INSIST

Deliverable

D2.1.1 Methods & Algorithms for Traffic Density Measurement

Editor:

[Çiğdem Çavdaroğlu]

ITEA 3 Project 13021

Date : 19.12.2016

Document properties

Security	Confidential
Version	Version 1.0
Author	Çiğdem Çavdaroğlu
Pages	

History of changes

Version	Author, Institution	Changes
0.1	Çiğdem Çavdaroğlu, KoçSistem	Initial Version
0.2		
0.3		
0.4		

Abstract

This document describes existing traffic density measurement methods and algorithms, compares this existing methods, and finally suggests the most suitable ones. In order to measure traffic density information, implementing traffic surveillance methods was planned in the initial version of INSIST project (2013). When the project is started in the beginning of 2016, two new methods were introduced. This document covers information about all these three methods.

Table of contents

1. Executive Summary

The main purpose of this deliverable is to provide information about traffic-related data collection methods, traffic-related knowledge extraction, and traffic-related data integrity.

The main goal of the second work package (WP2: Sensing and feature extraction algorithms) of the INSIST project is to develop a set of tools and algorithms to capture and analyze data from a visual sensor or a set of homogeneous visual sensors which are monitoring the environment where traffic monitoring and advertising systems are installed in the surrounding.

In January 2016, when the project was started, KoçSistem (the leader of the WP2) has decided to enhance the methods which were used in order to collect traffic-related data due to several reasons. The first reason is to acquire alternative or additional methods which will be used to ensure data integrity. The second reason is to collect data from the places where a visual sensor is unavailable.

Although there are already various traffic surveillance cameras around cities, it is difficult or impossible to access the data of these cameras. On the other hand, it is forbidden to set up a traffic surveillance camera at a point in the city. Therefore, we are limited to use the surveillance cameras around our own building.

The salient information of the observed scenes will be extracted and described, so that objects and related information can be effectively detected. This object data is then used by the considered application (e.g. traffic analysis to control street lighting) and/or it can be shared with other systems (city and safety control) for further processing and aggregation to develop integrated hybrid applications.

Second work package receives input from WP1 (Requirements & system definition) in the form of requirements and uses case definitions, and will provide input for WP3 (Integrated Connected Systems for Citizen's Safety & Comfort) and WP4 (Infrastructure & Data Representation), where integration activities between different systems will be carried out and various sensor modalities are combined.

In this deliverable, both a short literature review and existing methods survey will be provided for the selected sensors:

- Visual sensors
- Social sensors
- Map sensors

2. Traffic-Related Data Collection Methods

In order to measure traffic density information, implementing traffic surveillance methods was planned in the initial version of INSIST project (2013). When the project is started in the beginning of 2016, two new methods were introduced.

The goal of this enhancement is to put forward alternative traffic density estimation methods with lower cost and wider spatial coverage by utilizing one of the most popular social media sites Twitter and one of the most popular public traffic density map applications Google Map as the data sources.

- Visual Sensor: Traffic surveillance systems, traffic cameras (planned in 2013)
- Social Sensor: Social media channels (decided in 2016)
- Map Sensor: Public traffic density map applications and services (2016)

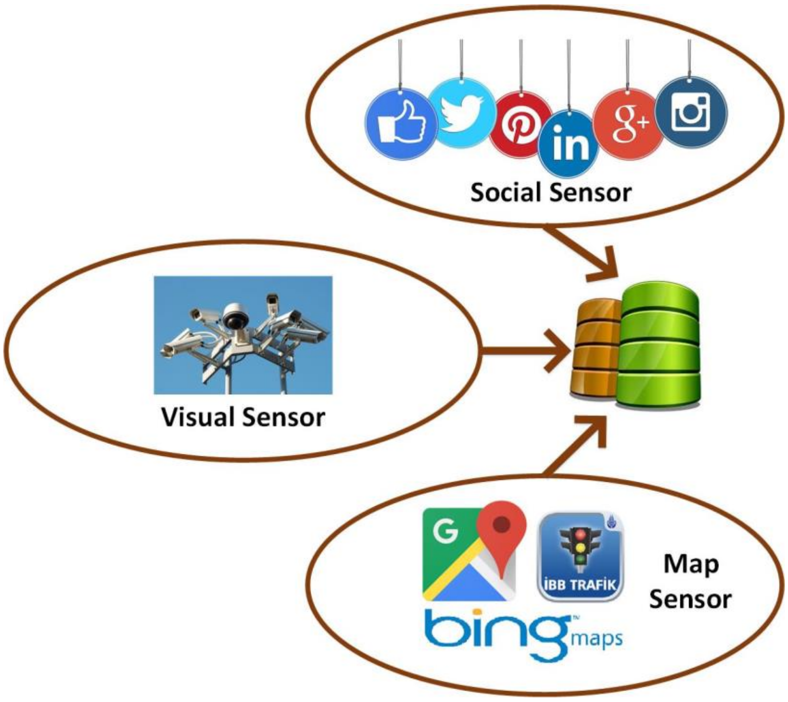

Figure 1 Traffic Related Data Collection Methods

2.1. Traffic-Related Data Extraction and Collection Methods

The main goal to enhance traffic collection methods is to acquire additional methods which will be used to ensure data integrity, and to collect data from the places where a camera surveillance is unavailable.

Each method collects traffic-related data and some additional data e.g. location, event, and etc.

