CIE5050-09 Additional Research Project

Public Transport Delay Pattern Analysis by Unsupervised Learning Approach

Yuxing Cheng – 5612233

Department of Transport & Planning, Delft University of Technology
Supervised by Panchamy Krishnakumari

06/2022-09/2022

Supervisors: Transport and Planning
P.K. Krishnakumari
O. Cats

ABSTRACT
To analyze latent multiple specific patterns in the line-based public transport daily delay occurrence, a data-driven explorative analysis of public transport daily delay spatial-temporal distribution pattern is performed based on the *k*-means clustering algorithm. Firstly, we used aggregated daily delay profile to visualize how the delay is distributed in space and time. And the pattern of daily delay distribution is represented by the image features. Secondly, the image features are extracted by the pre-trained neural network ResNet50, and the output image feature vector are used for implementing unsupervised k-means clustering algorithm. Finally, the *k*-means clustering results reveal five different daily delay patterns. The distinctive characteristics of these five delay patterns are analyzed and lead to some significant results, which could provide public transport operators with a better understanding of how delays occur on a specific line.

Keywords: Public transport, Delay pattern, Unsupervised learning, Clustering, AVL data, image recognition

INTRODUCTION

Punctuality is the essential quality that public transport systems are pursuing. To achieve this goal, strategic, tactical, and operational management are needed(*1*), which requires the operator to own prior knowledge of the characteristics of public transport (PT) running. However, each PT lines and transport hub in the PT system have distinctive functions, and their delay occurrence characters could be different. Are there any spatial-temporal characteristics of delay that occurred on a specific PT line? The increasing variety of PT-related data resources is providing more opportunities for the operators to censor different phenomena that appear in daily PT operations, especially automatic vehicle location (AVL) data and general transit feed specification (GTFS) data(*2*, *3*). AVL and GTFS datasets contain dynamic (e.g., locations) and static (e.g., stop information, geographic structure, and schedule) information collected when the tram lines are operated(*4*). This research aims to construct a methodology for data-driven exploration of PT line-based spatial-temporal patterns of delay occurrence. Understanding the spatial-temporal pattern of delay distribution on a single line could be meaningful for the operator(*5*).

A large amount of research effort has been built on the AVL data to better understand the characteristics of phenomena that occur in PT lines. Methods for extracting PT running information from these data were developed by previous studies to explore the service reliability and extracting the spatiotemporal load profile is one of the efficient approaches for this purpose(*6*, *7*). The spatiotemporal load profile was introduced to merge multiple data sources by building the algorithm to extract meaningful information from raw data and visualize them in one profile(*2*). The approach to making the profile for PT running visualization has been implemented and used in various research in the PT domain. For example, (*3*) implemented the operation profile to visualize the daily running situation of a single tram line. The profile images contain the real-time location of PT vehicles, combined with the passenger loads on the journey. And the defined bunching phenomenon is detected and clustered according to an unsupervised machine learning method. Similarly, to predict the short-term train loads, (*8*) introduced an image-processing-oriented methodology, and the image represents the train loads at each stop.

Among the research focused on the PT operation, multiple cases did the exploration analysis based on unsupervised learning(*9*). Unsupervised learning techniques have recently been employed to investigate spatial travel patterns and demand, given their natural advantages in solving clustering problems(*3*). In the area of public transport, many analyses related to clustering rely on *k*-means, which permit to cluster relatively large sets of data and require only a few parameters. One crucial parameter is the desired number of clusters(*10*). Most of the previous studies that implement the *k*-means algorithm in the PT field are based on the low-dimensional input, which means the number of attributes of each data point is relatively small and definite(*3*, *5*).

However, the disadvantage of the low-dimensional input for clustering algorithms is that the more complex spatiotemporal PT dynamics could not be well represented, and the clustering could only be based on the limited features among data points. The advantage of the profile derived from the real-time PT data (e.g., AVL data) is that it can construct a complete view of PT operation in time, space, and more dimensions. Still, the existing research methodology could not fully use this advantage. To the best of our knowledge, very few studies have attempted to use clustering on the high-dimensional input (e.g., image) that represents the PT dynamics. Combining clustering and representation learning is one of the most promising approaches for unsupervised learning of deep neural networks (8). Thus, this paper focuses on extracting the latent feature contained in the line-based PT daily delay profile image by a deep learning algorithm and performing the *k*-means

clustering based on the extracted feature vectors. The study aims to answer the question: how to recognize a single PT line's distinctive daily delay patterns and analyze them with the *k*-means clustering approach. The contributions of this paper are as follows:

- We build an image-based method to extract and visualize a single PT line's line-based daily punctuality information. The generated daily delay profile image could allow us to view how the delay occurs and its spatial-temporal distribution characteristics.
- This study applies a pre-trained convolution neural network architecture, Resnet50, for image feature recognition. The advantage of this feature recognition approach is that the abstract spatial-temporal distribution characteristics of delay occurrence could be extracted from the profile images. This technique is different from the previous studies that vectorize the daily punctuality data for the *k*-means algorithm or define the attributes of each sample manually.
- The clustering results provide a generalized overview of different delay patterns on a specific PT line. This can provide prior knowledge for further studies such as supervised learning on PT dynamic patterns or planning and management applications. This methodology is generalizable to be extended to other PT lines or systems.

The next section of this paper presents the proposed methodology from two aspects, the details of implementing the *k*-means algorithm in this research and the image processing approaches. Then, the case study setup based on the cleaned AVL data is introduced. After that, the results of clustered delay patterns are presented, with the analysis of spatial-temporal characteristics of each kind of delay pattern. Finally, the conclusion is drawn with discussion and suggestions for further research.

METHODOLOGY

In this section, the methodology for detecting the spatial-temporal characteristics of different daily delay patterns is introduced. An overview based on a conceptual model is given first. Then, the *k*-means clustering method is described in detail. Also, the image processing approaches in this research are discussed, as they provide the bridge between the daily delay profile image and the *k*-means clustering algorithm. Punctuality mentioned in this research refers to the time difference between the scheduled and real departure time for the tram at each stop. So, the value of punctuality could be positive (representing delay) or negative (indicating early arrival). For simplification, we mainly focus on exploring the delay patterns. And the feature of early arrival pattern can be obtained as complementary findings.

Overview

An overview of the methodological framework is shown in
Figure 1. The raw AVL and GTFS datasets are stored separately as input, which contains the daily running situation information and the PT lines network information. The raw data obtained from these two datasets are cleaned and useful information is extracted. Then, based on the three-step method, the daily delay patterns with different spatial-temporal characteristics are derived from the raw data and clustered based on the daily delay profile image. Finally, the daily delay patterns are interpreted to extract insights, which is this study's final output.

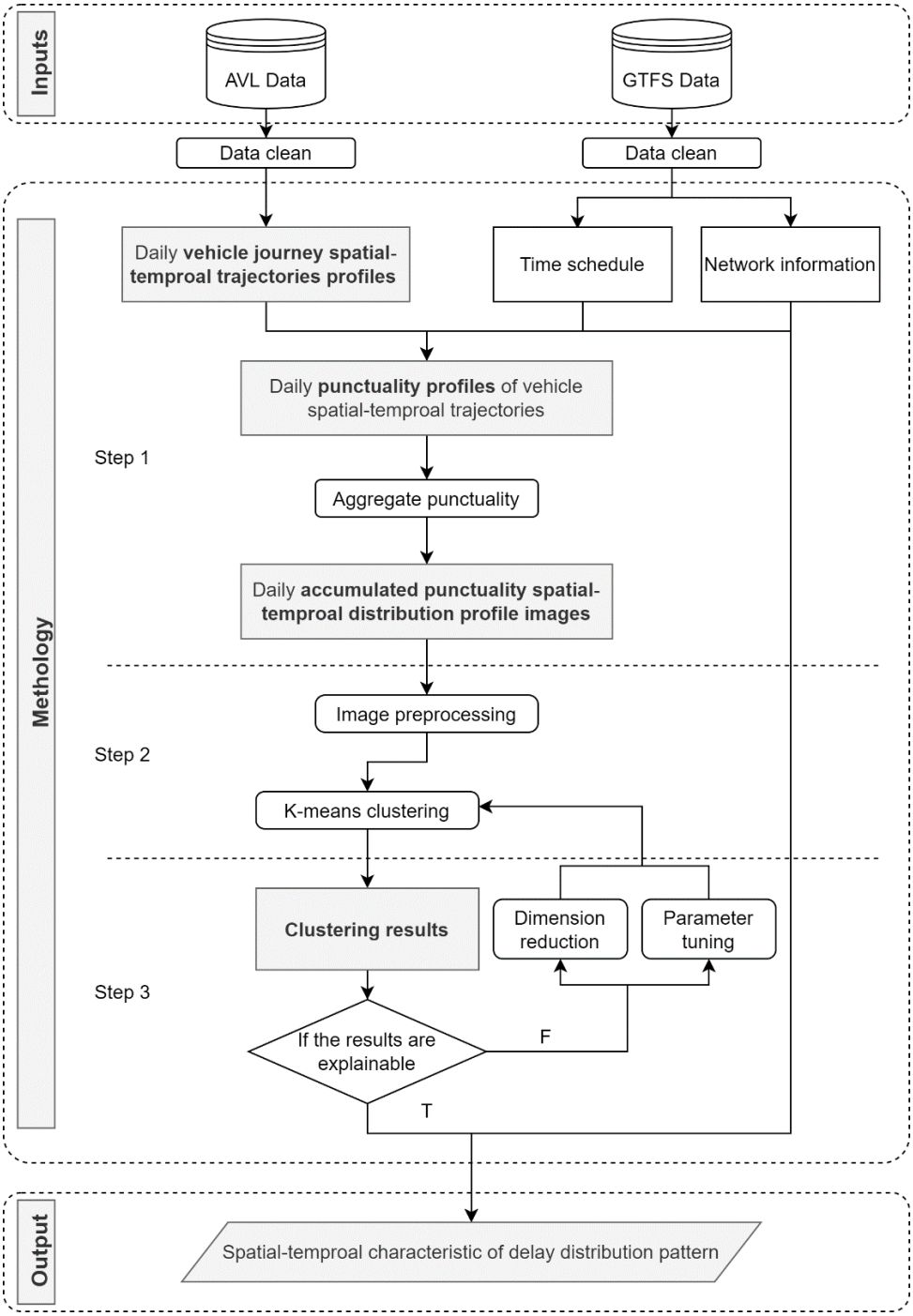


Figure 1 Overview of the methodology. The blocks with gray shadows represent the key (intermediate) results.

