2 is positive for a more significant period, and the pitch angle is also larger throughout the flight time. The speed of UAV2 is faster when the pitch angle is positive. Based on the values of pitch angle and speed change, it is inferred that the flight altitude of UAV2 is probably the highest. We then observe the whole flight process, the maximum speed of UAV3 is the immense value among the four UAVs, and UAV2 flies at the highest altitude, so we can see that our analysis is in line with the actual results. Second, we see that the gradient of the yaw angle of UAV3 is changing more, guessing that UAV3 should be in a state of constantly changing direction. Combined with Map View, we can learn that UAV2 and UAV3 are flying in a fixed area, and UAV3 is hovering during the flight, which is consistent with the analysis. Third, by observing the battery current and remaining battery power of UAV2, we found that UAV2 lost data during the selected period. However, the power attributes of other UAVs could still be displayed usually, and we guessed that the firmware of UAV2 returning battery attributes might have malfunctioned. Fourth, by observing the predicted values of the flight attributes (e.g., altitude) of the four groups of UAV firmware compared with the actual recorded values, we found that the predicted values did not differ much from the actual recorded values. It indicates that these sensors containing the predicted information were working correctly.

We set the control panel to display 27 attributes about the drone (Fig. 8(a)). We can see much information. Although we set many attributes to be displayed, the view is still in an appropriate viewing state by constantly adjusting the width of the subsector ring and adapting to the window size. It is easy to see that the flight speed of UAV0 in the state shown is about 19.507 m/s (Fig. 8(b)) and the estimated pitch angular rate of UAV3 is approximately equal to - 0.22 degs/s(Fig. 8(c)).

6.2 User Study

Evaluation Process. This section describes the evaluation of the system. We first invited several users and conducted a training session to introduce the system. We ensured that the users understood the system, including the visual coding and user interaction. Afterwards, users freely used our system to analyze datasets, and we interviewed users one-on-one to collect their feedback after the case study.

We first introduced the purpose and functions of the system to the users. Second, before participants experienced DSTVis, we asked users to evaluate their aesthetic, drone knowledge, and visualization domain knowledge and instructed them to complete a user information questionnaire. Next, they were instructed to approach the system, use it to complete a series of tasks, and complete a user experiment questionnaire. Finally, they completed user interviews and a post-survey.

User Tasks. The user tasks focused more on guiding participants to understand better the functionality of the overall visualization system to analyze the data. We recruited 20 participants (3 females, aged 20 to 27 years (mean = 23.2)) with normal or corrected-to-normal vision. All participants were students in the computer science department of our local university. All participants had experience with computers, and twelve of them had developed software systems independently. None of the participants had expert visualization skills. Almost all of them had four years of experience with computer software. Three of them had been involved in visualization-related software projects. We designed seven user tasks for the visualization and system technique, detailed in the accessories. We designed the task to cover answering questions by direct inspection and observation with the naked eye and allow the user to operate the system to achieve the task requirements. We recorded the completion time for each task and recorded feedback or questions from participants for later analysis. Before participants completed all seven tasks, we asked them to complete a questionnaire about if the DSTVis system is practical and aesthetic. These questions were designed to evaluate our system, and a seven-point Likert scale ranging from strongly disagree (1) to strongly agree (7) was used to measure the proposed research items. In addition, an informal post-

Table 2. Average of time spent by users to complete the task

Question	1	2	3	4	5	6	7
Average Time(s)	33.83	107.67	45	143.67	145.39	209.11	90.44

session interview was conducted with each participant to obtain their overall perception of our system.

6.3 Results

We tabulated the results of users' assessment of themselves and their assessment of the aesthetics and usefulness of the system (Fig. 6). The user's assessment scores for themselves are distributed in the range of 4-5, while the user's ratings for the system are distributed in the range of 5-6, and some can even be rated as 7, indicating that our system achieves better results in terms of aesthetics. In addition, we analyzed the answers to the objective questions filled out by the participants, and all participants were able to complete the tasks well and choose the correct option. We removed the maximum and minimum values from the user's time to complete the task to find the final average, and Table 2 shows the average time for participants to complete and submit their answers for each task. We can see that task one can find the result very quickly, and task six took much time since the user needed to keep zooming in and out of the map to find the point at the start moment and get the predicted value of the scroll angle at that moment. Overall, the system is designed to satisfy the need for practicality.