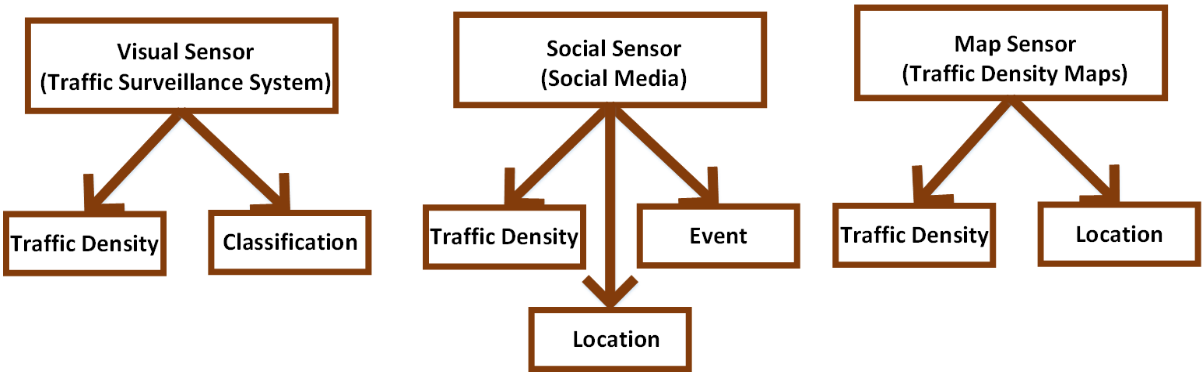

Figure 2 Data Collection Methods and Collected Data

2.2. Visual Sensor

2.2.1. Difficulties and Disadvantages of the Traffic Surveillance Systems

In this section, various difficulties and disadvantages of the traffic surveillance systems will be discussed.

2.2.1.1. Accessibility

Although there are already various traffic surveillance cameras around cities, it is difficult or impossible to access the data of these cameras. Some of the city cameras publish traffic video data streams as public, however these data have very low resolution and are insufficient to be processed.

2.2.1.2. Set Up Traffic Surveillance Systems

On the other hand, it is forbidden to set up a traffic surveillance camera at a point in the city. Therefore, project is limited to use the surveillance cameras around our own building. A dependency to the camera hardware may limit the project.

2.2.1.3. High Cost of Deploying and Maintaining

It is both expensive and impractical to monitor every point in the city by traffic surveillance cameras. The total cost of these systems are very high. The high cost of deploying and maintaining these equipment largely limits their spatial-temporal coverage.

2.2.1.4. Risky Aspects of Cameras

Traffic cameras are subject to damage caused by weather. Heat, wind, rain, snow and ice can all damage or ruin a traffic security camera. Since they are placed on busy roads and intersections, there is also a chance that accidents could damage traffic cameras.

2.2.1.5. Unstable Working Conditions

Another disadvantage of traffic cameras is that they often do not work correctly. The camera and recording system may not be maintained properly. They may require to be checked daily.

2.2.2. Common Methods in the Literature

In order to design and intelligent traffic system, there are commonly used methods in the literature such as fuzzy based controllers, and morphological edge detection techniques. By using these methods it is possible to measure traffic density by correlating the live traffic image with a reference image. The higher the difference is, higher traffic density is detected. The methods based on neural networks identify the vehicles and traffic density by processing traffic videos. By comparing two images, traffic load can be measured. These two images are the reference image, and the live traffic image. Traffic related algorithms usually implement image segmentation techniques and noise removing operations [12].

2.2.3. Proposed Method

The general system model of the proposed method is divided into four parts. The first part processes the video signal and image acquisition from fixed camera. The second part selects the target area where the vehicles could be present by using image cropping technique. The third part detects the defined objects by enhancing the features of the image. The last part counts the density (the number of vehicles).

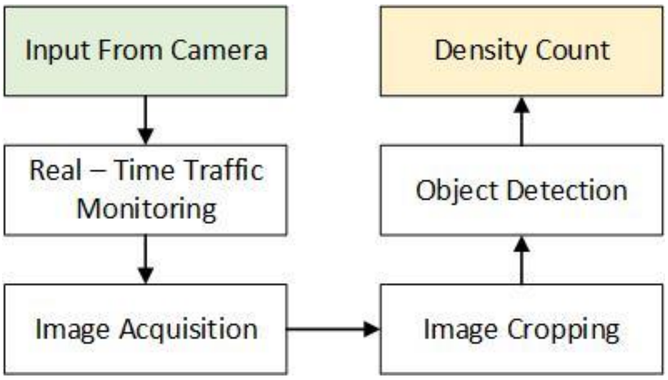

Figure 3 General System Model of the Video Processing [12]

The proposed method starts with an image acquisition process. During image acquisition process, the live video is processed by the stationary camera, mounted on any pole. It extracts one frame per second continuously from the live video. Each frame is processed by converting it into grayscale. An empty road image is selected as a reference image. An empty road image represents the case of "no traffic on the road". In the second step, image is cropped in order to select the area where the vehicles are present. This step filters out unnecessary surrounding information. In the next step, objects are determined in live video by taking the absolute difference of each extracted frame with the reference image. The presence of the objects is enhanced by binarization of the difference image. In the final step traffic density is calculated in the desired target area by counting the number of vehicles in the selected area.

The different lighting conditions at different times of the day cause noise in the images. In order to deal with this noise, a set of reference images are captured and stored at different time slots of the day. Video processing module cycles through these reference images according the current time of the day.