***k*-means algorithm (for image clustering)**

k-means is the most concise clustering algorithm in unsupervised learning, which was first proposed in (*11*), and has been leveraged in various fields. Given a dataset X, containing *n* datapoints with *m* dimensions, $X = \begin{pmatrix} x_{11} & \cdots & x_{1m} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{nm} \end{pmatrix}$, the algorithm aims to partition them into *k* ($k<n$) clusters $\{C_1, C_2, C_3, \dots, C_k\}$. For each cluster, the centroid is defined as the mean of data points belonging to the cluster and is calculated iteratively until the algorithm process is terminated. The definition of centroid could also be generalized in high dimensional space. For each iteration, each datapoint x is assigned to the nearest cluster, based on the distance to each centroid. Multiple methods are used to calculate the distance between data points (*10*), and the Euclidean distance is the most used. The algorithm iteration is terminated when the assignment results of all data point no longer change. The steps of generic *k*-means include the following steps:

1. Select *k* samples $\{u_1, u_2, u_3, \dots, u_k\}$ in the dataset randomly as initial cluster centroids of *k* clusters.
2. For all the other data points, calculate their distance to the initial cluster centroids, and assign them to the nearest cluster. The most common distance calculation is Euclidean distance D, which is:

$$D_i(x_i, C_i)^2 = \sum_{d=1}^{m} (x_{id} - C_{id})^2 = \|x_i - C_i\|_2^2 \quad (1)$$

   Where i denotes the label of a cluster, D_i denotes the Euclidean distance between datapoint x_i and centroid u_i both belong to the cluster C_i.
3. Calculate the mean of each cluster as the new centroid of the cluster.
4. Run steps 2 and 3 iteratively until the limitation of iteration or the assignment no longer changes, which means the within-cluster sum of square SSE (the sum of the distance of each datapoint to the corresponding cluster centroid).

$$SSE(k) = \sum_{i=1}^{k} \sum_{x \in C_i} D_i(x_i, u_i)^2 \quad (2)$$

For each cluster, the ideal situation is that the distance between each data point assigned to the cluster and the centroid could be as small as possible. In contrast, the difference (distance) among centroids of multiple clusters could be as significant as possible. Thus, the aim of the *k*-means algorithm is an optimization problem where the goal is to minimize the within-cluster sum of squared errors (SSE).

A common issue in the unsupervised machine learning algorithm is that clustering methods always return clusters even if the data does not contain any clusters. It is necessary to evaluate if there exists a significant clustering tendency in the vectorized input dataset and if it could obtain reasonable and meaningful clustering results. To evaluate the clustering tendency of a specific dataset, we implement the Hopkins Statistic *h* by estimating the data set's randomness (*12*, *13*). *h* will be in the range (0,1), and a high value indicates highly clustered data points. An *h* closer to 0 refers to a lower cluster tendency and is more regularly spaced. An *h* closer to 1 refers to a higher cluster tendency. If the data is uniformly distributed, the *h* will be 0.5(*14*).

The key point of *k*-means clustering is to determine the number of clusters *k.* Two methods have been commonly used for solving the problem: the elbow method based on the SSE curve and the silhouette analysis

The Elbow method is one of the most popular methods to select the optimal number of clusters by fitting the model with a range of values for *k* in the *k*-means algorithm. The Elbow method requires a line plot between SSE and the number of clusters and finding the point representing the "**elbow point**." However, the within-cluster SSE could not determine the optimum cluster number independently. The elbow could be unclear, and SSE could only reflect the data distribution within each cluster.

The other method is silhouette analysis. The silhouette analysis allows seeing how similar the points within the cluster are with the centroid point and how different they are from the points of other clusters(*15*). For each data point in a cluster, a silhouette score could be calculated scale from -1 to 1, and the average value of all silhouette score represent the results of silhouette analysis. The following equation calculates the average silhouette score.

$$S = \frac{1}{n}\sum_{i=1}^{n}\frac{b_i - a_i}{\max(a_i, b_i)} \tag{3}$$

Where, a_i represents the average distance between sample *i* and other samples in its cluster, and b_i denotes the minimum average distance between sample *i* and samples in other clusters. If the a_i verge to 0 or b_i is high enough, the average silhouette score getting closer to 1 indicates that the clustering algorithm works better.

Although the mentioned methods could provide quantifiable evidence for choosing the value of *k*, the number of clusters should also consider the purpose of clustering based on prior knowledge. In this research, we need to observe if universal and generic features exist among data points within each cluster. This way, the patterns of daily delay distribution characteristics could be extracted, concluded, and generalized.

Image processing approaches

As introduced before, *k*-means clustering is based on real vectors with multiple dimensions. However, the daily delay information is represented by the delay profile images. So, how the images are vectorized and plugged into the *k*-means algorithm needs to be discussed.

In the previous research, the dimensions of the *k*-means input are limited (usually no more than 10^1 or 10^2 attributes (dimensions)). However, in this research, the spatial-temporal distribution of the delay phenomenon is represented by the daily delay profile image and the dimension of image input we propose to obtain (more than 10^2 or 10^3, depending on the aggregation granularity of the delay profile images) could be relatively much higher. The challenges brought by the high dimensional input for *k*-means include two aspects:

1) The *k*-means algorithm determines the data points' cluster affiliation based on the pair-wise distance, but all the points are at a similar distance from the others when the dimension increases. Thus, the notion of "nearest points" vanishes in the high dimensional space(*16*).
2) The input for *k*-means distance calculation is one-dimensional vectors. If an image is unfolded to a one-dimensional vector directly, any possible translation or disturbance of the image could significantly impact the clustering result due to the difference in pixel

values(*17*). For example, in two identical images, if one of them is translated to a one-pixel distance, no difference could be found in their appearances. Still, they may be attributed to two different clusters.

3) The spatial relationship among the image pixels would be ignored if we directly unfold the image pixel values to a one-dimensional vector. One of the most significant advantages of representing the daily delay information by the punctuality profile image is that the spatial-temporal characteristics could be visualized and analyzed.

To solve the problems caused by the characteristics of the data in this research, image recognition based on the pre-trained deep neural network architecture Resnet-50 is implemented, combined with two kinds of dimensionality reduction approaches. Using the image feature recognition algorithm Resnet-50, the spatial-temporal distribution characteristics of daily delay could be extracted. And the dimensionality reduction could aid the *k*-means algorithm to be more efficient. Thus, the appropriate input attributes for the *k*-means algorithm could be obtained.

Image feature extraction

For many image clustering or classification problems, replacing raw image data with features extracted by a pre-trained convolutional neural network (CNN) leads to better clustering performance(*18*, *19*). The previous research compared multiple neural network architectures and proved that the ResNet50 could perform relatively better than other prevailing architectures(*20*, *21*). Residual Network is a classic neural network used as a backbone for many computer vision tasks, which was first proposed by Kaiming He in 2015(*22*). ResNet-50 is a convolutional neural network with 50 layers. The pre-trained Resnet-50 deep neural network architecture could effectively recognize the features of the images and has been widely used in computer vision, including image classification and detection applications. The process of implementing the ResNet50 is done by the PyTorch deep learning framework.

Dimensionality reduction

The dimensionality reduction method could transfer the high-dimensional data into low-dimensional space, vital in feature engineering, data visualization, and saving computation time (*16*). There are two kinds of dimensionality reduction methods: projection and manifold learning(*23*). Principal Component Analysis (PCA) and t-Distributed Stochastic Neighbor Embedding (t-SNE) are two typical algorithms belonging to these two kinds of dimensionality reduction methods respectively.

As a traditional and popular dimensionality reduction technique, PCA is a linear technique that keeps the low-dimensional representations of different data points far apart. However, PCA cannot account for complex polynomial relationships between features. Unlike PCA, t-SNE is a nonlinear dimensionality reduction algorithm based on the probability distribution of random walks on the neighborhood graph to find the structure within the data. It maps multidimensional data to two or more dimensions suitable for human observation. In the research conducted by van der Maaten and Hinton(*24*), t-SNE is better than existing techniques at creating a single map that reveals structure at many different scales. It is essential for high-dimensional data that lie on several different but related, low-dimensional manifolds, such as images of objects from multiple classes seen from various viewpoints.

The *k*-means calculate which cluster a data point belongs to base on the distance among vectors with multiple dimensions, and the dimension is high in this research. Accordingly, the

dimensionality reduction could be plugged into the clustering method in two roles: 1) processing the output of ResNet50 with $3 \times 3 \times 512$ dimension to be the input of the *k*-means algorithm. 2) visualize high-dimensional *k*-means output data by giving each data point a location in a two- or three-dimensional map. The first role may not be necessary, but for some specific data, the dimensionality reduction before *k*-means could help with feature selection and reducing time complexity(*25*). So, a comparison is made among the data preprocessing methods with (PCA or t-SNE) and without dimensionality reduction before *k*-means. The second role is one of the critical parts of analyzing the effectiveness of the clustering algorithm, which is necessary. So, the t-SNE is chosen to visualize the clustering results.

CASE STUDY

For this study, the General Transit Feed Specification (GTFS) dataset and the Automatic Vehicle Location (AVL) dataset are used to extract the historical real-time running information of all the PT lines in The Hague, covering 79 days across June, July, and August in 2019. Among all the PT lines, tram line 1 was selected for the case study. As the oldest and longest tram line in The Hague, line 1 runs from Scheveningen Noord to Delft Tanthof, via The Hague city center, Hollands Spoor station, Rijswijk Haagweg, and Delft station, as shown in Figure 2. The line has been running in and around The Hague for decades. The diversity of the land use pattern where tram line 1 pass by could ensure that multiple kinds of daily delay patterns exist for the line.

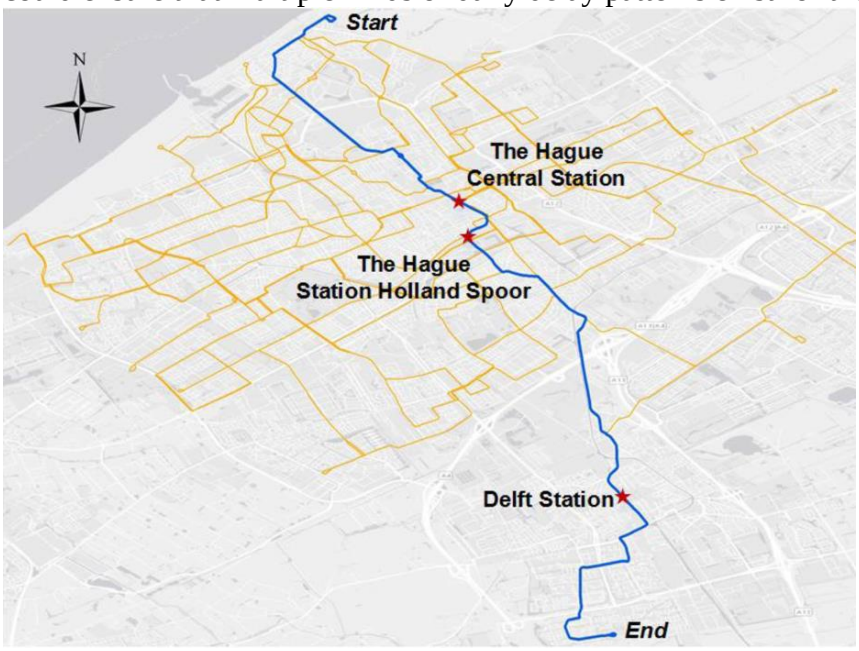


Figure 2 Tram line 1 from Scheveningen Noorderstrand to Delft Tanthof (TBD)

The dynamic information is derived from the AVL data, Table 1 which contains: line number, vehicle number, journey number, the actual arrival/departure time to the platform of the vehicle, stop (platform) code, distance to the last platform, punctuality, and more related information of all the PT lines operated in The Hague as shown in Table 1. Besides, the static network information is derived from the GTFS dataset, which contains each platform's stop name, code, and geographic location. To extract the available information, we select the data point with the actual stop name and collect the exact time and position data when a tram leaves the platform. Finally, 78 platforms

(34 stops in one direction) are identified and all the stops of tram line 1 are selected. And for each data point, the punctuality is derived from the schedule and departure time. Besides, some of the data points with extreme attribute values are deleted from the dataset, as these could be caused by a particular situation like rare equipment failure or temporary traffic control. The well-organized data could make the subsequent computation more efficient through data cleaning.

