Time for change

The combination of cheaper, better computers and enhanced wireless systems in cities means that more sensors can be deployed to manage increasingly problematic road traffic conditions. Research at the UK's University of Cambridge could point the way

Words | **David Evans, Jean Bacon, Alastair R. Beresford, Richard Gibbens & David Ingram, Computer Laboratory, University of Cambridge, UK**

The volume of road traffic in the UK, as in most other countries, has increased enormously over the past 50 years. During this time, computers and wireless communication technology have also changed for the better. The capacity and behavior of the traveler unfortunately has not changed as much over the same period, and altering the travel patterns of the typical human remains difficult.

Traveling is often an automatic and habitual process, and the collection, analysis, and distribution of traffic information from sensors to travelers is one important mechanism for encouraging the use of alternative travel times or modes of transportation, increasing the effective capacity of the road network, and reducing travel times during peak periods. In addition, the same data can be used for strategic planning over longer timescales.

To maximize the benefit derived from increased data gathering and processing, it is necessary to build an open platform to allow many companies and individuals to collaborate and share raw and processed data. Such a platform should create a marketplace in which companies can securely share and sell gathered data in order to encourage investment in sensors, networking, and processing facilities. It should also enable local residents and visitors to collaborate, contributing data and building their own applications. Such techniques have become known as crowdsourcing; examples include the creation of Wikipedia as one of the largest and most respected online encyclopedias, and the construction of freely available maps by the OpenStreetMap project.

The aims of the University of Cambridge's research in the Transport Information Monitoring Environment (TIME) project are to investigate, design, and build suitable sensor and network technology, and design and build reusable software components to distribute, process, and store sensor data in real time – and do both of these things with due regard to personal privacy and commercial interests.

Data collection methods

Useful transport data includes measurements of the movement of vehicles in the road network. This is a well-researched area. The traditional approach is to use sensors such as inductive loops in the road itself to detect the presence of vehicles at fixed points on the network and measure their speed. Other systems make use of automatic license plate recognition cameras to record the license plates of vehicles at fixed locations and use the history of sightings of license plates to estimate point-to-point journey times. The cost of installation and maintenance of these systems limits their use to inter-urban highways and some major urban routes, providing sparse coverage.

The University of Cambridge is interested in what can be done with cheap sensors, inexpensive computing systems, and wireless technology, especially when such systems are installed in places that are easy to maintain and upgrade; existing street furniture, mobile phones and satellite navigation units are a few good examples.

Sensing traffic phenomena

The existing traffic sensor deployment in Cambridge, UK, consists of 112 inductive loop sensors that generate flow and occupancy readings and propagate them to a central server along wired links. This highly intrusive sensor technology is expensive to maintain and, in this instance, is coupled with a closed architecture using proprietary protocols aimed at a single application. This is a poor fit with the University of Cambridge's vision of an open platform for data sharing, and it proposes an alternative.

Sensors in cities face a hostile environment. In the UK in particular, there can be little space for installation of new infrastructure, meaning that 'boxes at the roadside' may not be feasible. Streets are not straight, so covering a substantial fraction of the road network may require a great number of sensors, so low cost is desirable. When surroundings are dense with traffic, both human and vehicular, it means that invasive installation or maintenance of the sensors is very disruptive and can be dangerous.

Mounting sensors on lamp posts is effective at addressing these problems. The University of Cambridge believes that it has the following advantages. First, ubiquity: lamp posts are commonplace throughout cities and are located on most urban roads. This leads to good coverage of the road network with few areas of interest receiving no attention. Second, density: lamp posts tend to be close together as the light they cast has a limited range. This means that with careful sensor design, areas of observation can overlap, aiding in calibration and data validation. Furthermore, communication strategies such as wireless mesh networks become more viable.

Third, they are stationary, which eases the design of data processing and communication algorithms. Finally, street lights need power, meaning that it is available to any sensing, computation, and communication devices installed. Although designing for low power consumption is always a good idea, the fact that power is not a severe restriction is liberating as it allows more innovative designs and the deployment of prototypes whose power use might not be as efficient as production models.