7 DISCUSSION

After completing the questionnaire, we interviewed the participants to learn about their satisfaction with the system or their problems in realizing the tasks. One of the users told us about the excellent performance of the DSTVis system in analyzing multidimensional attributes of UAVs and the adaptive adjustment of the Circle View. He said it is a perfect performance in the face of many attributes and does not create a redundant effect visually when adding attributes. In addition, the user also praised our Search Attribute feature. There are many attributes on the control panel, which can cause much trouble to the user in the process of filtering attributes for display. However, our system can search for attributes in the search box and automatically jump to the attribute's location, which dramatically saves time in the analysis task and facilitates work progress. Other users pointed out that displaying the spatial attributes carried by the points on the path on the map is an outstanding feature because they no longer need to switch between multiple tools to solve the problem. In addition, there is a style for presenting the current moment in the view, which is a fascinating interaction. It is obvious how the points on different trajectories are sequenced on the timeline. We introduced some other analysis tools to our users, who said that our analysis tools were at the forefront of aesthetics and animation design. After using the mouse to drag the main sector ring to rotate and complete the merged sector ring, some users expressed high appreciation for the interaction. Also, in the History View, users found the design of the scan line to pick up the value of a particular attribute of a drone at a specific time very smooth.

Limitations. Since we are still in the early stages of the software lifecycle, it is not easy to assess its long-term impact. We want the DSTVis system to be part of the user's daily workflow. Only then can we identify some problems with the system in time. Based on the results of our user study and expert interviews, we have identified certain limitations of our system.

In addition to the positive feedback, participants suggested some improvements to our system. For Map View, three participants used a relatively long time to complete the Map View task. Due to the limited maximum magnification of the map, picking up points on the path is difficult. So we reworked the magnification and set a new linkage between Circle View and Map View, where tapping the map in Circle View would automatically Map View get the geographic location of the click and centre that coordinate in Map View. Some participants initially did not understand why we set the main sector ring to represent

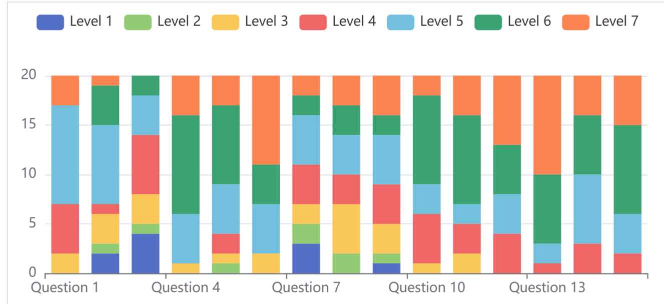


Fig. 9. Users' ratings of their own aesthetics and of the system's aesthetics and usability.

the target and each circle layer to represent the attribute. They thought each layer represented a target, and the sector rings with the same concentric angle represented the same attribute. We did a comparison experiment to clarify our original idea of this design to the users, and they finally agreed that our arrangement was the most appropriate. During the interviews, some users suggested that colour coding is not friendly to people with soft colours. Since multi-factor time-varying data has a large amount of information of different nature, differentiation by colour alone does cause some visual distress to the user. We are also looking for a suitable coding method for differentiating different attributes or targets. There are also some shortcomings that anomalous information cannot be marked in the system at the moment, and our users expect the system to be able to warn them in time to recover the loss. In fact, these are also what we are considering achieving. In addition, our Map View is a 2D view, and for height information, we display it in the form of an area map. We will try to do some 3D point cloud reconstruction work to achieve a 3D view of the flight status of the UAV on Map View, which is more conducive to UAV management and analysis.

