Visualising the Past: Annotating a Life with Linked Open Data

Ashley Smith
University of Southampton
Southampton
United Kingdom
ads04r@ecs.soton.ac.uk

Kieron O'Hara
University of Southampton
Southampton
United Kingdom
kmo@ecs.soton.ac.uk

Paul Lewis
University of Southampton
Southampton
United Kingdom
phl@ecs.soton.ac.uk

ABSTRACT

This paper introduces a novel lifelogging system. The novelty of our system is that it uses open linked data to reduce the burden on the user to an absolute minimum by combining the user's private logs with data obtained automatically from the web. This in turn is annotated with data generated by the latest version of the author's portable 'life annotation' software to generate a detailed, interactive, user-friendly visual representation of the user's life experiences with little or no manual work required. This paper covers several key parts of the system that make it work. Specifically, the analysis of linked data, entity comparison with known objects and events from the user's own logs in order to determine relevance, and careful annotation of the user's data ensuring that the data's provenance is stored alongside the annotations generated.

1. INTRODUCTION

Lifelogging is, in essence, the collection of data in order to illustrate a person's life. This is useful for many purposes, from simple personal interest to helping assist the memories of dementia sufferers [4].

The basic concept of lifelogging dates back to 1945, when Vannevar Bush first envisioned what he called a Memex [6]. Bush's Memex was to be "a device in which an individual stores all his books, records and communications", and one capable of creating links between documents both local and remote. It may arguably be likened to the internet-connected personal computers of today, as most of the storage, indexing, annotating and linking capabilities are implemented by them. However, the Memex was to have functionality that the average PC does not generally have, and it is this functionality that lifelogging projects seek to realise, by collecting data the ordinary PC user may not need or be able to collect. 'Memex' is a portmanteau of 'memory index', and is described as "an enlarged intimate supplement to [the user's] memory", rather than simply a well organised document storage system, and work exists to store the elements of a user's life that a normal PC is not capable of recording.

Currently, several lifelogging systems exist. The most famous is Microsoft Research's MyLifeBits [11, 3], which is an ongoing study into digitally storing the life of test subject Gordon Bell. The field is vast, including work on wearable cameras and computers [14] and location-sensing technology. Some work [11, 9] goes so far as to calculate the amount of digital storage space necessary to record an entire life, and a recent surge in the amount of available portable technology has enabled this missing piece of Memex to become a popular research area.

The concept is not limited to research and academia. One of the earliest uses of the phrase "LifeLog" was the DARPA project 'LifeLog' [1]. Its mission was to "trace the 'threads' of an individual's life in terms of events, states and relationships" by aggregating raw data into a timeline, or "episodic memory". LifeLog was cancelled after an outcry from civil liberties groups, as the project was deemed to have been collecting too much personal information, such as private phone calls, and because of the involvement of a government organisation. As lifelogging systems become more advanced and begin logging interactions between people rather than just the one user, privacy is a genuine concern and one to which we will return in section 6.

The next logical step for lifelogging systems is to actually make sense of, and organise the user's lifelogs.

1.1 Nomenclature

Throughout this paper, we use the phrases 'lifelogging' and 'life annotation'. They are not to be considered synonymous. Life annotation is a phrase coined in previous work [20] and differs slightly to lifelogging. We define lifelogs as data or content, such as photographs and sound recordings, that represent life experiences whereas life annotations are sparse fragments of metadata, such as location information obtained from a GPS, which may describe the lifelogs, the experiences to which they refer, or both. The two concepts work hand-in-hand, as life annotations tend to provide an index to lifelogs, but for the purposes of this paper it is necessary to disambiguate the two.

WebSci '11, June 14-17, 2011, Koblenz, Germany.

2. PREVIOUS WORK

Previously we introduced the Imouto[1] life annotation system [20]. Originally designed for photo annotation, the system eventually evolved to a portable life annotation tool designed to run on a mobile device [19]. The system passively collected GPS logs, the user's calendar and logs of nearby Bluetooth devices, and a companion system running on a PC combined this with additional information for the purpose of study. Since then the system has evolved beyond a simple data collection system into a true lifelogging system. A description of the current system will be presented in section 4, but first we present a description of the problems that the system attempts to solve, previous work in the field, and justifications for our design and implementation decisions. The data for our studies comes from a willing test subject who has been using our Imouto life annotation system since 2006, and the results in this paper are based on the data generated by this test subject for the three years 2008 to 2010 inclusive.