2.3. Alternative Methods Requirement

Due to the high cost of deploying and maintaining video surveillance equipment and the limitation of their spatial-temporal coverage, alternative solutions with lower cost and wider spatial coverage are required. Using “Social Sensors“ and “Map Sensors“ it is possible to extract traffic-related data from Twitter or processing traffic density maps in real-time. These methods may provide the required low cost and flexibility. In the next sections these alternative methods will be introduced.

Istanbul is the first largest and crowded city in Turkey. Many districts of the Istanbul encounter major traffic problems due to the various reasons. These reasons include increasing population, inefficiency infrastructure, on-going activities in the city center. Due to the fact that Istanbul is the most difficult case in Turkey, main pilots of the INSIST project is planned for Istanbul.

Today, it is common for official transportation departments of Istanbul to release real-time traffic data to the public through social media channels. “IBB Trafik KontrolMrk“ is a Twitter account operated by Istanbul Transportation Authority (IETT). This account posts real-time formal traffic information about Istanbul. “İBB Trafik Radyosu“ and “Radyo Trafik“ are other Twitter accounts which post real-time traffic information about Istanbul.

2.4. Social Sensor

With the rising popularity of social media, Twitter has become an indispensable platform for citizens and public authorities (traffic authority accounts) to obtain/share real-time information.

Social sensors will collect and analyze social media feeds which relate traffic-related information (e.g. accidents, delays, road constructions, last minute road closure from reputable sources), and city events which characterize social events of various type (e.g. music, sport, politics). The main objective is to develop a framework which is capable to process social media posts from regional traffic agencies and feed a common database with the obtained meaningful information in real-time.

In the scope of INSIST project, Twitter will be used as a social media environment. Analysis and processing take into account tweets posted by regional traffic agencies.

Extracting useful information from tweets presents some challenges.

- The provided information may be completely unstructured.
- Tweets may contain grammatical errors and abbreviations.
- Each social media channel has its own style of writing:
    - the information may be incomplete (a street name with no more information about city, etc.),
    - the information may be false
    - the information may be not credible.
- Determining geographic interpretations for place names, or toponyms, involves resolving multiple types of ambiguity.

- Determining the real location of the traffic event is a challenge task. Some tweets lack the exact location information. (e.g. “Heavy traffic to Taksim“: It is very difficult to locate which road segments are in congestion currently.)
- Data may be sparsity. Although Twitter data can cover a much larger area of the city with respect to the traffic surveillance systems, not all the traffic events can be captured from Twitter.
- Heterogeneous data: Twitter posts may be multi-typed. They may include congestion, accident, road construction, disabled cars, road closure, etc. Besides traffic events, social events such as protests, concerts, football matches may also affect traffic conditions nearby. Traffic events and social events are useful data in order to estimate non-recurrent traffic congestions. One of the challenges in this subject is to model impacts of the diverse traffic and social events on traffic congestion.
- In order to automatically recognize the relevance of tweet post for the purpose of detecting traffic events is a challenge task (e.g. the tweet "excessive speed is the main cause of car accidents in Liverpool" is not relevant for the purpose of the work to be addressed here, but it can be considered as a traffic related tweet).

Although it is possible to extract the locations of the traffic events and determine the real locations through NLP techniques, it is a challenge task and still is a hot topic within social-media mining search area. In our approach, names of the places, traffic status, and the traffic events are extracted from tweet posts, and they are integrated a NER (Named Entity Recognizer) engine. The main steps in the approach are listed below:

- collect a set of traffic-related tweets
- extract the road segment and location
- extract the status of the traffic
- extract the type of event
- geolocate the event

e.g.

“Basın Ekspres İkitelli-Başakşehir Yönü sol şerit Trafik kazası (hasarlı). 1 şerit trafiğe kapalı. Ekip Sevk Edildi.“

- extract the road segment and location: Basın Ekspres İkitelli-Başakşehir Yönü sol şerit
- extract the status of the traffic: kapalı
- extract the type of event: kaza – hasarlı
- geolocate the event: real location of the “Basın Ekspres İkitelli-Başakşehir Yönü“

In the next section, a short literature will be provided in the area of traffic-related data extraction, and event detection through crowdsourcing and social networks.

2.4.1. Definitions

Natural Language Processing (NLP): is the processing and understanding of human language by machines. Information Extraction can be considered a subclass of NLP. Information Extraction is also known as Knowledge Extraction.

Data Mining: is the process of discovering new patterns from large data sets.

Text Mining: is the data mining of text. Used tools are usually called text analytics.

Information Extraction: is the process of extracting and labeling relevant data from large data sets, usually text. Large means manually unreasonable. Information Extraction is a hot topic, and an active research area.

Unstructured Text Data: is the primary form of human-generated information. Social media posts, business reports, government reports, blogs, web pages, news, scientific literature, online reviews, and etc. Information extraction is required in order to understand data. In order to manage, search, mine, store, and utilize this data effectively, a structure should be given to it.

2.4.2. Common Known Methods in Literature

Extracting useful and meaningful facts from Twitter is a hard problem. Tweet posts often contain unstructured texts, automated posts from third-party websites, and/or temporary facts. Another difficulty in knowledge extraction is that it relies on syntactic-semantic processing, and stateof-the-art parsing models fair badly on Twitter data (Foster et al. 2011).

“Knowledge Discovery“ in data is a process of identifying valid novel potentially useful and ultimately understandable patterns in data [3].