8 CONCLUSION AND FUTURE WORK

In this work, we propose DSTVis, a visual analysis system that helps users analyze the flight data of a large number of UAVs. It consists of three main views: 1) Circle View for analyzing and comparing real-time attribute information; 2) Map View, which combines temporal and spatial attributes of drones and aims to obtain information about drone movements in past moments with fewer operations; 3) History View, which is used to analyze a large amount of drone flight information from past periods. In addition, we provide a series of practical system interactions, which greatly improves the work efficiency. Finally, we have carried out a user experiment, which proves the effectiveness and practicability of our system.

The system has proven its effectiveness in the current experimental phase. However, it is too early to tell whether DSTVis will become part of our users' daily workflow or develop into a practical system that will be widely circulated. In our future work, we will focus on experimentation and user feedback to plan, implement and adjust features to better meet user goals by continuously improving our system. We will continue to explore reliable ways to colour code and improve 2D views into 3D effects. In the future, DSTVis will become a widely circulated visualization system.

ACKNOWLEDGMENTS

This work was supported in part by National Natural Science Foundation of China (201904d07020010), Key Scientific Research Foundation of the Education Department of the Anhui Province (KJ2020A0470), and the Scientific and Technological Achievement Cultivation Project of Intelligent Manufacturing Research Institute of Hefei University of Technology (IMIPY2021022).

REFERENCES

[1] Ieee approved draft standard for drone applications framework. *IEEE P1936.1/D7.0, 2021*, pp. 1–32, 2021.

[2] M. Alwateer and S. W. Loke. A two-layered task servicing model for drone services: Overview and preliminary results. In *2019 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, pp. 387–390, 2019. doi: 10.1109/PERCOMW.2019.8730701

[3] G. Andrienko, N. Andrienko, J. Dykes, S. I. Fabrikant, and M. Wachowicz. Geovisualization of dynamics, movement and change: key issues and developing approaches in visualization research, 2008.

[4] A. Bitar, A. Jamal, H. Sultan, N. Alkandari, and M. El-Abd. Medical drones system for amusement parks. In *2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA)*, pp. 19–20, 2017. doi: 10.1109/AICCSA.2017.62

[5] D. Cashman, S. Xu, S. Das, F. Heimerl, C. Liu, S. R. Humayoun, M. Gleicher, A. Endert, and R. Chang. Cava: A visual analytics system for exploratory columnar data augmentation using knowledge graphs. *IEEE Transactions on Visualization and Computer Graphics*, 27(2):1731–1741, 2020.

[6] A. Elsharkawy, K. Naheem, and M. S. Kim. Build a real-time flight control algorithm for the lighter than air indoor robot using px4 autopilots support from uav toolbox. In *2021 21st International Conference on Control, Automation and Systems (ICCAS)*, pp. 2226–2229. IEEE, 2021.

[7] T. Fujiwara, N. Sakamoto, J. Nonaka, K. Yamamoto, K.-L. Ma, et al. A visual analytics framework for reviewing multivariate time-series data with dimensionality reduction. *IEEE transactions on visualization and computer graphics*, 27(2):1601–1611, 2020.

[8] H. Guo, D. Lenz, J. Xu, X. Liang, W. He, I. R. Grindeanu, H.-W. Shen, T. Peterka, T. Munson, and I. Foster. Ftk: A simplicial spacetime meshing framework for robust and scalable feature tracking. *IEEE Transactions on Visualization and Computer Graphics*, 27(8):3463–3480, 2021.

[9] K. M. Hasan, W. S. Suhaili, S. H. Shah Newaz, and M. S. Ahsan. Development of an aircraft type portable autonomous drone for agricultural applications. In *2020 International Conference on Computer Science and Its Application in Agriculture (ICOSICA)*, pp. 1–5, 2020. doi: 10.1109/ICOSICA49951.2020.9243257

[10] V. Hassija, V. Chamola, A. Agrawal, A. Goyal, N. C. Luong, D. Niyato, F. R. Yu, and M. Guizani. Fast, reliable, and secure drone communication: A comprehensive survey. *IEEE Communications Surveys Tutorials*, 23(4):2802–2832, 2021. doi: 10.1109/COMST.2021.3097916

[11] J. T. Jang and S. Han. Analysis for vtol flight software of px4. In *2018 18th International Conference on Control, Automation and Systems (ICCAS)*, pp. 872–875. IEEE, 2018.