Table 1 Description of AVL data of each tram vehicle

Name	Data type	Example	Description
receive	datetime	2019-06-05 04:51:52.086383	Time of sending message by the source system
messagetype	string	DEPARTURE	The status of the vehicle
operatingday	datetime	2019-06-05	a specific date the datapoint belongs to
dataownercode	string	HTM	Operator company
lineplanningnumber	int	1	Line number of the journey belongs
journeynumber	int	30004	Public journey number
userstopcode	int	9594	Stop number of the stop where the arrival/leave is.
punctuality	int	20.0	Current deviation from the scheduled arrival time in seconds for this stop. Too early <0, too late >0, on time =0
rd_x&rd_y	float	86795.0, 454011.0	RDS in meters. RD coordinates refer to locations in the Netherlands according to the Rijksdriehoek system4
vehiclenumber	int	4047	Vehicle identification number

RESULT: EXPLORED DAILY DELAY PATTERNS

Punctuality visualization

To represent the daily delay spatial-temporal distribution pattern of tram line 1, the delay profile is obtained from the AVL and GTFS data, as Figure 3 illustrates. Figure 3 contains the spatial-temporal trajectories of tram line 1 tram in a single operation day, in both directions. The dots represent the time and location when a tram sends the signal that it is leaving a platform. The shade colors of the dots represent the punctuality in the unit of seconds, with the value from -200 to 200 seconds. The positive value represents delay, and the negative value represents early arrival. The dash lines connect multiple dots representing the trajectories of tram vehicles. The blank segmentation that appears between platforms 38 and 39 represents the end of the single-direction journey. The range of the y-axis from platform 1 (“Den Haag, Zwarte Pad”) to platform 38 (“Delft, Abtswoudsepark”) represents one direction, and this direction is defined as “direction 1”. The range of the y-axis from platform 39 (“Delft, Abtswoudsepark”) to platform 78 (“Den Haag, Zwarte Pad”) is defined as “direction 2”.

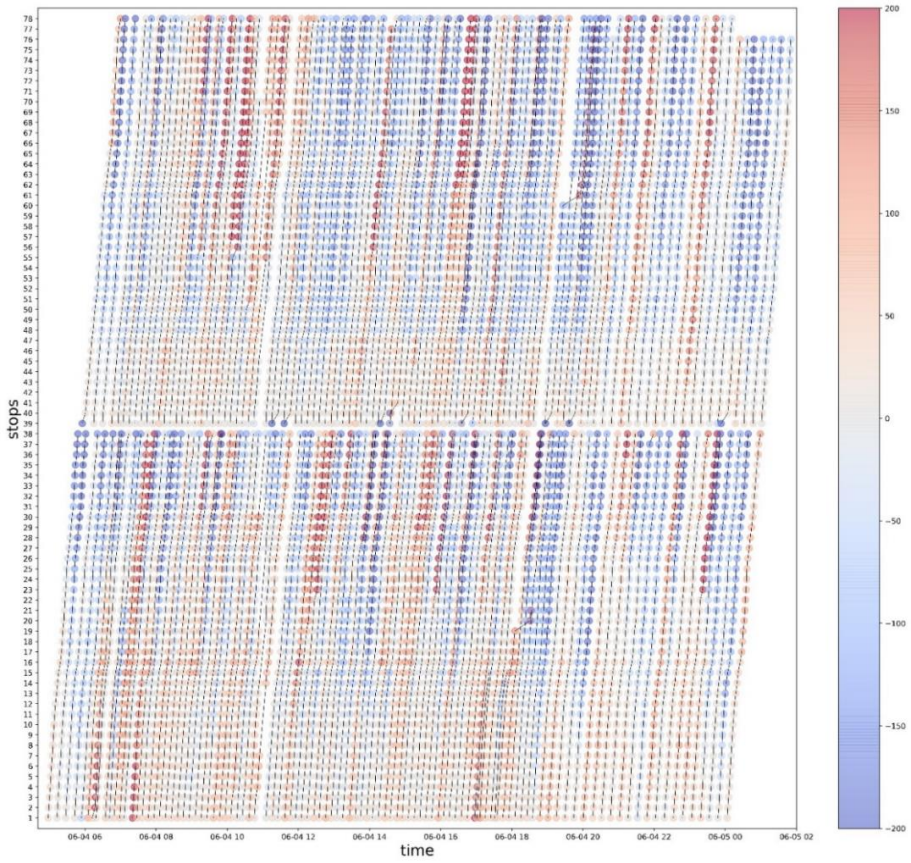
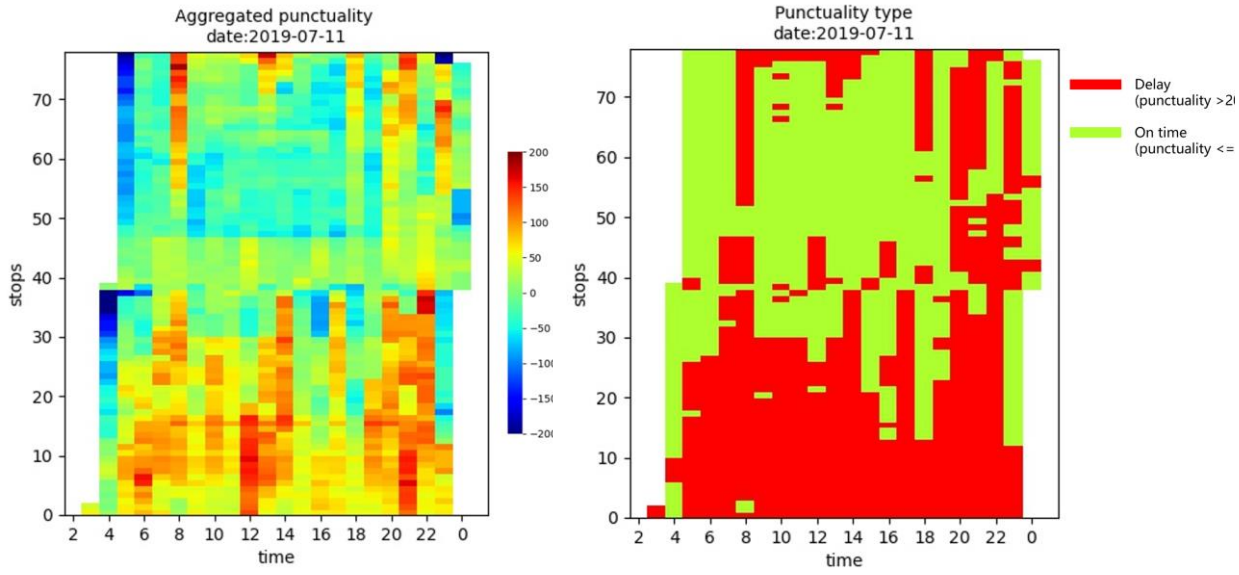


Figure 3 The punctuality profile of tram line 1 on the date: 2019-07-11

Figure 4 visualizes the aggregated punctualities based on the punctuality profile. Two levels of punctuality aggregation are used: aggregate to the original value and aggregate to two punctuality types ("delay" or "on time"). Figure 4a shows the original punctuality value in space and time scale. Figure 4b shows the classified punctuality types (delay or no delay), which could indicate the temporal-spatial distribution of delay occurrence but ignore the delay severity. For Figure 4b, the criteria of delay definition are derived from the distribution of the punctuality values in the whole dataset. The purpose of using two kinds of punctuality aggregation levels is to compare which one could better reflect the feature of the daily delay pattern and lead to clustering results with more distinctive features. Thus, the output images of 79 days aggregated daily delay profiles with different aggregation levels will be processed by ResNet50 and compared.



a) Original punctuality values **b) Classified punctuality types**

Figure 4 Two kinds of aggregated punctuality profile images extracted from Figure 3

Punctuality profile images clustering

In this section, the proposed methodology for extracting the spatial-temporal characteristics of delay distribution is implemented. Based on the visualization of the daily delay distribution, the clustering could separate the daily delay information into clusters as dissimilar as possible. So, the feature of the punctuality profile images should be extracted in the appropriate approach and form the input dataset to be clustered.

Multiple methods could be used for preprocessing the input data of clustering. However, the precondition of meaningful clustering results is that the data be nonuniformly distributed and show the clustering tendency. All the cluster methods will always lead to a result, regardless of whether the input data points have a clear cluster tendency or not. Also, the clustering tendency of the input data points obtained by different approaches should be compared to evaluate if these acquired datasets could lead to meaningful clustering results.

Three image preprocessing approaches were implemented to compare their performance in transforming images into input data for *k*-means with significant clustering tendency: 1) ResNet50, 2) ResNet50 + PCA, 3) ResNet50 + T-SNE. Besides, two kinds of images we obtained in the former steps are used, which are “Original punctuality value images” and “Classified punctuality images.” These two types of images contain different kinds of daily delay distribution characteristics with different granularity. We compare these two kinds of images to see which one can obtain clusters with clear distinctions. Combining the above-mentioned image preprocessing approaches and image types, 6 combinations are tested and compared using the Hopkins Statistic *h.* The results are illustrated in Table 2 and Figure 5.