Finally, designing sensors and computing components for physical hardiness is difficult, but the solidity of lamp posts allows installation of devices inside them that might otherwise be vulnerable. Furthermore, the tops of these posts are not very accessible, decreasing opportunities for vandalism.

Prototype lamp post

A schematic of the installation is shown in Figure 3. The column is similar in design to others in the city, although it becomes narrower at a greater height than usual, resulting in an expanded chamber at the base for prototype equipment. Production versions are expected to be smaller. Overall, the production of the physical column is not exotic and is unlikely to be much more expensive than that of normal columns.

Power enters the post as usual and is used to run the light, which has not been altered. This power also operates a sensor's power supply, which converts it to a voltage suitable for sensors and computers. The computer is general-purpose but has no moving parts, making it attractive for the hostile environmental conditions inside the column. Nevertheless, even the prototype is inexpensive, costing in the region of £40 (US$65).

For expediency, the prototype uses a consumer wireless router running OpenWrt. This means that WiFi networking is available, as are serial ports. There is an abundance of wired networking that at present has only been used for debugging, but is attractive for attaching network-aware sensors, a cellular modem, or hardware suitable for wireless mesh networking. For ease and low cost, WiFi has been used for communication, an antenna having been attached to the top of the lamp post (it is the white cylinder in Figure 4 and to the right of the sensor and is shown in green in Figure 3). Communication is by line-of-sight to the University of Cambridge's laboratory over a distance of approximately 100m. The system is, of course, not limited to this communications method and GPRS/HSPA has also been tried with success. Power, the sensor's serial port, and the WiFi antenna are located at the top of the lamp post and are connected to the computer using a cable.

The sensor is shown in Figure 4. It is an IRISYS thermal detector provided by InfraRed Integrated Systems Ltd. These units are relatively cheap and are self-contained. As it must by necessity be exposed to the elements, it is ruggedized for outdoor use. It measures 13 x 17 x 10cm and is light enough to be supported by a simple bracket that is part of the lamp column. The unit determines traffic volume and speed by running image-processing algorithms on data from a low-resolution (16 x 16 pixel) thermal sensor; a sample is shown in Figure 5. Using a thermal sensor simplifies image processing by removing background image clutter, as vehicles and bicycles appear as bright objects on an otherwise uniform grey background, and helps to preserve privacy as license plates and people's faces cannot be discerned. Lines, shown in green and blue in the figure, are placed on the image to define the 'count lines' for each lane of traffic. The appropriate count is incremented when a 'blob' corresponding to a vehicle crosses one of these lines in the correct direction.

The prototype was put into operation in early November 2008 and has operated continuously except for an interruption from work on the lamp post's power source. So far the prototype has functioned in rain, heat, cold, snow, and wind.

Figure 6 summarizes, for a single day (Monday, June 15, 2009), the vehicle counts and speeds as measured by the sensor on the prototype lamp post. The two plots on the left reflect the inbound direction (that is, traffic approaching the city center). The upper plot shows traffic flow in vehicles per hour (each point assesses the flow over a five-minute interval) versus the time of day. The lower plot shows the classic speed/flow diagram with speed on the vertical axis and flow on the horizontal axis. The two plots on the right-hand side show the same things for traffic traveling away from the city center. The color of each point corresponds to the time of day when the sample was taken.

This collection of four plots makes it easier to understand the nature of the traffic on a single day. (There are also structural patterns in the traffic across days, such as the effect of weekends and seasonal variations.) The inbound traffic flow shows a steep increase toward 08:30, when it can exceed 1,000 vehicles per hour. The flow then reduces through the morning to a level of around 500 vehicles per hour. There is a smaller peak in the late afternoon around 17:30, although the rate does not exceed 1,000 vehicles per hour. The lower speed/flow plot shows that speeds are reasonably steady as flow increases

Figure 1: The prototype lamp post, standing beside the one it replaced. Appearance and design are very similar

until it exceeds about 1,000 vehicles per hour. At such rates there is a considerable drop in speeds to around a quarter of their previous values.