[12] W. Javed and N. Elmqvist. Exploring the design space of composite visualization. In *2012 ieee pacific visualization symposium*, pp. 1–8. IEEE, 2012.

[13] D. A. Keim. Information visualization and visual data mining. *IEEE transactions on Visualization and Computer Graphics*, 8(1):1–8, 2002.

[14] H. Kim, J. Choo, H. Park, and A. Endert. Interaxis: Steering scatterplot axes via observation-level interaction. *IEEE transactions on visualization and computer graphics*, 22(1):131–140, 2015.

[15] S. Kim, S. Jeong, I. Woo, Y. Jang, R. Maciejewski, and D. S. Ebert. Data flow analysis and visualization for spatiotemporal statistical data without trajectory information. *IEEE transactions on visualization and computer graphics*, 24(3):1287–1300, 2017.

[16] A. Lee, D. Archambault, and M. A. Nacenta. The effectiveness of interactive visualization techniques for time navigation of dynamic graphs on large displays. *IEEE Transactions on Visualization and Computer Graphics*, 27(2):528–538, 2020.

[17] C. Li, G. Baciu, and Y. Han. Streammap: Smooth dynamic visualization of high-density streaming points. *IEEE transactions on visualization and computer graphics*, 24(3):1381–1393, 2017.

[18] C. Li, G. Baciu, Y. Wang, J. Chen, and C. Wang. Ddlvis: Real-time visual query of spatiotemporal data distribution via density dictionary learning. *IEEE Transactions on Visualization and Computer Graphics*, 28(1):1062–1072, 2021.

[19] D. Liu, P. Xu, and L. Ren. Tpflow: Progressive partition and multidimensional pattern extraction for large-scale spatio-temporal data analysis. *IEEE transactions on visualization and computer graphics*, 25(1):1–11, 2018.

[20] C. Ma, Y. Zhou, and Z. Li. A new simulation environment based on airsim, ros, and px4 for quadcopter aircrafts. In *2020 6th International Conference on Control, Automation and Robotics (ICCAR)*, pp. 486–490. IEEE, 2020.

[21] N. Matsumura, K. Nishimori, R. Taniguchi, T. Mitsui, and T. Hiraguri. Performance improvement of drone mimo relay station using selection of drone placement. In *2018 IEEE International Workshop on Electromagnetics:Applications and Student Innovation Competition (iWEM)*, pp. 1–1, 2018. doi: 10.1109/iWEM.2018.8536637

[22] S. McKenna, D. Mazur, J. Agutter, and M. Meyer. Design activity framework for visualization design. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2191–2200, 2014.

[23] L. Meier, D. Honegger, and M. Pollefeys. Px4: A node-based multithreaded open source robotics framework for deeply embedded platforms. In *2015 IEEE international conference on robotics and automation (ICRA)*, pp. 6235–6240. IEEE, 2015.

[24] T. Munzner. *Visualization analysis and design*. CRC press, 2014.

[25] K. D. Nguyen and T.-T. Nguyen. Vision-based software-in-the-loop-simulation for unmanned aerial vehicles using gazebo and px4 open source. In *2019 International Conference on System Science and Engineering (ICSSE)*, pp. 429–432. IEEE, 2019.

[26] J. Pálenik, J. Byška, S. Bruckner, and H. Hauser. Scale-space splatting: Reforming spacetime for cross-scale exploration of integral measures in molecular dynamics. *IEEE Transactions on Visualization and Computer Graphics*, 26(1):643–653, 2019.

[27] V. Peña-Araya, E. Pietriga, and A. Bezerianos. A comparison of visualizations for identifying correlation over space and time. *IEEE transactions on visualization and computer graphics*, 26(1):375–385, 2019.