Table 2 Comparison among multiple images preprocessing approaches before *k*-means

Image preprocessing approaches		Hopkins Statistic *h*					
		mean	std	min	25%	75%	max
Original punctuality value images	1) ResNet50	0.632	0.0068	0.6175	0.6311	0.6407	0.6505
	2) ResNet50 + PCA	0.567	0.0084	0.5454	0.5571	0.5647	0.5770
	3) ResNet50 + T-SNE	0.5972	0.0752	0.4424	0.5505	**0.6555**	0.7525
Classified punctuality images	4) ResNet50	0.6224	0.0056	0.6058	0.6191	0.6269	0.6324
	5) ResNet50 + PCA	0.5616	0.0060	0.5549	0.5707	0.5793	0.6000
	6) ResNet50 + T-SNE	0.5960	0.0826	0.4684	0.5318	0.6453	**0.8289**

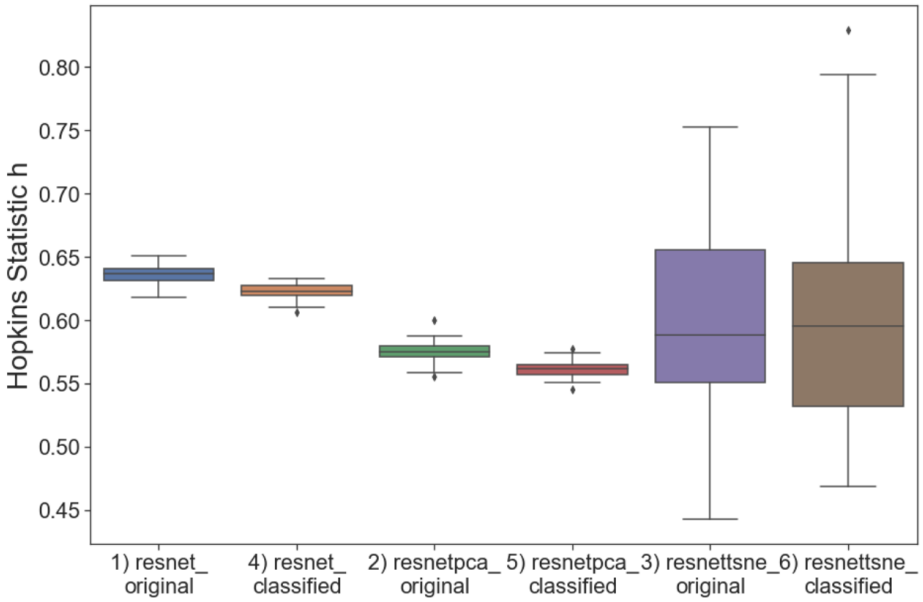


Figure 5 Comparison of Hopkins statistic *h* of six image preprocessing approaches before *k*-means clustering

Table 2 and the boxplot in Figure 5 illustrate that for all the implemented image preprocessing approaches, the means of *h* statistics are all higher than 0.5, which means these approaches could obtain the input data with a clustering tendency. Also, this could suggest the clustering tendency in daily delay dynamics. Thus, it is possible to classify the daily delay pattern and make the conclusion of delay pattern characteristics. According to Figure 5, the preprocessing method 1), 4), 2), and 5) could obtain a dataset with a relatively more stable clustering tendency, with higher means and lower deviation of *h*. However, methods 3) and 6) have the probability of obtaining the image data with a higher clustering tendency with acceptable fluctuation.

The final goals of the research are to cluster and distinguish the different daily delay patterns. So, the higher the clustering tendency, the more preferred the results to be used for explaining, even though the preprocessing method may not be so stable with the higher deviation of *h*. According to this, the "ResNet50 + T-SNE" combination is chosen as the preprocessing method before the *k*-means algorithm.

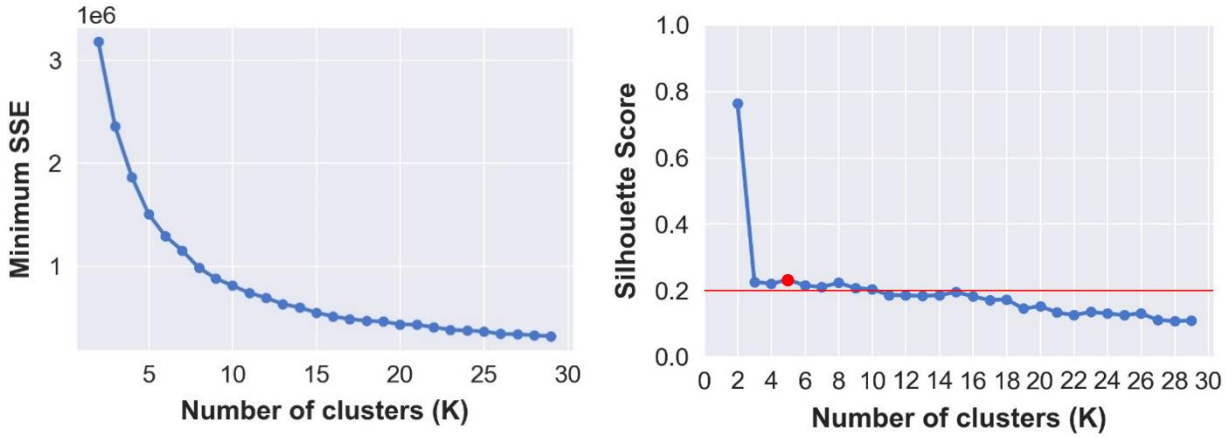
Clusters number

By choosing the suitable number of clusters *k*, the daily punctuality profile can be separated into meaningful delay patterns with distinct features. And for the *k*-means clustering algorithm, the first

issue that needs to be solved is the appropriate number of clusters *k*. To determine the *k* value, the SSE and silhouette analysis is implemented on the scale of *k* from 2 to 30. Also, the application and the analysis objectives should be considered so that the generalized daily delay distribution patterns are more desirable from the planning perspective and more explainable.

The curve of SSE and silhouette score with different *k* values is shown in Figure 6. The within-cluster sum of squared errors reflects the data homogeneity in each cluster. Figure 6a indicates that the SSE declines rapidly before *k* =8, and then the decline slope alleviates, and the approximate elbow point is in the range of 5 and 8. When *k* is lower than 8, the impact of changing the *k* value on SSE is relatively significant, so the clustering performance of *k* values in this range should be compared carefully.

Figure 6b shows the silhouette score fluctuation with the *k* value. The *k* =2 will lead to a score much higher than any other *k* value, which means the best separation could be obtained between clusters. However, combined with the corresponding SSE value when *k* =2, the variability within clusters doesn't allow for a good description of cluster characteristics. Moreover, the red line in Figure 6b denotes the score equal to 0.2. When *k* is larger than 8, the score will decrease gradually, which means the appropriate *k* value that could lead to a satisfying clustering result is between *k* =3 to 8. Besides, the red dot denotes the second-highest silhouette score at *k* =5, which leads to a relatively optimum clustering result.



(a) (b)

Figure 6 Analysis for determining the number of clusters (*k*): (a) SSE decrease exponentially as the number of cluster increases (b) Silhouette Score

According to the analysis above, the clustering results of different *k* values, ranging from 2 to 7, are visualized in Figure 7 to make a detailed comparison among these results. In Figure 7, different colors represent data points belonging to different clusters, and the number represents the cluster labels. The distance between the data points could be recognized as their similarity. The closer data points mean they are more similar, and vice versa. We can observe that the clustering results when *k* is larger than 2 are based on the clustering when *k* equals 2, and the red lines denote this phenomenon. This means the data could largely be divided into two clusters, and in each cluster, the data have the potential to be separated into multiple sub-clusters. This feature is marked by the red line in Figure 7, which is the approximate boundary between the two primary clusters.

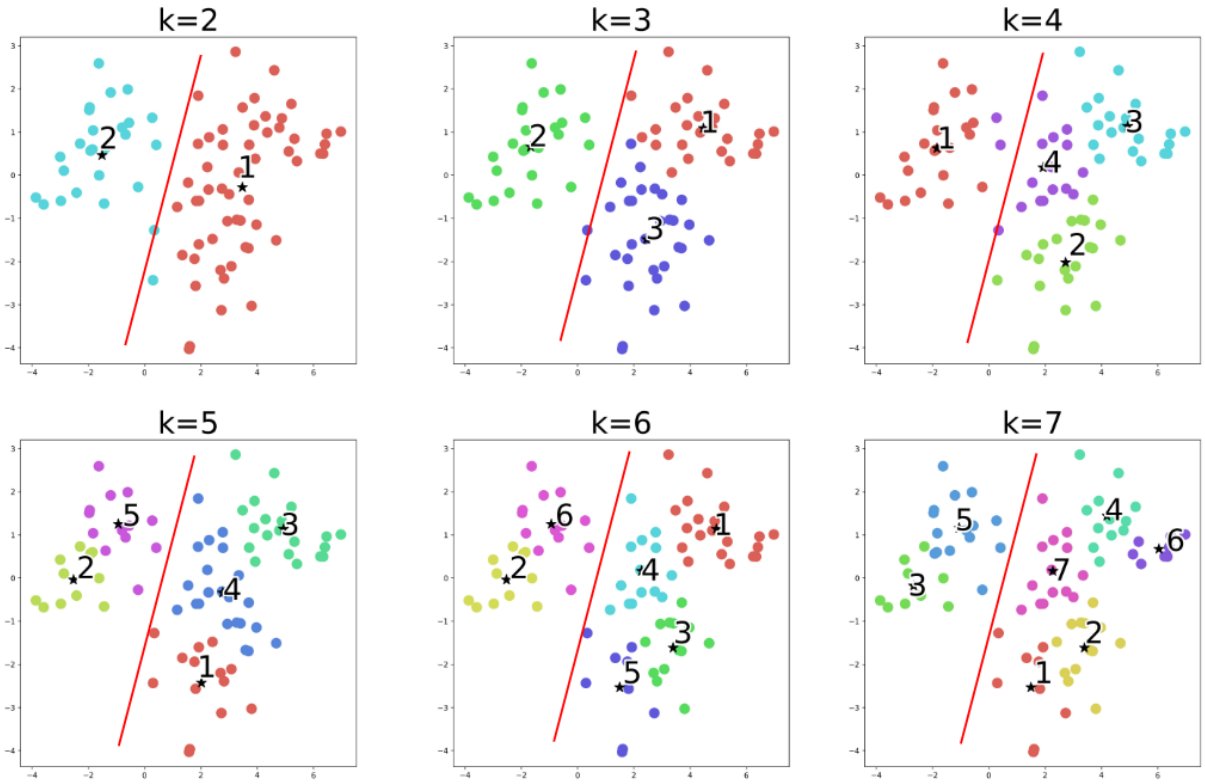


Figure 7 Clustering results with different *k* values. The red line denotes the approximate boundary between the two primary clusters.

The clusters should have as significant differences as possible, be self-contained, and be coherent. Also, the clusters shouldn't be too large to make reasonable explanations and easily be generalized. Accordingly, k =5 is chosen for the k-means cluster algorithm, and the result of clustering is shown in Figure 8.