3. MAKING SENSE OF THE DATA

It has been suggested [22, 19] that life annotations may be used to generate narrative and it is from this idea that we take much of our inspiration for this system. Our previous work shows that we can gain more understanding from life annotations if they are varied and from different sources, and in section 5 we will investigate the combination of personal data with open linked data available on the web. But even before we consider the benefits of open linked data in the field of lifelogging, there is already much we can gain through opportunistic analysis of the collected life annotations. We can also exploit collected information for purposes not originally intended.

3.1 Person Detection

Person detection in our system was initially limited to information obtained from Bluetooth scans, but recent revisions allow information to be obtained from image processing technologies applied to the user's photographs. Both person detection methods employed by the system are semi-automatic processes, as they involve the user telling the system which devices are owned by known people and training the face recognition system.

Each Bluetooth device has its own globally unique identifier, or MAC address. When a device performs a local area scan, known as 'discovery', the MAC address and 'friendly name' of all nearby devices are returned. While the friendly name is customisable by the user, the MAC address cannot be changed through conventional methods. This gives us a way of uniquely identifying devices. Our system allows us to manually associate a device with a person. As owners of mobile devices such as PDAs and smartphones are unlikely to routinely switch devices with other people, this is a reasonably robust method of determining which people are present.

Person detection may also be achieved through face recognition. The popular Picasa photo management software [2] has this functionality built in, and exposes its annotations in plain text format which is simple to parse. Its semi-automatic interface is also quite simple, requiring the user to enter the name of each person manually in the first instance, and then simply asking a yes or no question from then on. We can then assume for the purposes of life annotation that if the user took the photo, they were very likely to be in the presence of the person in the photo.

We carried out a study to determine whether face recognition or Bluetooth recognition is a more efficient method of person detection within our system. Using the life annotations collected by a user of our system for a period of three years, we split the period into events and counted how many people were correctly identified as being with the user during each event. An explanation of events follows in section 3.3. Once we had found all events with people present, we calculated what percentage of these people were detected using Bluetooth scans and what percentage through image recognition. Over the course of the three year test sample, 89.5% of people detected came from the Bluetooth analysis and 14.3% came from the face recognition, suggesting that detecting people via Bluetooth is vastly more effective. However, when we filtered the results and analysed only the events during which photographs were taken, the face recognition produced a considerably higher percentage of matches than the Bluetooth analysis. The reason for this is simple – the user has a complete Bluetooth log of the three years, but was not taking photographs for this entire period. But as the two datasets of detected people compliment each other, it was decided to use the two in parallel rather than pick one over the other. The main conclusion that can be drawn from this study is that wearable cameras such as the Microsoft Sensecam are potentially beneficial to our lifelogging system, as they increase the number of photographs taken and do not rely on the user actively photographing someone.

3.2 Location Detection

With most mobile devices featuring GPS and A-GPS location sensing technologies, it seems sensible for any mobile life annotation system to store this data and our system has been doing so since its very first version. But GPS is not flawless, and there are other ways of sensing location using non specialist hardware.

As previously mentioned, our mobile system collects Bluetooth data, intended for person recognition. It is designed to make regular Bluetooth scans in order to determine the unique MAC address of all devices within range of the user. Assuming that most other Bluetooth devices are mobile devices owned and carried by other people, in certain conditions one can infer which of the user's colleagues are nearby at any given time, as well as which are most familiar to the user [13]. But Bluetooth can also be used for location sensing.

In a computer science lab, such as our lab at the University of Southampton, it is possible to determine roughly whereabouts in the lab a person is located by using the Bluetooth MAC addresses of static devices such as desktop PCs, as-

[1]Imouto is a japanese word meaning 'little sister', a humourous reference to the fact that the system contains certain properties of Orwell's Big Brother, yet is its opposite in many ways.

[2]http://picasa.google.com/

suming training data is available. While not perfect, indoors this works much better than GPS due to the difficulty of obtaining a GPS signal from within a building. We can theoretically expand this to the outside world with larger amounts of training data, but this will be further discussed in section 8. Limitations of using Bluetooth for location sensing are similar to the problems encountered by Lavelle et al [13]; Bluetooth signals can penetrate walls and operate in a three-dimensional plane meaning that the system may wrongly detect the user to be in the room next door, or even above or below his actual position. There is work yet to be done, but theoretically this problem may be limited with a higher density of static Bluetooth devices for which the location is known.