2.4.3. Information Extraction Method

2.4.3.1. Entity

An entity is something that has a distinct, separate existence, although it need not be a material existence (Wikipedia). Main features of entities include: permanent vs transient, unique vs common, animate vs inanimate, small vs large, mobile vs sessile, place vs thing, abstract vs real, bio labels.

Entity tagging: Identifying mentions of entities (e.g., person names, locations, companies) in text.

Extracting entities and relations

- Entities: named (e.g., Person) and generic (e.g., disease name)
- Relations: entities related in a predefined way (e.g., Location of a Disease outbreak, or a CEO of a Company)
- Events: can be composed from multiple relation tuples

Common extraction subtasks

- Preprocess: sentence chunking, syntactic parsing, morphological analysis
- Create rules or extraction patterns: hand-coded, machine learning, and hybrid
- Apply extraction patterns or rules to extract new information
- Post process and integrate information: Co-reference resolution, deduplication, disambiguation

2.4.3.2. Named Entity Recognition (NER)

Named-entity recognition (NER) is also known as entity identification, entity chunking, and entity extraction. NER is a sub task of information extraction. It seeks to locate, classify and label named entities in text into pre-defined categories such as locations, traffic status, and traffic events.

The main research on NER system in INSIST project is structured as taking an unannotated block of text, such as this one:

- Küçükçekmece-Hacışerif Yönü sağ şerit Trafik kazası (hasarlı). 1 şerit trafiğe kapalı. Ekip Sevk Edildi.

And producing an annotated block of text that highlights the names of entities:

- [Küçükçekmece] LOCATION –[Hacışerif Yönü] DIRECTION [sağ şerit] LINE [Trafik kazası] [(hasarlı)] EVENT. 1 şerit trafiğe [kapalı] STATUS. Ekip Sevk Edildi.

In this example, location, direction, line, event, and status are detected and classified.

In this example, a person name consisting of one token, a two-token company name and a temporal expression have been detected and classified.

Stanford Named Entity Recognizer (CRG-NER)

Stanford NER is a Java implementation of a Named Entity Recognizer. Stanford NER has well-engineered feature extractors, and many options for defining feature extractors. Its feature extractors are for English, and particularly for the 3 classes: PERSON, ORGANIZATION, LOCATION [4].

In INSIST project, Stanford NER library is implemented to extract traffic – related information from the collected Twitter posts. It was used and was tested for Turkish language and it worked well. The pre-defined feature extractors for INSIST implementation are LOCATION, STATUS, EVENT, EVENTDETAIL.

2.4.3.3. Data Mining & Text Mining

2.4.3.3.1. Data Mining

Data mining is an information extraction activity whose goal is to discover hidden facts contained in bulk data. It is an iterative and interactive process of discovering valid, novel, useful, and understandable knowledge in massive databases.

Data mining is used to find patterns and relationships in data. Patterns can be analyzed via 2 types of models:

- Descriptive: Describe patterns and create meaningful subgroups or clusters.
- Predictive: Forecast explicit values, based upon patterns in known results.

This data become useful through knowledge discovery.

Data → Information → Knowledge → Understanding

Data mining = data fishing / data dredging (in 1960s) = data mining (in 1990s) = knowledge discovery in databases (in 1989s) = data archaeology = information harvesting = information discovery = knowledge extraction

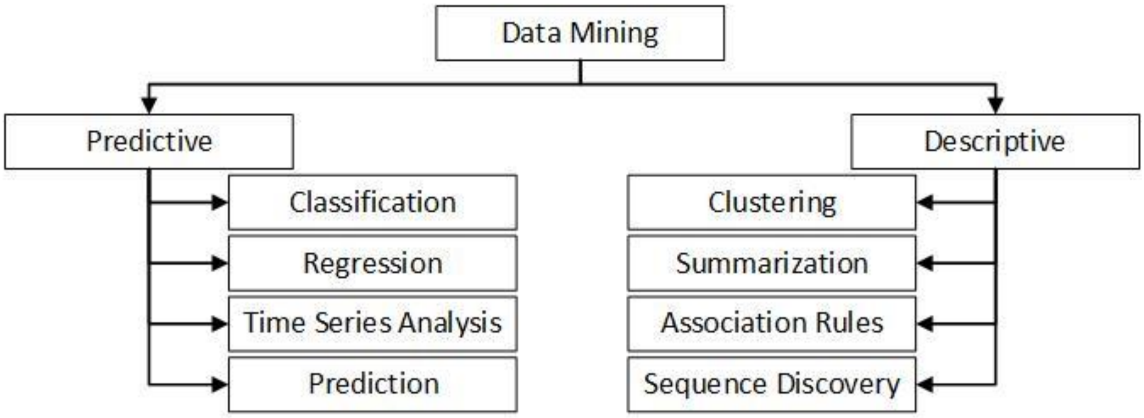

Figure 4 Data Mining Methods

2.4.3.3.1.1. Data Mining Methods

- Clustering: Find all groups and classes of objects represented in the data.
- Classification: Classify new data items using the known classes & groups. Assign data items to predetermined groups.
- Associations: Find associations and patterns among different data items. Associate data with similar relationships. The beer-diaper example is an example of associative mining.
- Artificial Neural Networks: Non-linear predictive models that learn through training and resemble biological neural networks in structure.
- Decision Trees: Hierarchical sets of decisions, based upon rules, for rapid classification of a data collection.
- Pattern Recognition
- Correlation/Trend Analysis
- Principal Component Analysis
- Regression Analysis
- Outlier/Glitch Identification
- Visualization
- Autonomous Agents
- Self-Organizing Maps (SOM): Organize information in the database based on relationships among key data descriptors.
- Link (Affinity) Analysis: Identify linkages between data items based on features shared in common.