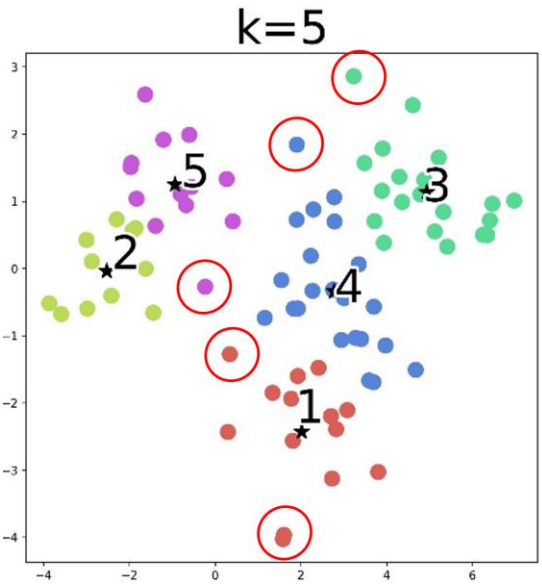


Figure 8 Clustering results with *k*=5. The black stars denote the centroid of each cluster. The distance between clusters 2&5 and 1&3&4 is relatively apparent. The red circles indicate the latent outliers.

In Figure 8, the results of *k* =5 are shown in detail, with the clusters centroids and latent outliers marked. We notice that some latent outliers are lying approximately at the boundaries between multiple clusters, which means those data points' features may not be as clear as those data points cluster around the centroids, or these data points have features of two adjacent clusters and hence are ambiguous.

Clustering results

As described in the earlier sections, the clustering is based on the daily delay profile images. Thus, the spatial-temporal feature of daily delay distribution could be represented. After processing the image feature recognition algorithm Resnet 50 and dimensionality reduction, the daily delay patterns of 79 days are clustered into five types, as shown in Table 3. For each cluster, the centroid is calculated iteratively in the k-means algorithm until stable. We define the image closest to the cluster centroid as a "centroid image," which could best represent the feature of the cluster members. The corresponding images of the cluster centroid and elements in each cluster are listed in the appendix, followed by the analysis of each centroid image. The capital letter with number in Table 3 "spatial imbalance distribution," denotes the auxiliary lines in the appendix figures, representing the platforms' locations or times with significant delay dynamics.

The most significant feature difference among the five types of delay patterns is that each pattern has a specific day-of-week distribution. We found that there exist two main types of delay patterns: "**weekend delay"** (represented by clusters 2 and 5) and "**weekday delay"** (represented by clusters 1,3,4). This feature can be identified in Figure 9. The number and color in the blocks denote the number of days belonging to the specific day and cluster. Most of the daily delay patterns that occur on weekends belong to the "weekend delay", while most of the delay that happens on weekdays belongs to one in three "weekday delay" clusters. Also, we found that more than half of Monday's daily delay distribution has a similar feature as cluster 3, and clusters 1 and 4 occur more frequently on Thursday and Friday.

Looking at the cluster-level features, we found a significant feature difference among the five types of delay patterns. These clusters have distinctive combinations of imbalance distribution

on space, time, line directions, and corresponding delay severity. The feature differences are described in Table 3.

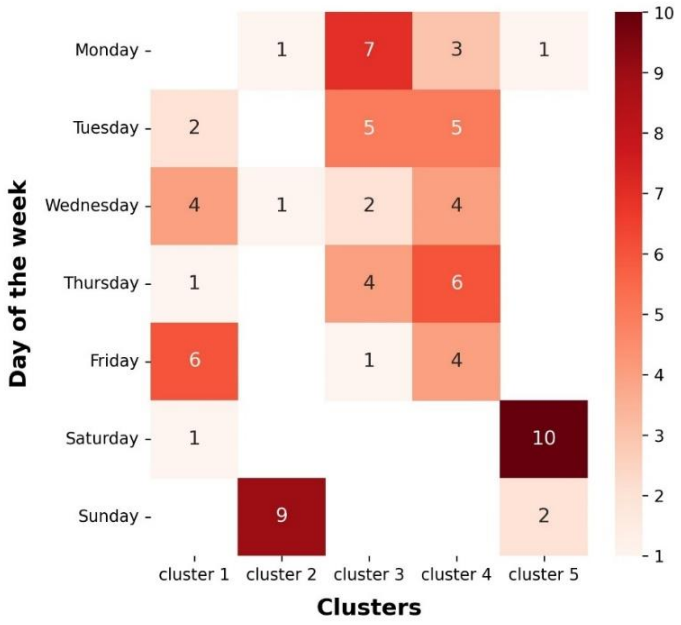


Figure 9 Cluster distribution on the day of the week

Table 3 Clustering results and attributes of delay pattern of each cluster

Class		**"Weekday delay"**			**"Weekend delay"**	
Cluster (count)		**Cluster 1 (14)**	**Cluster 3 (11)**	**Cluster 4 (22)**	**Cluster 2 (20)**	**Cluster 5 (13)**
Feature of the delay distribution	**Temporal imbalance distribution**	The heavy delay occurs during the evening peak hours	The heavy delay occurs between the morning peak end and evening peak end, from 9 am to 7 pm	Heavy delay evenly distributes all-day	Heavy delay evenly distributes all-day	Delay occurs at intervals
	Directional imbalance distribution	Heavier delay on direction 1	Slightly heavier on direction 1	Heavier delay on direction 1	Heavier delay on direction 1	More delay on direction 1; More early arrive at direction 2
	Spatial imbalance distribution	Always punctual between C2 and C2 (Delft)	Evenly distribute	Always delay near Den Haag Kurhaus terminal (Den Haag, Scheveningen) More early arrive between C1 and D (Delft)	Always delay between A1 and Den Haag Kurhaus terminal (Den Haag)	Always arrive on time between C2 and D (Delft); Always delay between A1 and Den Haag Kurhaus terminal (Den Haag)
Mean of statistics of punctuality	**Average of daily delay**	28	3	13	**48**	**45**
	std	123.93	100.47	109.67	119.15	113.11
	25%	-39	-52	-45	-18	-16
	50%	18	6	11	34	34
	75%	85	54	65	100	98
Day type distribution	**Workday**	**92.9%**	**100%**	**100%**	18.2%	7.7%
	Weekend	7.1%	0%	0%	**81.8%**	**92.3%**
	Significant frequently	Even in **weekday**	**Monday**	Even in **weekday**	**Sunday**	**Saturday**

Moreover, the percentile values of the average daily punctuality of each cluster are illustrated in Figure 10. It is obvious that the "weekend delay" (cluster 2 and 5) have a higher delay time than the "weekday delay" (cluster 1,3,4) on all percentiles as well as a higher average daily delay. From this, we can conclude that the "weekend delay" patterns generally exhibit relatively more serious delays than the "weekday delay". Combined with Figure 11, we can see that there is a distinct periodic fluctuation in daily punctuality. The daily average punctuality has a significant peak almost every weekend and reaches the valley almost every Monday.

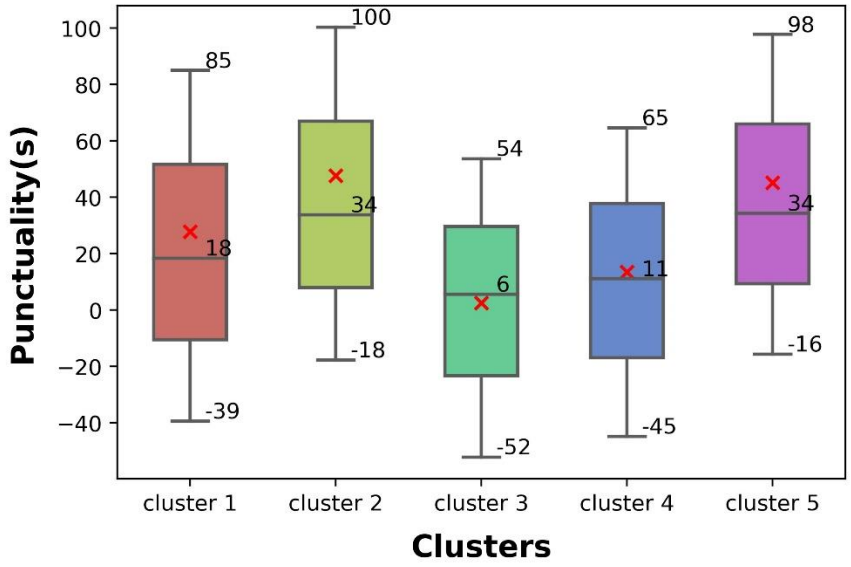


Figure 10 Distribution of daily punctuality values in each cluster. The lines within each box from top to bottom represent the 75%, 50%, and 25% percentile of punctuality. The red dots denote the mean value of average daily punctuality for each cluster.

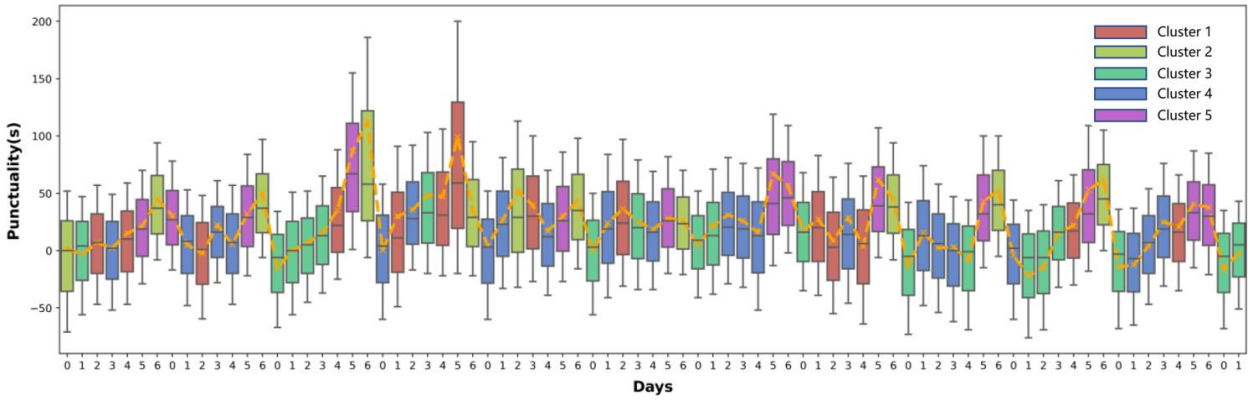


Figure 11 The daily delay pattern and punctuality statistics of each day. The orange dash line denotes the mean of daily punctuality. The color of boxplots denotes the clustering result of each day. The x-axis denotes the day of the week ('0' represent Monday), while the y-axis denotes the punctuality value in seconds.