[28] M. V. Sabando, P. Ulbrich, M. Selzer, J. Byška, J. Mičan, I. Ponzoni, A. J. Soto, M. L. Ganuza, and B. Kozlíková. Chemva: interactive visual analysis of chemical compound similarity in virtual screening. *IEEE Transactions on Visualization and Computer Graphics*, 27(2):891–901, 2020.

[29] D. Sacha, F. Al-Masoudi, M. Stein, T. Schreck, D. A. Keim, G. Andrienko, and H. Janetzko. Dynamic visual abstraction of soccer movement. In *Computer Graphics Forum*, vol. 36, pp. 305–315. Wiley Online Library, 2017.

[30] D. Saffo, A. Leventidis, T. Jain, M. Borkin, and C. Dunne. Data comets: Designing a visualization tool for analyzing autonomous aerial vehicle logs with grounded evaluation, Jun 2021.

[31] S. Storm-Furru and S. Bruckner. Va-trac: Geospatial trajectory analysis for monitoring, identification, and verification in fishing vessel operations. In *Computer Graphics Forum*, vol. 39, pp. 101–114. Wiley Online Library, 2020.

[32] M. Van Dortmont, S. van den Elzen, and J. J. van Wijk. Chronocorrelator: Enriching events with time series. In *Computer Graphics Forum*, vol. 38, pp. 387–399. Wiley Online Library, 2019.

[33] Y. Wu, X. Xie, J. Wang, D. Deng, H. Liang, H. Zhang, S. Cheng, and W. Chen. Forvizor: Visualizing spatio-temporal team formations in soccer. *IEEE transactions on visualization and computer graphics*, 25(1):65–75, 2018.

[34] K. Xu, M. Xia, X. Mu, Y. Wang, and N. Cao. Ensemblelens: Ensemble-based visual exploration of anomaly detection algorithms with multidimensional data. *IEEE transactions on visualization and computer graphics*, 25(1):109–119, 2018.

[35] D. Yallappa, M. Veerangouda, D. Maski, V. Palled, and M. Bheemanna. Development and evaluation of drone mounted sprayer for pesticide applications to crops. In *2017 IEEE Global Humanitarian Technology Conference (GHTC)*, pp. 1–7, 2017. doi: 10.1109/GHTC.2017.8239330

[36] J.-H. Yang and Y. Chang. Feasibility study of rfid-mounted drone application in management of oyster farms. In *2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pp. 3610–3613, 2017. doi: 10.1109/IGARSS.2017.8127780

[37] S. Yang, Y. Liu, and W. Yang. Research and application of 5g drone inspection technology based on high altitude and alpine areas. In *2021 IEEE International Conference on Electronic Technology, Communication and Information (ICETCI)*, pp. 93–97, 2021. doi: 10.1109/ICETCI53161.2021.9563499

[38] X. Yue, J. Bai, Q. Liu, Y. Tang, A. Puri, K. Li, and H. Qu. sportfolio: Stratified visual analysis of stock portfolios. *IEEE transactions on visualization and computer graphics*, 26(1):601–610, 2019.

[39] H. Zhang, Y. Hou, D. Qu, and Q. Liu. Correlation visualization of time-varying patterns for multi-variable data. *IEEE Access*, 4:4669–4677, 2016.

[40] J. Zhang, T. Yu, J. Chen, L. Hou, J. Diao, and Y. Zhang. Design of ground monitor and control system for uav remote sensing based on world

wind. In *2012 IEEE International Conference on Computer Science and Automation Engineering*, pp. 51–54. IEEE, 2012.

[41] X. Zhao, Y. Wu, W. Cui, X. Du, Y. Chen, Y. Wang, D. L. Lee, and H. Qu. Skylens: Visual analysis of skyline on multi-dimensional data. *IEEE transactions on visualization and computer graphics*, 24(1):246–255, 2017.

[42] J. Zong et al. *Designing interactive visualizations by demonstration*. PhD thesis, Massachusetts Institute of Technology, 2020.