2.4.3.3.2. Text Mining

Data mining is not the single technique for extracting knowledge from data. Text mining also can be used to extract high-quality data from texts.

Text mining is a burgeoning new field. It attempts to extract meaningful information from natural language text. Natural language text is unstructured, amorphous, and difficult to deal with algorithmically. The field of text mining deals with unstructured texts. Extracting meaningful information from texts automatically is a big challenge [7].

It may be loosely characterized as the process of analyzing text to extract information that is useful for particular purposes. In the other words, text mining equivalents to text analytics. Text mining is the process of deriving the required information with high-quality from text. High-quality information is typically derived through the devising of patterns and trends through means such as statistical pattern learning.

Text mining usually involves the process of structuring the input text. To struct the input text, parsing along with the addition of some derived linguistic features and the removal of others, and inserting them into a database subsequently are required. Parsed text data are evaluated and interpretated.

Typical text mining tasks include text categorization, text clustering, concept/entity extraction, production of granular taxonomies, sentiment analysis, document summarization, and entity relation modeling (i.e., learning relations between named entities).

Text mining consist of a broad variety of methods and technologies such as:

- Keyword – based technologies: In this technique, input is filtered as a series of character strings (keywords). A pre-defined keyword list should be selected.
- Statistics technologies: Statistics techniques refer to the systems based on machine learning. Statistics technologies leverage a training set of documents used as a model to manage and categorize text.
- Linguistic – based technologies: This method may leverage language processing systems. The output of text analysis allows a shallow understanding of the structure of the text, the grammar and logic employed.

The common feature of these different techniques is that they are all concerned with processing text in an approximate way since they are not capable to understand them [5].

2.4.3.3.3. Cognitive Technology

Cognitive technology does not guess at the meaning of words. It is designed to understand and analyze text by the methods based on a deep semantic analysis. This method has more precise, complete, and effective understanding of text as a person would [5].

2.4.3.3.4. Machine Learning

In the context of text mining, machine learning is a set of statistical techniques for identifying some aspect of text such as entities, sentiments, etc. In supervised machine learning, a model is generated and then it can be applied to other texts. In unsupervised machine learning, a set of algorithms work across large datasets to extract meaning.

2.4.3.3.4.1. Supervised Machine Learning Methods

Commonly used supervised machine learning methods are [6]

- Bayesian Networks (Bayesian Learning)
- Maximum Entropy
- Conditional Random Field
- Instance-Based Learning
- K-Nearest Neighbors
- Neural Networks / Deep Learning
- Support Vector Machine

2.4.3.3.4.2. Unsupervised Machine Learning Methods

The unsupervised methods to extract meaning from a collection of text without a pre-defined model are [6]:

- Clustering: By clustering, the groups of "like" words together into clusters of words.
- Latent Semantic ındexing: Latent semantic indexing extracts important words which occur in conjunction with each other in the text.
- Matrix Factorization: Matrix factorization allows to factor down a very large matrix into the combination of two smaller matrices. It uses latent factors. Latent factors are similarities between the words.

2.4.4. A Brief History

In Citywide Traffic Congestion Estimation with Social Media, (S. Wang, et.all, 2015) a new method named "Citywide Traffic Congestion Eestimation (CTCE)" is provided. This method propose a coupled matrix and tensor factorization model to effectively integrate rich information. The proposed traffic congestion estimation framework uses Twitter as the data source. Compared with traditional models, their method can cover a much larger area of the city with lower cost. Framework is able to extract road segments in congestion, social events, and road features to facilitate congestion estimation. The model is evaluated by comparing Chicago Transit Authority public bus GPS data. The results provide a promising performance (0.126 MAE in rush hours, around 80% accuracy on top100 congested road segments) [1].

In " Mining Complaints for Traffic-Jam Estimation: A Social Sensor Application", a novel traffic-congestion estimation model is proposed. This model utilizes the volume of messages and complaints in online social media, based on when they happen. Solution provides higher accuracy, traffic jam severity and compare the results with several baselines. The model achieves at least 38% improvement of absolute error and more than 45% improvement of relative error, when compared with a baseline that assumes linear correlation between traffic and social volume [2].

2.4.5. Information Extraction and Text Mining in INSIST

In INSIST project, Named Entity Recognition is used to extract traffic density related data from the collected tweets. INSIST used an unsupervised NER method. In order to extract traffic related events, INSIST project uses statistics technologies by implementing supervised machine learning methods (Support Vector Machine).