According to the analysis in this section, the prominent findings could be summarized as follows:

1) There are five types of daily delay patterns with distinct characteristics for tram line 1.
2) The delays on Monday, Saturday, and Sunday have significant distinct daily delay patterns. Weekends usually have more severe delays than weekdays. Tram line 1 is the most punctual on Monday, and least punctual on Sunday with a higher possibility of severe delay hampering large areas.
3) Direction 1(from "Den Haag Kurhaus" to "Delft, Tanthof") usually has a more serious delay than direction 2 (from "Delft, Tanthof" to "Den Haag Kurhaus"). And direction 2 has earlier arrival than direction 1.
4) The delay that occurs in the morning peak usually exceeds 9 am, and the delay that occurs in the evening peak usually exceeds 7 pm, although the morning and evening peak hours generally, tend to end after these two time points.
5) The tram usually arrives earlier at stops in Delft than at stops in the Hague.
6) A latent boundary on the daily delay profile exists at the location of stop 'Den Haag Frankenslag' in both directions, where the location is also the approximate boundary between the Den Haag Scheveningen (near the beach and the terminal of the tram line) and Den Haag downtown area. The spread of delay may be disturbed (intensified or weakened). This means the delay pattern of the Den Haag Scheveningen and Den Haag downtown area is different. The same boundary also exists at 'Delft station. This can be caused by the onboard/arriving passengers at those stops, especially at transit hubs, or caused by the land use pattern, which leads to less or more delay.

CONCLUSION

In this paper, we implemented the k-means clustering on the daily punctuality information of tram line 1 in the Hague. The patterns of daily delay distribution are detected, extracted, and clustered according to the proposed methodology. 79 days of daily delay profile images are clustered into **five** types with different spatial-temporal delay distribution features. The case study results indicate distinct weekdays and weekend patterns. The data-driven explorative analysis proposed in this research can make significant contributions to PT operators and planners. Firstly, such an analysis technique could elucidate the operator's general understanding of the regularity of delay occurrence on a specific line. Based on the new perception of delay characteristics, precise improvement of PT management could be conducted and evaluated. Besides, clustering could provide the researchers with prior knowledge of typical delay patterns of PT networks. Furthermore, the proposed methodology can easily be extended for other transit lines and other networks with GTFS data and AVL data and can explore and extract more abstract delay pattern characteristics.

There are several limitations to this study. *K*-means algorithm could have limited validation opportunities on high-dimensional data, which is a common issue for the distance-based algorithm. In this research, the dimensionality reduction methods, PCA and t-SNE, are implemented for simplification. More possible solutions to the dimensionality of image input could be implemented and compared, such as the sub-spacing method. Besides, the k-means algorithm is not the only choice for clustering. We found that the clustering result in this research has ambiguity due to the outliers at the boundaries of the clusters. The more advanced technique like density-based clustering algorithms (DBSCAN) algorithm could solve this problem. Finally, the current analysis

only provides a qualitative analysis. This could only provide public transport operators with a general insight into delay characteristics and could be less suitable for practice.

We envision four potential directions for further research. First, Automatic Passenger Count (APC) data could be incorporated to estimate and analyze the average passenger delay per passenger. This delay can more closely represent the actual delay that the passengers experienced in their commute. Second, extending the methodology for network level instead of line level analysis. This allows for understanding delay propagation at a city or regional scale and allows the operator to gain insight for providing a recommendation for PT network management advice. Third, to provide a more quantitative analysis of each delay pattern, a more explainable clustering algorithm is required. The public transport operators need to not only know different delay patterns but also, more specific characteristics of these patterns to assist them to implement targeted operational strategies more precisely. A more explainable clustering algorithm will make a clearer connection between the criterion of image cluster decision and delay characteristics, by providing the statistical description of these characteristics. Finally, the image representation of delay patterns proposed in this research allows us to efficiently collect delay patterns from a longer period, thus creating a rich fused dataset. This opens various possibilities from the computer vision domain for a comprehensive understanding of long-term delay patterns of PT networks which in turn can be used for predicting delay propagation.

ACKNOWLEDGMENTS

We would like to thank NDOV for providing the AVL data.

AUTHOR CONTRIBUTIONS

The authors confirm their contribution to the paper as follows: study conception and design: Panchamy Krishnakumari, Yuxing Cheng; Data source: Panchamy Krishnakumari; Analysis, interpretation of results, draft manuscript preparation: Yuxing Cheng; Revision: Panchamy Krishnakumari and Yuxing Cheng. Both authors reviewed the results and approved the final version of the manuscript.

REFERENCES

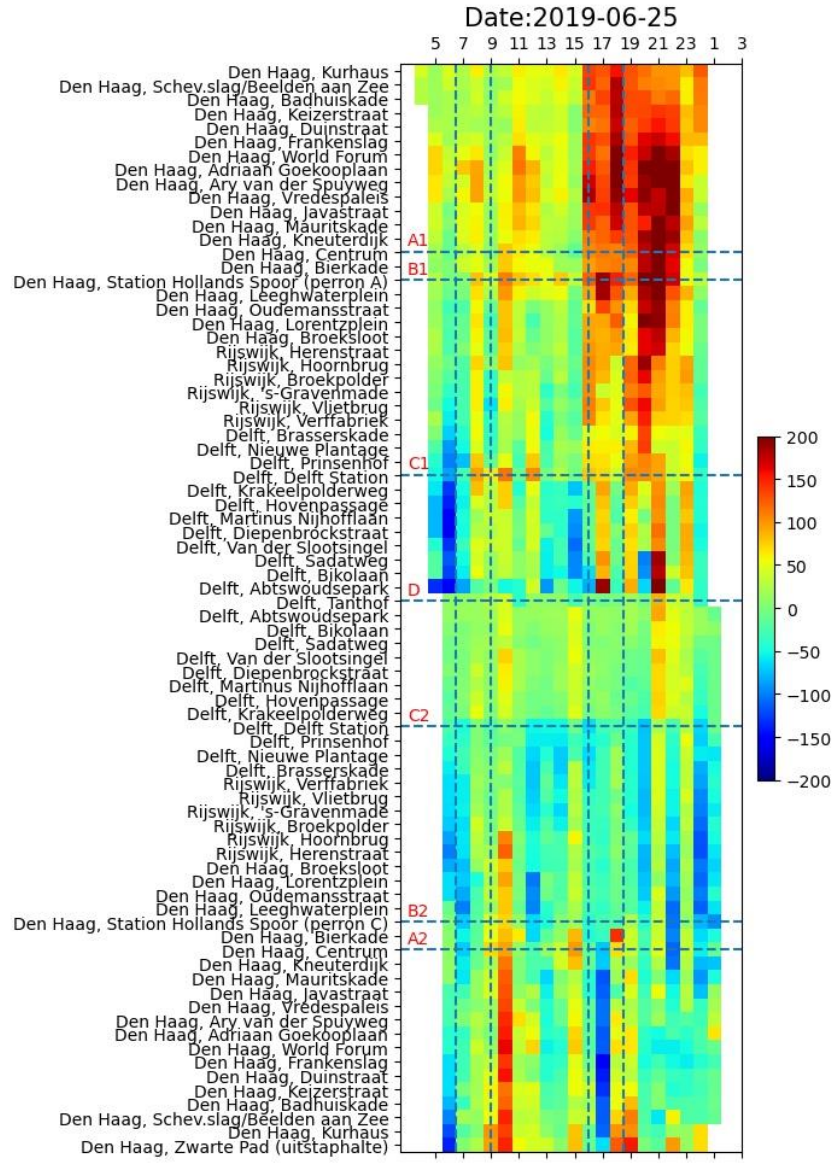
1. Ceder, A. *Public Transit Planning and Operation: Modeling, Practice and Behavior, Second Edition*. CRC Press, 2016.
2. Luo, D., L. Bonnetain, O. Cats, and H. van Lint. Constructing Spatiotemporal Load Profiles of Transit Vehicles with Multiple Data Sources. *Transportation Research Record: Journal of the Transportation Research Board*, Vol. 2672, No. 8, 2018, pp. 175–186. https://doi.org/10.1177/0361198118781166.
3. Degeler, V., L. Heydenrijk-Ottens, D. Luo, N. Oort, and J. W. C. Lint. Unsupervised Approach towards Analysing the Public Transport Bunching Swings Formation Phenomenon. *Public Transport*, Vol. 13, 2021, pp. 1–23. https://doi.org/10.1007/s12469-020-00251-z.
4. Wong, J. Leveraging the General Transit Feed Specification for Efficient Transit Analysis. *Transportation Research Record*, Vol. 2338, No. 1, 2013, pp. 11–19. https://doi.org/10.3141/2338-02.
5. Szymanski, P., M. Zolnieruk, P. Oleszczyk, I. Gisterek, and T. Kajdanowicz. Spatio-Temporal Profiling of Public Transport Delays Based on Large-Scale Vehicle Positioning Data from GPS in Wrocław. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 19, No. 11, 2018, pp. 3652–3661. https://doi.org/10.1109/TITS.2018.2852845.
6. Krishnakumari, P., O. Cats, and H. van Lint. Estimation of Metro Network Passenger Delay from Individual Trajectories. *Transportation Research Part C: Emerging Technologies*, Vol. 117, 2020, p. 102704. https://doi.org/10.1016/j.trc.2020.102704.
7. Liu, T. L. K., P. Krishnakumari, and O. Cats. Exploring Demand Patterns of a Ride-Sourcing Service Using Spatial and Temporal Clustering. 2019.
8. Bapaume, T., E. Côme, J. Roos, M. Ameli, and L. Oukhellou. Image Inpainting and Deep Learning to Forecast Short-Term Train Loads. *IEEE Access*, Vol. 9, 2021, pp. 98506–98522. https://doi.org/10.1109/ACCESS.2021.3093987.
9. Nguyen, T. T., P. Krishnakumari, S. C. Calvert, H. L. Vu, and H. van Lint. Feature Extraction and Clustering Analysis of Highway Congestion. *Transportation Research Part C: Emerging Technologies*, Vol. 100, 2019, pp. 238–258. https://doi.org/10.1016/j.trc.2019.01.017.
10. Agard, B., V. P. Nia, and M. Trépanier. ASSESSING PUBLIC TRANSPORT TRAVEL BEHAVIOUR FROM SMART CARD DATA WITH ADVANCED DATA MINING TECHNIQUES. 2013, p. 13.
11. MacQueen, J. Some Methods for Classification and Analysis of Multivariate Observations. *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics*, Vol. 5.1, 1967, pp. 281–298.
12. Hopkins, B., and J. G. Skellam. A New Method for Determining the Type of Distribution of Plant Individuals. 1954. https://doi.org/10.1093/OXFORDJOURNALS.AOB.A083391.
13. Banerjee, A., and R. N. Dave. Validating Clusters Using the Hopkins Statistic. No. 1, 2004, pp. 149–153 vol.1.
14. Aggarwal, C. C. Cluster Analysis. In *Data Mining: The Textbook* (C. C. Aggarwal, ed.), Springer International Publishing, Cham, pp. 153–204.
15. Rousseeuw, P. J. Silhouettes: A Graphical Aid to the Interpretation and Validation of Cluster Analysis. *Journal of Computational and Applied Mathematics*, Vol. 20, 1987, pp. 53–65. https://doi.org/10.1016/0377-0427(87)90125-7.
16. Giraud, C. Introduction to High-Dimensional Statistics. p. 361.