3.3 Event Detection

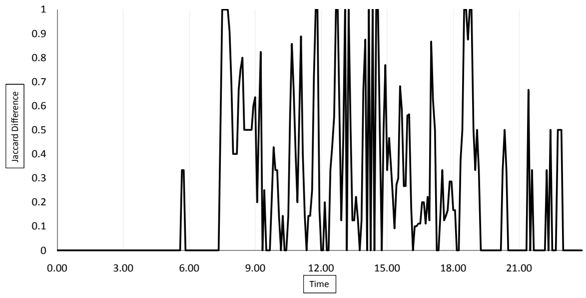

Figure 1: Bluetooth cluster difference over time on 5th November 2010.

It became obvious near the beginning of development that any visualisation of lifelog data needs to be split into smaller sections in order to avoid information overload. Initial tests splitting the data by hour or by day began well but once real data was used it became apparent that this was not acceptable. The key organising principle here is not an uncontextualised timeslice, but rather an event, defined by Jain [12] as a significant occurrence, happening, gathering or activity. Often an 'event' in the user's day will last for much longer or shorter than a previous event – for example, a 20 minute drive to a 2 hour meeting, followed by another 20 minute drive home should be three nonequal events – and even if the data is split by day, events may cross the day boundary making the generation of a coherant narrative difficult. It is for this reason that a better method for splitting life data into 'events' is needed.

There is already work in event detection for lifelogs, although the vast majority is based around visual features and relies heavily on some kind of wearable camera. Nonetheless, many novel methods of event detection and classification already exist. Doherty [10] describes a method of event detection using data from additional external sensors such as audio, and the work of Byrne et al [7] uses low level image features combined with light levels, but then categorises these events using GPS and Bluetooth data captured at the same time. Inspired by this, we began investigating other methods of event detection.

Our first attempt to split a life into events using our data was to use location sensing to segregate events, which works, but is flawed as multiple consecutive events in the same location are considered to be one. An example is a typical day of the test user, which involves a period at home, a journey to work, a day at work lasting around eight hours, a journey home and then a final period at home. Using location alone this entire day would be split into only five events and one would be eight hours long despite the fact that it may contain many sub-events such as multiple meetings on different subjects, a lunch break, coffee breaks and so on. Clearly basing our event detection on location alone is not sufficient. We then trialled event detection based on people. This is fine when the user is known to be with others, but over the three year trial period a person was only detected within range of the subject for 28.6% of the time, so most of the user's life would contain no discernible events. However, despite the apparent lack of known people, there are very few periods of time where no Bluetooth devices are detected at all. It is for this reason that we began experimenting solely with raw Bluetooth scan data for event detection.

Our Bluetooth scans are limited by hardware and device battery life. As Bluetooth scans are active rather than passive, it is not feasible without some custom modified hardware to leave a Bluetooth device in 'discover' mode and log each individual device, as is possible when scanning for wifi devices. Instead, the Imouto software instructs the Bluetooth hardware to carry out a scan at predetermined intervals, and our data set was gathered with the software set to scan once per minute. For this reason it is entirely possible that the user is in the proximity of a device for a long amount of time, yet the device is busy or otherwise not responding to scans at the precise second the scan is taking place. Fortunately, an 'event' lasting only a few minutes is probably not that significant, therefore we may set our cluster length to something slightly larger. We begin by clustering our device scans into five minute blocks; it was decided that five minutes is an acceptable minimum length of time for an 'event', and if a device disappears for more than this time it is either genuinely out of range or malfunctioning. Once we have our clusters of devices, we use an inverse of the Jaccard index[3] to compare the difference between them, so a value of zero between two clusters indicates no difference, and a value of one indicates two completely different clusters. The reason for inverting the value is that it helps when comparing empty sets when there are no Bluetooth devices available; comparing empty sets requires dividing by zero.

In order to determine event thresholds from the similarity peaks we simply decide how many events per day we require, and pick the highest difference values throughout the course of the day. For example, figure 1 shows a graph of difference values over time on a typical day. We can see that the highest spikes occur at 7.30am, 6.00pm and between midday and 3pm. Comparing this to other life annotations, particularly GPS and appointment data, the spikes correspond with the user arriving at work at 7.30, leaving work at 6.00 and several meetings in different rooms throughout the afteroon. There are smaller spikes throughout the day which show things such as coffee breaks, when the subject is generally with the same group of people but in a differ-

[3] Jaccard index is a common method for comparing two sets and involves dividing the size of the intersection of the two sets by the size of the union.

ent position in the building. The time between two event thresholds is considered to be an event. In the unlikely case that we have a number of equal-height peaks that total more than the maximum number of events per day, we select the peaks based on their distance from each other in order to get a reasonably diverse set.