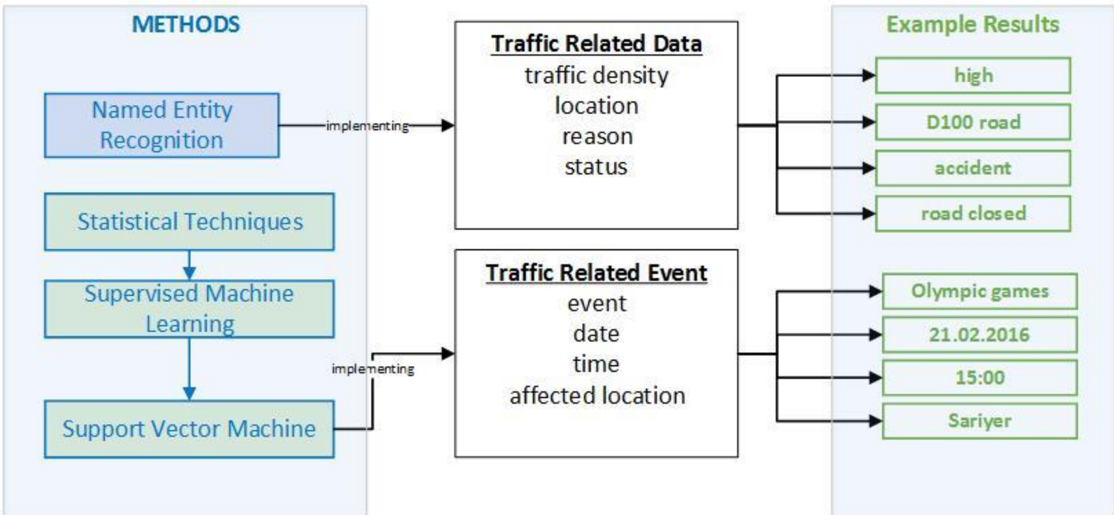

Figure 5 Information Extraction and Text Mining Methods in INSIST Project

INSIST project uses machine learning and information extraction techniques which work in tandem with pre-defined rules and patterns. These methods together generate the semantic and context information from the tweets.

The text analysis module of INSIST project consists of patterns and rules. These patterns and rules are determined by analyzing the posts of followed twitter accounts. Analysis module has a three-layered function architecture. In the low-level, initial processing functions turn unstructured text into structured data. Mid-level functions extract the real content of a social post. They determine the location, traffic related event, and the status. High-level functions implement sentiment analysis. These functions determine the entities, theme, and expected affects of these events to the traffic.

2.4.6. Support Vector Machine (SVM)

Support Vector Machine method is based on structural risk minimization principle [8] from computational learning theory. The main idea of structural risk minimization is to find a hypothesis "h" for which we guarantee the lowest true error [9].

As a learning machine, SVM classifies an input (X) using the decision function: f(X) = < X, W >+ b. SVMs are hyper plane classifier. They determine which side of the hyper plane X lies. In the formula, the hyper plane is perpendicular to W and at a distance b / ||W|| from the origin. SVM maximizes the margin around the separating hyper plane. The decision function is fully specified by a subset of trining samples. This subset of vectors is called support vectors [10].

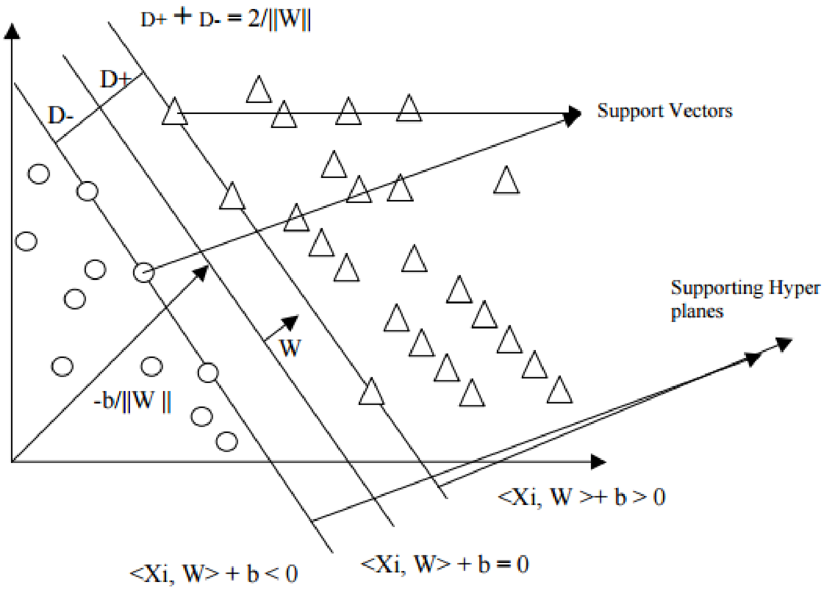

Figure 6 Support Vectors Lying on Supporting Hyper Planes

Pseudo Code for SVM Learning [11]

1. Check out all the command line arguments.
2. Load the Python Module
3. Parse_Parameters: Sets the attributes of sparm based on command line arguments.
4. Read_Examples: Reads and returns x, y example pairs from a file.
5. Initialize_model: Initializes the learning model
6. Construct cache of Ψ(x, y) vectors used in training.
7. Train model and iterate over all training examples until no constraints are added.
8. Return a feature vector describing the input pattern x and correct label y.
    - If Margin scaling, find the most violated constraint and then classify example. Return y' associated with x's most violated constraint.
    - If Slack scaling, find the most violated constraint slack and then classify example. Return y' associated with x's most violated constraint.
    - Return a feature vector describing pattern x and most violated label y'.

- Return the loss of y' relative to the true labelling y.
- If the new constraint significantly violates the existing Quadratic Programming, add it to the SVM QP.
- Print_Iteration_Stats
- Print the statistics once learning has finished.