17. Solem, J. E. *Programming Computer Vision with Python: Tools and Algorithms for Analyzing Images*. O'Reilly Media, Inc., 2012.
18. Picon, A., A. Alvarez-Gila, M. Seitz, A. Ortiz-Barredo, J. Echazarra, and A. Johannes. Deep Convolutional Neural Networks for Mobile Capture Device-Based Crop Disease Classification in the Wild. *Computers and Electronics in Agriculture*, Vol. 161, 2019, pp. 280–290. https://doi.org/10.1016/j.compag.2018.04.002.
20. Guérin, J., and B. Boots. Improving Image Clustering With Multiple Pretrained CNN Feature Extractors. http://arxiv.org/abs/1807.07760. Accessed Jul. 29, 2022.
21. Asano, Y. M., C. Rupprecht, and A. Vedaldi. Self-Labelling via Simultaneous Clustering and Representation Learning. http://arxiv.org/abs/1911.05371. Accessed Jul. 29, 2022.
22. He, K., X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. 2016.
23. Tr, T. Dimensionality Reduction: A Comparative Review. p. 36.
24. Maaten, L. van der, and G. Hinton. Visualizing Data Using T-SNE. *Journal of Machine Learning Research*, Vol. 9, No. 86, 2008, pp. 2579–2605.
25. Noor Mathivanan, N. M., N. A. Md.Ghani, and R. Mohd Janor. A Comparative Study on Dimensionality Reduction between Principal Component Analysis and K-Means Clustering. *Indonesian Journal of Electrical Engineering and Computer Science*, Vol. 16, No. 2, 2019, p. 752. https://doi.org/10.11591/ijeecs.v16.i2.pp752-758.

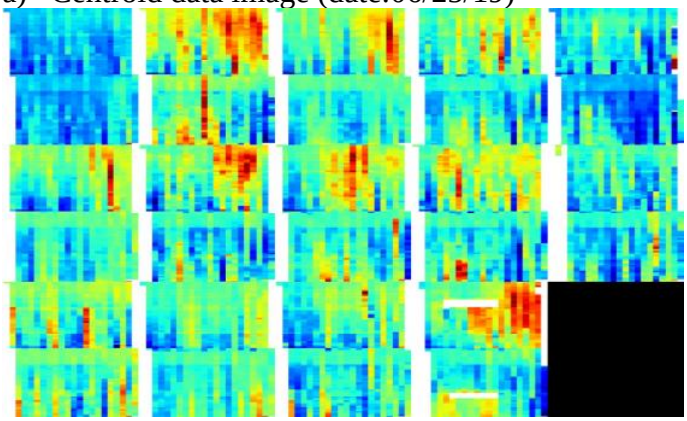
APPENDIX A: THE CLUSTERING RESULTS OF *K*-MEANS ALGORITHM

Figures in the appendix illustrate the daily delay profile images of each cluster in detail. For each figure, subfigure a) is the "centroid image" (the daily delay profile image closest to the centroid), which could best represent the cluster features. The subfigure b) are all elements in the cluster. Few of the elements in subfigure b) may not have the obvious same feature as other elements (correspond to the outliers denoted in Figure 8), as the outliers are not always avoidable.

To better identify the daily delay distribution, the auxiliary dash blue lines are drawn in each subfigure a). The horizontal lines (A1, B1, C1 and A2, B2, C2) represent the location of stops at train stations ("Den Haag, Centrum", "Den Haag, Station Hollands Spoor", "Delft, Delft Station") and the terminal station (blue dash line D, "Delft, Tanthof"). The same capital letter of auxiliary lines represents the same location, and the number after the capital letter represents a different platform on the tram line direction 1 or 2. (Direction 1 is from "Den Haag Kurhaus" to "Delft, tanthof", and direction 2 is from "Delft, tanthof" to "Den Haag Kurhaus").



a) Centroid data image (date:06/25/19)

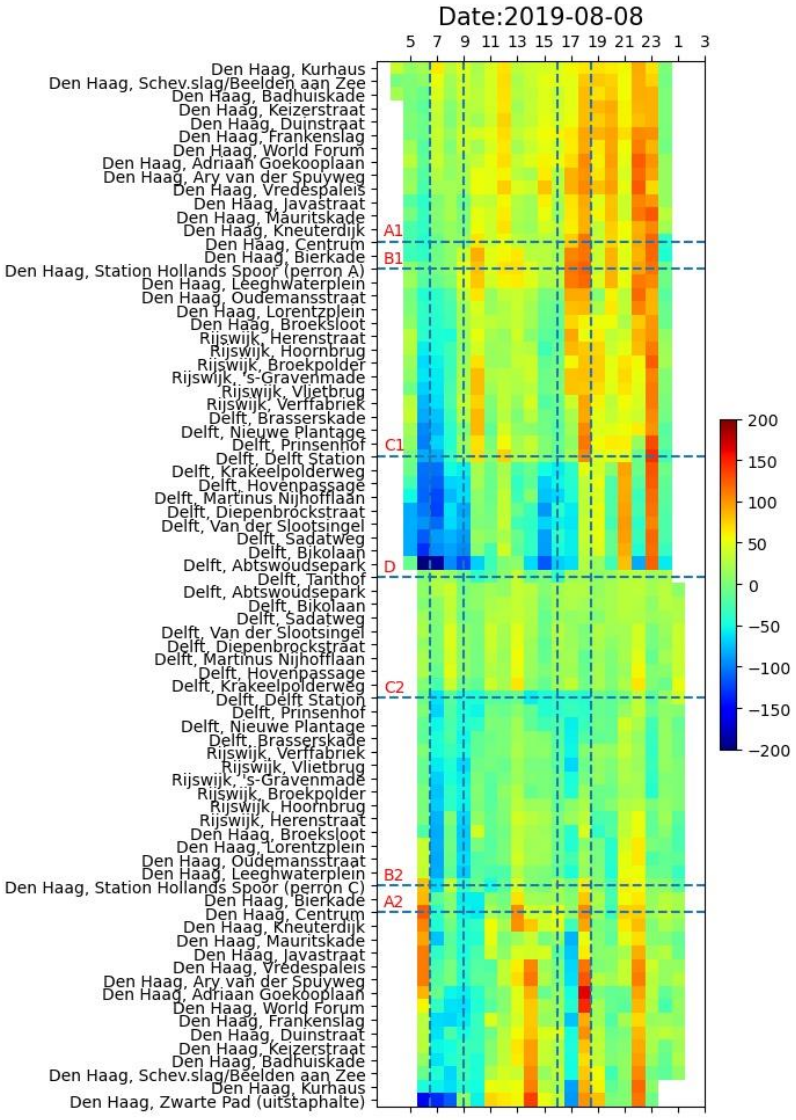


b) All elements in cluster 1

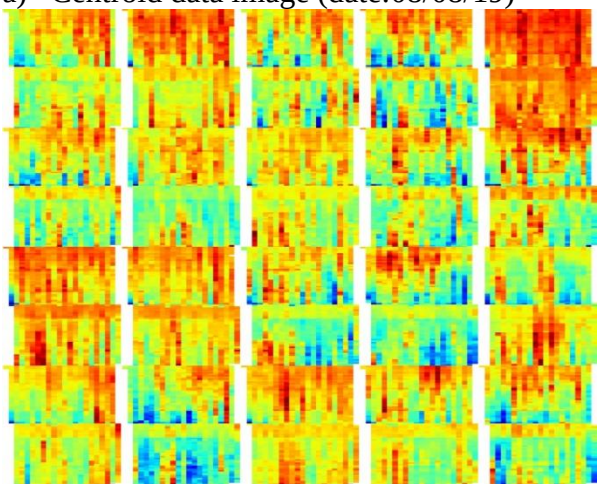
1 **FIGURE A-1 The centroid data and all elements image of cluster 1**

Feature:

1) **Temporal imbalance distribution:** heavy delay occurs during the evening peak hours
2) **Directional imbalance distribution:** heavier delay on direction 1
3) **Spatial imbalance distribution:** Always punctual between C2 and C2 (Delft)



a) Centroid data image (date:08/08/19)

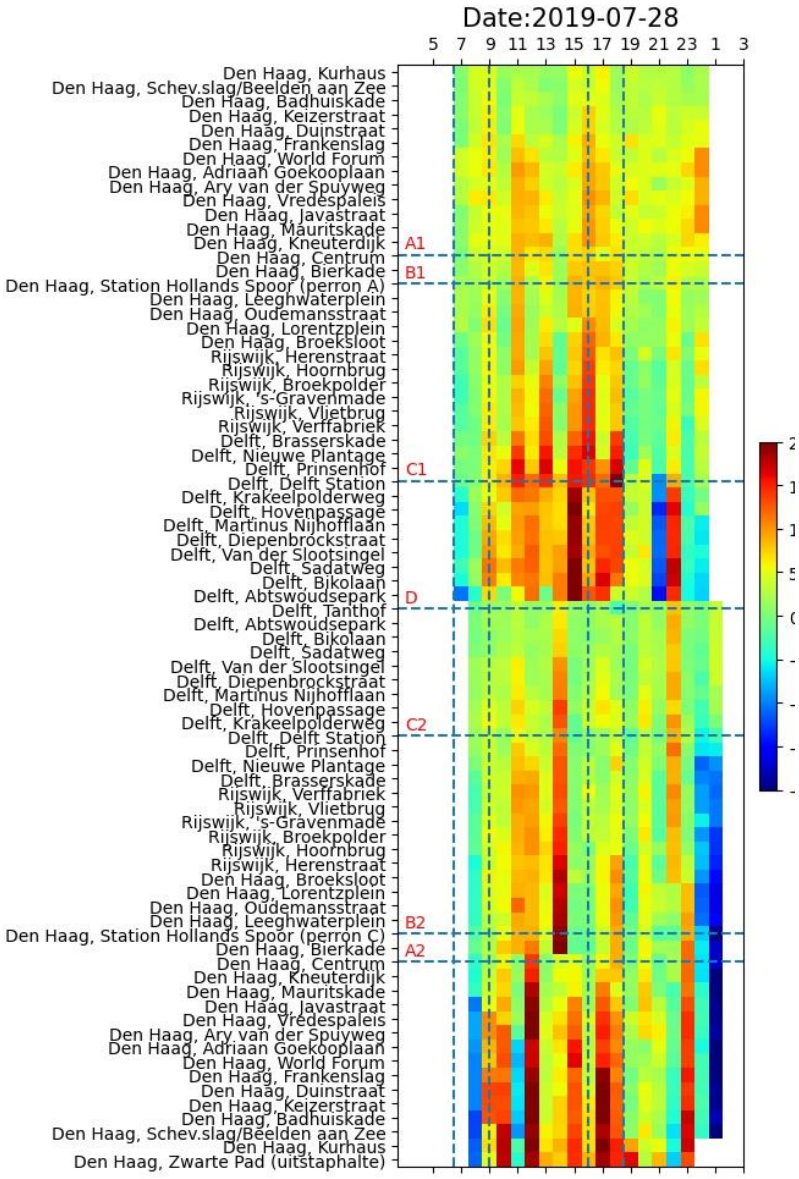


b) All elements in cluster 2

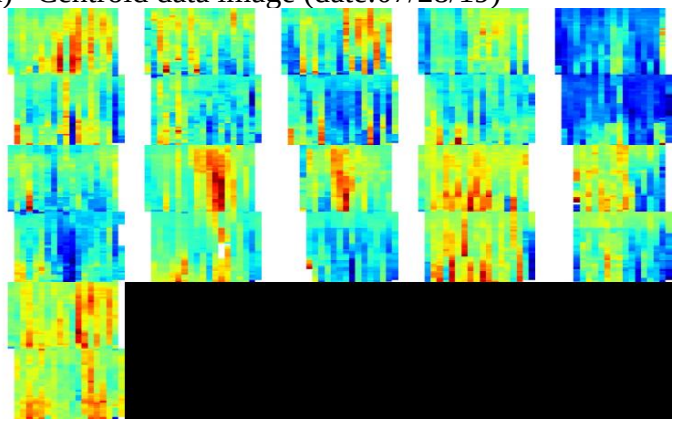
1 **FIGURE A-2 The centroid data and all elements image of cluster 2**