4. THE IMOUTO LIFE ANNOTATION SYSTEM

Since its first version, the Imouto system has retained original intention, passive and effortless collection of life annotations that may be combined with additional information to produce a rich description of life events. However it has been improved and is now a full lifelogging system incorporating the original software which is now known as Imouto Mobile.

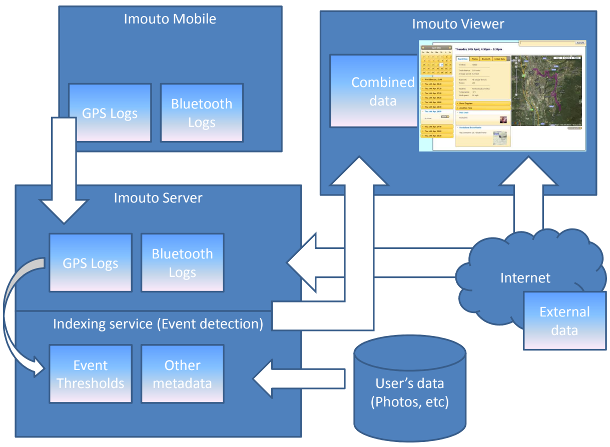

Figure 2: System architecture diagram.

Imouto Mobile is very similar to the original system, and merely collects personal life annotations. Imouto Server is an annotator service that runs on the user's home computer in order to manage the collected life annotations and produce more annotations based on the user's documents. It is this system that splits the user's life into events and interfaces with the external processes, such as importing the face recognition data from Picasa. Finally, Imouto Viewer is a web application written in PHP and JQuery. It is used to produce a user-friendly display of the user's life in a web browser, alongside other relevant information from the web.

4.1 Imouto Mobile

The current version of Imouto Mobile runs on a modern smartphone and uses very few system resources, meaning it can run in the background on a phone that is in normal everyday use. This differs from its original incarnation, which required a dedicated mobile device. The system is written for the .NET framework and runs on phones running Windows Mobile 5 or 6. In the modern Android and iOS dominated market this may seem an odd choice, but the system was originally designed for a PocketPC and much of the original code still exists. Additionally, most iOS and Android based devices do not currently allow the low level access to the hardware required by the system.

In order to keep the hardware requirements low, the system does as little as possible. A small executable runs in the background all the time and collects information on nearby Bluetooth devices, phone signal and power status, and a foreground application runs when required in order to collect location information from GPS and also to import the user's calendar and SMS inbox. The collected information is stored locally on the device in a sparse plain text format in order to save space. It was decided that the GPS should not run all the time as it is a big drain on battery, and would not work if the device was indoors anyway.

When the device is plugged into the host PC for charging and synchronisation, the Imouto Server imports the stored text files into its own database and removes them from the device to save memory.

4.2 Imouto Server

Imouto Server runs on a Windows PC, and is also written using the .NET Framework. The vast majority of the data processing is handled by this part of the system. It is primarily a desktop application but remains active in the background when not in use, so it is best run on a PC that is rarely turned off. The test user is running the application on a medium-spec low power consumption PC which is running nothing other than Imouto Server and its required MySQL database. It remains on 24 hours a day.

The server software works in two parts. Firstly, as an unattended annotation system which connects to various other services and programs in order to gather and annotate data. This part of the software is responsible for processing the raw text files stored on the mobile device and importing them into a local relational database. It also has access to the local area network where the user stores his photographs, and from here it imports the EXIF data as well as any extra metadata generated by photograph organisation software Picasa, such as descriptions and face data. The second purpose of this software is as an application framework for custom 'collector' scripts. Collector scripts are mini-programs that return data in a specific format, and the main software calls the scripts at set intervals throughout the day in order to store the data returned. The simplest of these scripts calls and parses the Yahoo Weather RSS feed in order to store a complete log of the weather in areas the user visits often, but more complex scripts are possible for storing more information. This idea of having a main application that calls external scripts was to ensure development on individual scripts could proceed without having to halt or restart the system overall. Effectively, the server program is the part of the overall system that requires the least interaction, as it continues running all the time with no user input whatsoever.