9. Train model, and iterate over all training samples until no constraints are added.
10. Print_Learning_Stats: Print statistics once learning has finished.
11. Write_Model: Dump the struct model to a file.
12. Exit

Pseudo Code for SVM Mining [11]

1. Check out all the command line arguments.
2. Load the Python Module
3. Parse_Parameters_Classify: Process the custom command line arguments
4. Read_Model: Load the struct model from a file
5. Read_Examples: Reads and returns x, y example pairs from a file.
    - Classify_example
        - Given a pattern x, return the predicted label
    - Write_label
        - Write a predicted label to an open file.
        - Return the loss of y' relative to the true labelling y
    - Eval_Prediction
        - Accumulate statistics about a single training example.
6. Iterate over all examples
7. Print_testing Stats
8. Print statistics once classification has finished.
9. Exit

The expected accuracy value of the SVM method is over than 80%.

2.5. Map Sensor

Map sensor collects traffic – related data from public traffic density map applications such as Google Map, Yandex, İBB Traffic. In order to collect data, two different methods are used:

- Implementing API of the applications
- Processing traffic maps through image processing techniques

2.5.1. API Implementation

Yandex and Google Maps provide an API to display the maps in non-commercial applications.

The map image, provided by Yandex API {13], consists of one or more layers such as "Roadmap", "Satellite", "Hybrid" stacked on top of each other. Layers can be combined. By giving specific parameters, it is possible to show traffic information on the map images.

Google Maps [14] provides functions to create engaging web and mobile applications using a powerful mapping platform including driving directions, street view imagery and more. Google Directions API provides driving directions in 199 countries, and allows helping users find ways to specific locations. It also gives the best route based on the current traffic, and predictive travel time based on historical time of day and day of week data. Major streets can be color coded to reflect the current volume of traffic in real time.

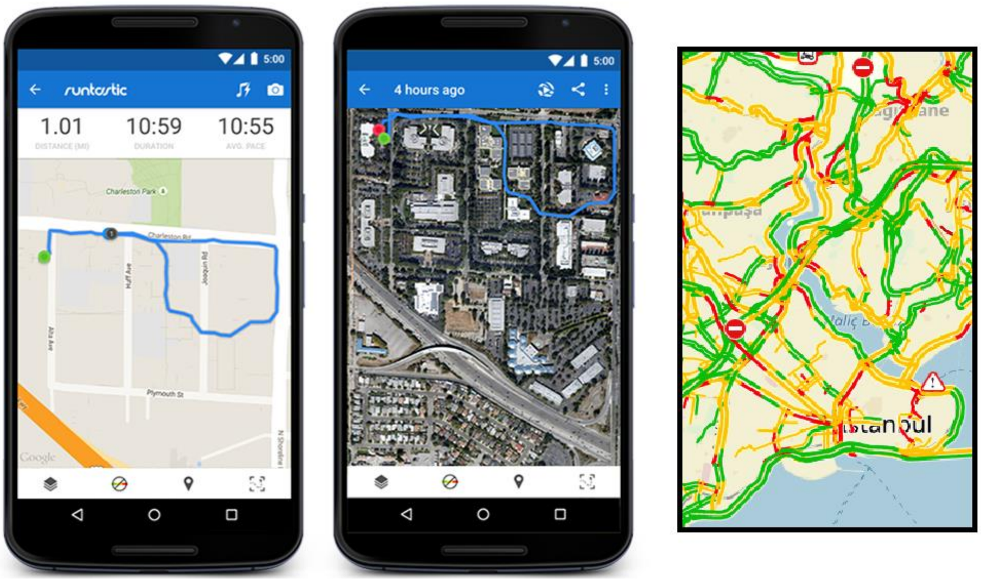

Figure 7 Google Maps API and Yandex API

2.5.2. Colect Data Using Image Processing Techniques

This method is developed for testing purposes only. In this method, a server application takes screenshots of a public traffic density map application which is opened by a browser. Screenshot images are collected in a directory. Another application takes these screenshot images one by one, and processes them. Processing consists of five different steps.

- Converting image into map (digitalization) and generate e translation formula
- Detecting road objects in the image
- Detecting density colours in the image
- Relate road objects and detected density colours
- Convert image coordinates into world coordinates using translation formula

Although there is no any other method which carries out this procedure in the literature, it is possible to liken this process to the process which extracts road objects from satellite imagery using image processing algorithms.

2.5.2.1. Road Extraction

Although identifying road objects in satellite images manually is an easy task, extracting them automatically is quite difficult task. Road extraction plays a vital role in many areas such as like military, and map implementation etc. Extraction is usually based on spectral signature. As a result of having the same spectral signature of roads and soils, spectral information alone is insufficient. In order to separate road objects from other objects, various pre-process functions should be implemented such as morphological operations.

Most of the previous works in this area are based on statistical and rule – based techniques [15]. This methods are depended primarily upon spectral information. On the other hand, some objects share the same spectral information, therefore it is not possible to separate these objects using their spectral information. Some other methods implement segmentation techniques using mathematical morphology [16].

2.5.2.2. Characteristics of Road Objects in Satellite Imagery

Locally, road objects are modeled as elongated regions with a locally constant spectral signature in the multispectral imagery and a maximum width. Globally, road objects are modeled in terms of their function for humans. The goal of the road design is to provide optimal transport of goods and people from one point to another. As a result of this, they usually from a topological network.

Some of the road extraction systems are based on fuzzy logic which is proposed by Amini in 2002 [18]. Another method segments color images using the Dempster-Shafer theory to extract linear features [19]. In 2003, P´eteri and Ranchin employ a multiresolution snake for the extraction of urban road networks given existing but imprecise GIS data [20]. Doucette et al. presented a semi-automatic approach in 2001. This approach uses a pre-classified imagery to detect roads using self-organising road map (SORM) [21]. Zhang and Couloigner provided a multi – resolution analysis approach, road junction detection method, and grouping in 2004. Multi – resolution analysis in this method is based on wavelets. Mohammadzadeh et al. introduced an approach based on fuzzy logic and mathematical morphology in 2004 [22].