The outbound traffic flow shows peaks at around 08:30 and 12:30, and a more substantial peak around 17:30, corresponding to morning, lunchtime, and evening. The peak flow is lower than for the inbound traffic. The lower speed/flow plot shows an interesting form of behavior. Speeds are almost constant throughout the evening peak period, but show some reduction during the middle part of the day, despite flows being at fairly low levels.

Such plots are a useful starting point and help to indicate the underlying nature of the patterns of traffic that determine journey times as well as the utilization of road resources.

Importance of middleware

Handling the flow of data between sensor, processing, storage, and output nodes in a large-scale sensor system is a complex task. It is important that connections can be reconfigured without modifying applications. As well as unanticipated requirements and new uses for the sensor data, there is steady churn as sensors and other components fail, are replaced, moved, added, or replicated. Furthermore, continuous, low-maintenance operation is needed. Taking the system down to reconfigure it is not an option; sensors produce data continually and where possible data feeds from sensors should survive any disruption from changes.

If the system relies on extensive intervention by human operators, it will be too expensive to keep running for long. Inexpensive sensors run by a multitude of organizations and individuals make central control of monitoring and maintenance impossible. For these reasons, actions such as restarting connections need to be mostly automatic and the middleware must be decentralized.

To achieve these goals, the University of Cambridge has developed a middleware layer called PIRATES (Peer-to-peer Implementation of Reconfigurable Architecture for Typed Event Streams). By delegating responsibility for interconnection to PIRATES, the system uniformly gains the reliability and flexibility described above.

The basic PIRATES unit is the component, and each component has a number of endpoints. Endpoints on different components are connected or 'mapped' and all communication between components takes place via mapped endpoints. The basic mechanism is point-to-point; components send messages to peers directly without requiring an intermediate broker, making the architecture decentralized. Components may be sensors, output devices (such as phones or web browsers), storage (databases of historical sensor data), filters (data cleaning, format conversion, anonymization, etc) or data fusion operators that combine multiple streams.

A mapping between a pair of components may be set up by the component at either end of the

Figure 2: Inside the base of the lamp post, showing the power distribution box and the wireless router that provides computation and communication facilities

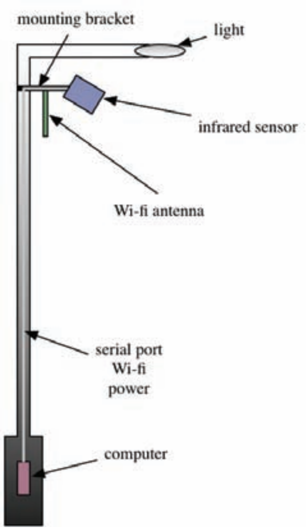

Figure 3: A schematic showing the components inside the prototype lamp post and how they are connected

Figure 4: (previous page) The sensor on the lamp post, along with the WiFi antenna and apparatus allowing air collection for measuring pollution

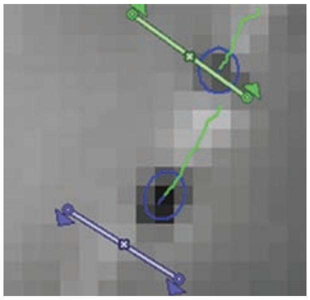

Figure 5: (above) The infrared sensor's view of traffic, from which it derives vehicle counts and average speeds

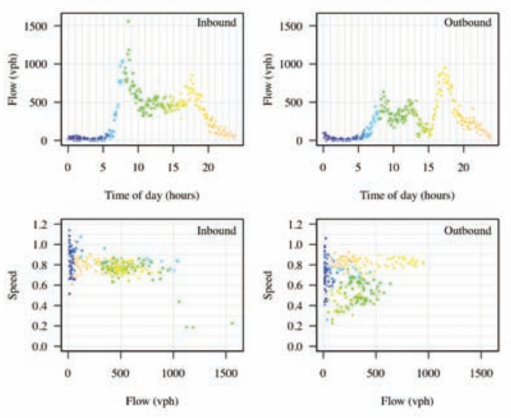

Figure 6: (above right) Measured speeds and flows on a single day at five-minute intervals. Inbound traffic is shown in the two left-hand plots and outbound traffic is shown in the two right-hand plots. In each plot, the hour of the day is shown by a different color ranging from blue at the start of the day through to orange at the end of the day. Each of the lower plots is a classic speed/flow diagram with speeds on the vertical axis and flows on the horizontal axis

connection or by a third-party component. The latter option is the primary means of establishing mappings and is responsible for PIRATES' flexible run-time reconfiguration abilities. Components that create their own mappings correspond to traditional network programming idioms but are undesirable because the description of the component to connect to is typically hard-coded or entered as a parameter by the user.