Feature:

1) **Temporal imbalance distribution:** Heavy delay evenly distributes all day
2) **Directional imbalance distribution:** Heavier delay on direction 1
3) **Spatial imbalance distribution:** Always delay between A1 and Den Haag Kurhaus terminal (Den Haag)



a) Centroid data image (date:07/28/19)

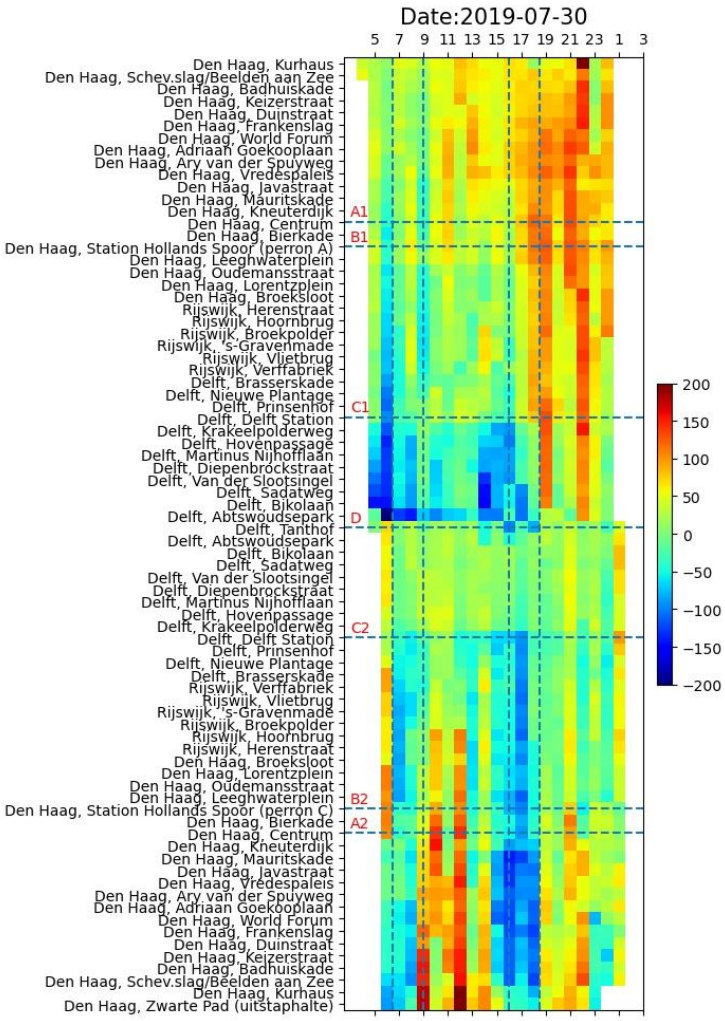


b) All elements in cluster 3

FIGURE A-3 The centroid image and all elements of cluster 3

Feature:

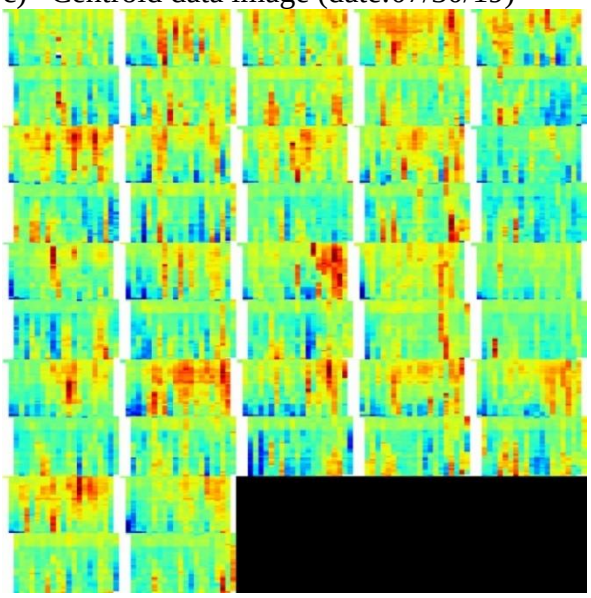
1) **Temporal imbalance distribution:** Heavy delay occurs between the morning peak end and evening peak end, from 9 am to 19 pm
2) **Directional imbalance distribution:** Slightly heavier on direction 1
3) **Spatial imbalance distribution:** Evenly distribute



Feature:

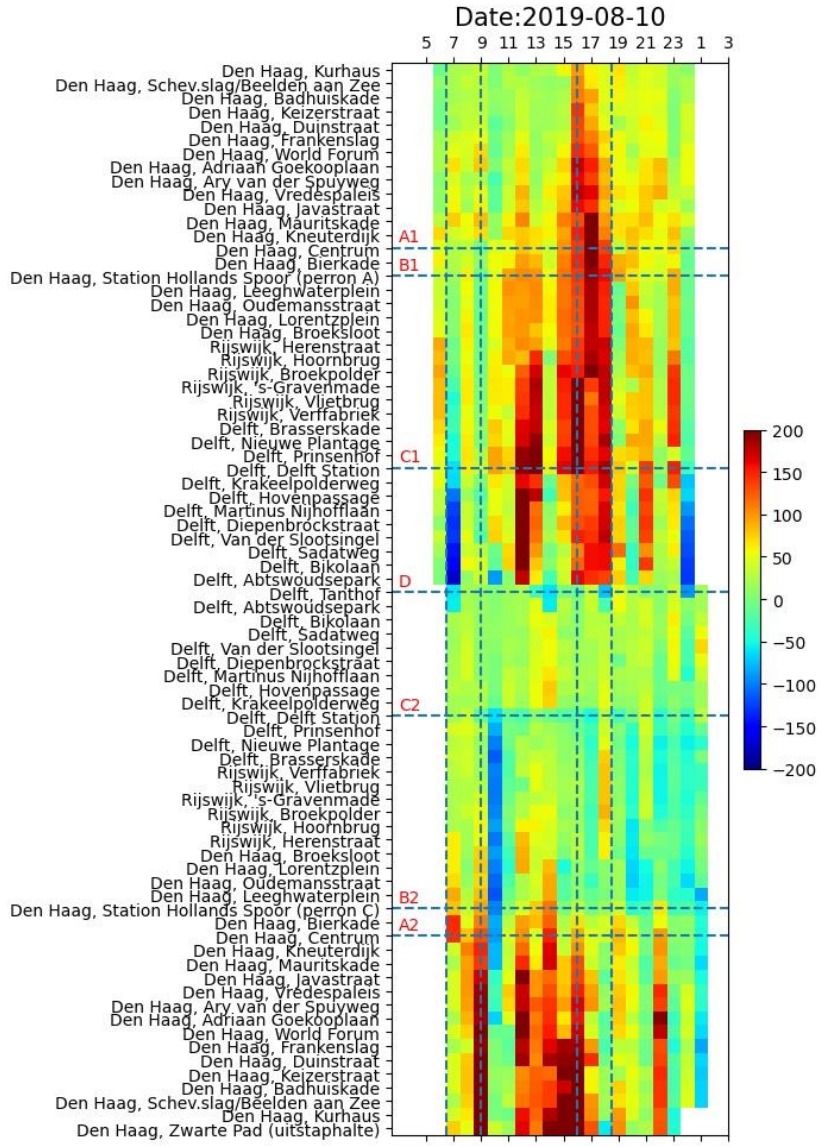
1) **Temporal imbalance distribution:** Heavy delay evenly distributes all day
2) **Directional imbalance distribution:** Heavier delay on direction 1
3) **Spatial imbalance distribution:** Always delay near Den Haag Kurhaus terminal (Den Haag, Scheveningen) More early arrive between C1 and D (Delft)

c) Centroid data image (date:07/30/19)

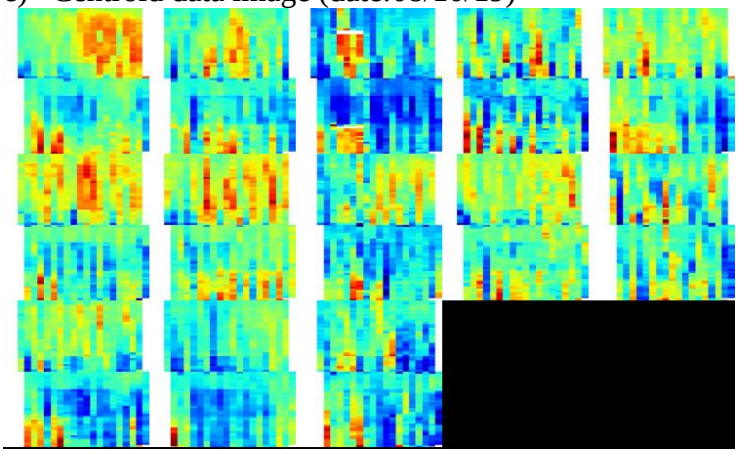


d) All elements in cluster 4

FIGURE A-4 The centroid image and all elements of cluster 4



e) Centroid data image (date:08/10/19)



f) All elements in cluster 5

1 **FIGURE A-5 The centroid image and all elements of cluster 4**

Feature:

1) **Temporal imbalance distribution:** Delay occurs at intervals
2) **Directional imbalance distribution:** More delay on direction 1; More early arrive at direction 2
3) **Spatial imbalance distribution:** Always arrive on time between C2 and D (Delft); Always delay between A1 and Den Haag Kurhaus terminal (Den Haag)