In order to extract road objects from satellite imagery, local and global characteristics of the road objects can be used [17].

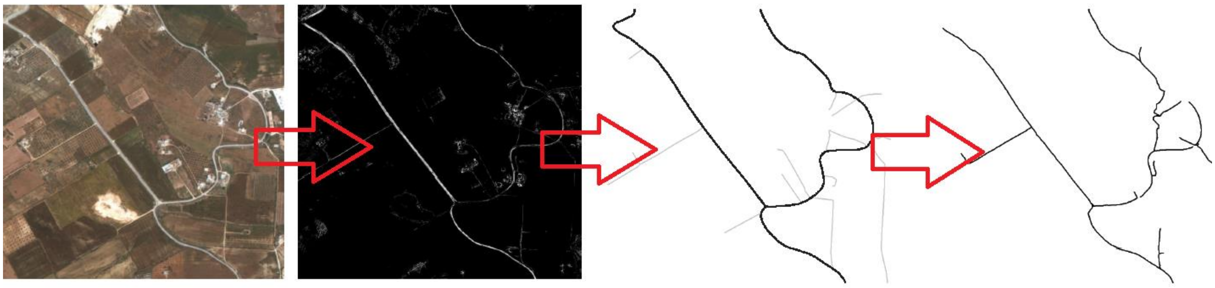

Figure 8 Road Extraction in Satellite Imagery [17]

3. References

[1] Citywide Traffic Congestion Estimation with Social Media, S. Wang, L. He, L. Stenneth, P. S. Yu, Z. Li, 2015.

[2] Mining Complaints for Traffic-Jam Estimation: A Social Sensor Application, T. Georgiou, A. E. Abbadi, X. Yan, and J. George, 2015.

[3] Advances in Knowledge Discovery and Data Mining, Fayyad, Piatetsky-Shapiro, Smyth, and Uthurusamy, (Chapter 1), AAAI/MIT Press 1996.

[4] http://nlp.stanford.edu/software/CRF-NER.html

[5] http://www.expertsystem.com/text-mining-vs-data-mining-differences/

[6] https://www.lexalytics.com/lexablog/2012/machine-learning-vs-natural-language-processing-part-1

[7] Text Mining, I. H. Witten, Computer Science, University of Waikato, Hamilton, New Zealand.

[8] The Nature of Statistical Learning Theory, Vladimir N. Vapnik, Springer, New York, 1995.

[9] Text Categorization with Support Vector Machines: Learning with Many Relevant Features, T. Joachims, Universit at Dortmund, Germany.

[10] Text Mining with Support Vector Machines and Non-Negative Matrix Factorization Algorithms, N. Guduru, University of Rhode Island, 2006.

[11] Text Mining and Classification of Product Reviews Using Structured Support Vector Machine, J. B. Chrystal, S. Joseph, IIITM-K, Trivandrum, Kerala, 2015.

[12] Real Time Traffic Density Count using Image Processing, N.Abbas, M. Tayyab, M.T. Qadri, International Journal of Computer Applications (0975 – 8887) Volume 83 – No 9, December 2013.

[13] https://tech.yandex.com.tr/maps/doc/staticapi/1.x/dg/concepts/traffic-docpage/

[14] https://enterprise.google.com/intl/en/maps/products/mapsapi.html# ,

[15] Road Extraction from Aerial and Satellite Images by Dynamic Programming, A. Gruen, H. Li, ISPRS Journal of Photogrammetry and Remote Sensing, pp. 11-21, Vol. 50, No. 4, 1995.

[16] Automatic Road Extraction from Satellite Image, B.Sowmya, A. Hameed.

[17] Automatic Road Extraction from Multispectral High Resolution Satellite Images, U. Bacher, H. Mayer, Institute for Photogrammetry and Cartography, Bundeswehr University Munich, CMRT05 - IAPRS, Vol XXXVI, Part 3/W24 Vienna, Austria, 2005.

[18] Road Identification from Satellite Images based on a Fuzzy Reasoning Model, J. Amini, Department of Surveying Engineering, University of Tehran, Tehran, Iran, 2002.

[19] Color Image Segmentation Using the Dempster-Shafer Theory of Evidence for the Fusion of Texture, J. Mena, J. Malpica, In: International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. 34 3/W8, pp. 139–144, 2003.

[20] Multiresolution Snakes for Urban Road Extraction from Ikonos and Qickbird, R. P´eteri, T. Ranchin, In: EARSeL Symposium, T., 2003.

[21] Self-Organized Clustering for Road Extraction in Classified Imagery, P. Doucette, P. Agouris, A. Stefanidis. and M. Musavi, ISPRS Journal of Photogrammetry and Remote Sensing 55, pp. 347–358, 2001.

[22] A Wavelet Approach to Road Extraction from High Spatial Resolution Remotely-Sensed Imagery, Q. Zhang, I. Couloigner. Geomatica 58(1), pp. 33–39, 2004.

[23] Automatic Linear Feature Extraction of Iranian Roads from High Resolution Multi-Spectral Satellite Imagery, A. Mohammadzadeh, A. Tavakoli, M. Zoej, The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. (35) B3, pp. 764–767, 2004.