With the University of Cambridge's paradigm, components create the necessary endpoints, but initially start with them unmapped and inert. Mapping may then be performed by an administrator (with a graphical network visualizer, for example) or by mapping engines. These operate on rule bases stored in configuration files, for example 'any endpoint of description A should always be connected to at least one endpoint of description B'.

If a component needs to be upgraded or migrated (perhaps to deal with malfunctioning hardware), this can be done without interruption to service by remapping its endpoints. If a connection to a required peer component is lost, the mapping engine can re-establish a connection to a suitable alternative automatically. The isolation, which is enforced, of the business logic of components from the communications mechanisms managed by PIRATES means that mappings can be diverted or re-established in the event of failure without business logic being aware that anything has occurred.

Content-based filtering can be applied to select which messages are sent, much as subscription expressions are used in publish-subscribe systems. Because the use of intermediaries such as event brokers is not enforced, messages are filtered but still sent point-to-point for minimum latency and in many cases reduced bandwidth. If message aggregation is required, there is an optional event broker component that accepts any message and forwards it to any listener for which the subscription is satisfied.

PIRATES uses a schema language called LITMUS (Language of Interface Types for Messages in Underlying Streams) to describe message formats. Every endpoint has a schema associated with it to describe the types of message it handles. The middleware ensures that only type-compatible endpoints can be mapped. All messages are transmitted and stored together with type identifiers called LITMUS codes. This makes messages self-identifying, providing a fast probabilistic type check that is stronger than a unique type ID and does not require a central type authority.

Each PIRATES component provides an identification service that returns the names and types of its supported endpoints and the identity of all peers that are currently mapped to the component. Locating other components can be achieved in the same way that a search engine indexes and searches the internet. A location service can determine the network topology by crawling the graph of connected components. There are also local resource discovery components that can be registered with automatically.

Conclusion

The University of Cambridge has barely touched the surface of what is possible with small, plentiful sensors. In the future, it hopes to expand the range of data contributed by individuals, beginning by exploring what can be done with GPS data collected from satellite navigation systems and smartphones. It should be possible to use this data to improve travel-time estimates for all road users. Such a system would have the benefit of no roadside infrastructure, which means no maintenance issues or power problems, and no expensive network communications.

This research is supported by EPSRC grant EP/C547632 and by British Telecommunications plc

Bibliography

1) S. Kenyon and G. Lyons, "The value of integrated multimodal traveller information and its potential contribution to modal change", Transportation Research Part F: Traffic Psychology and Behaviour, vol. 6, pp. 1-21, Mar. 2003.
2) D. C. Brabham, "Crowdsourcing as a model for problem solving: An introduction and cases," Convergence, vol. 14, pp. 75-90, Feb. 2008.
3) M. Haklay and P. Weber, "OpenStreetMap: User-Generated street maps," Pervasive Computing, IEEE, vol. 7, no. 4, pp. 12-18, 2008.
4) P. B. Hunt, D. I. Robertson, R. D. Bretherton, and R. I. Winton, "SCOOT – a traffic responsive method of coordinating signals," Tech. Rep. LR1014, Transport and Road Research Laboratory, 1981.
5) R. Gibbens and Y. Saatci, "Data, modelling and inference in road traffic networks," Phil. Trans. R. Soc. B, vol. 366, pp. 1907-1919, June 2008.
6) "OpenWrt." http://www.openwrt.org/.
7) A. May, Traffic flow fundamentals. Prentice Hall, Englewood Cliffs, NJ, 1990.
8) D. Ingram, "Reconfigurable middleware for high availability sensor systems," in Proceedings of the Third International Conference on Distributed Event-Based Systems, ACM Press, 2009.