4.3 Imouto Viewer

Imouto Viewer is written in PHP and Javascript (with JQuery) and runs entirely in a web browser. All the life annotations and other data may be viewed in an interactive, user-friendly way. Upon selecting a date from the calendar, the user is presented with a number of events from that day, titled if

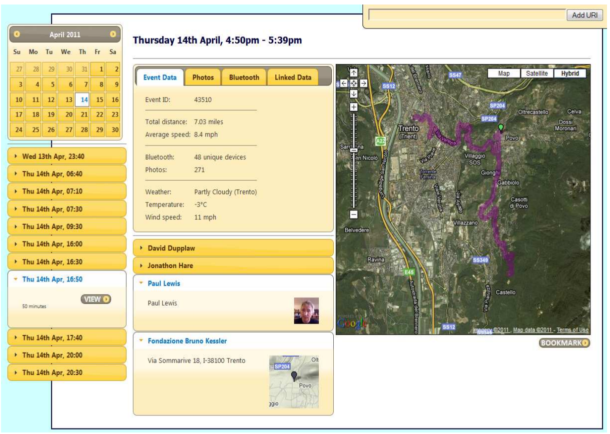

Figure 3: Imouto Viewer.

possible, and each event has an associated page containing a summary of the event. Summaries contain, if known, the places the user visited, the people with whom he associated, and other statistical information such as the weather, distance travelled, etc. The user may select alternative views which feature lower level information, such as the Bluetooth devices encountered, and the photos taken for each event. Additionally, everything is clickable so the user may, for example, click on a person and get any known information on them, as well as a list of other times they were encountered.

Information displayed by Imouto Viewer is primarily taken from the user's personal lifelogs and manually added annotations, but it is also possible to import an RDF document from the web in order to automate the annotation further, and we will discuss this in the next section.

5. LINKED DATA

When the system was first developed, it had a fundamental drawback. The user needed to manually insert data in order for the system to generate any kind of useful output. People could be pulled from the user's phone book, but places are a different story, as there is no common way of storing known locations on a phone, unless the user uses his or her phone as a sat nav, and this is not likely to be accessible via a common API. Thankfully, by the time the system had reached this point in development, open linked data had begun to appear on the web.

The Imouto Viewer application contains an input field available at all times which allows the user to import an RDF document from a web location. This document is then parsed by the system and the triples within it are imported into the system's internal database. This database is queried for relevant information by the viewer application whenever part of the user's life is visualised.

5.1 Connecting Entities

The system is partially intelligent in that it supports various ways of determining if two URIs refer to the same entity. In the case of people it does this using the `foaf:mbox` property. It is stated within the FOAF specification that two `foaf:Person` entities cannot have the same email address. Although this is not always true in real life – children may share their parents' email address, for example – we implement this element of the FOAF vocabulary by assuming that if two entities have the same email address that they are actually one and the same. This allows us to link `foaf:Person` entities from RDF documents to entries in the user's phonebook which may not necessarily have a URI assigned to them.

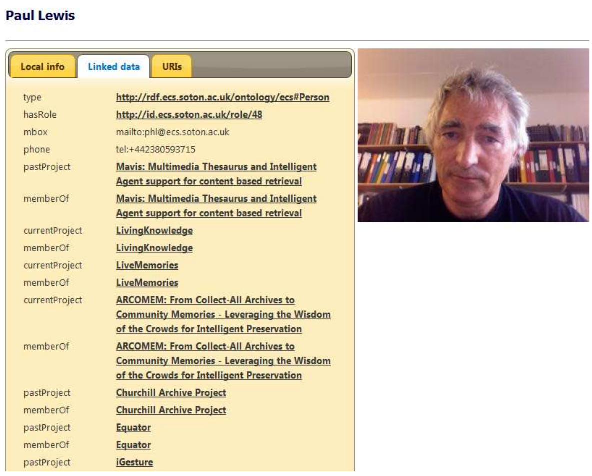

Figure 4: Imouto Viewer after importing linked data about a person.

Figure 4 shows the result of importing an RDF description of a person. The user has the email address of Paul Lewis in his address book but no other information. After importing his RDF description from `http://rdf.ecs.soton.ac.uk/person/48`, Imouto Viewer now shows Paul's full name and title, desk phone number, home page and projects as well as a photograph.

For entities which are not people, we use the `owl:sameAs` property. If a triple exists that states a URI is the same as another URI, it is assumed that this is true and the two entities are effectively combined to form one, although for reasons of provenance, to be detailed in section 5.3, this combination only exists within the viewer and not in the internal database.

5.2 Manual Linking

Not everything can happen automatically, so some form of manual method is required within the viewer application. The interface currently contains two such methods. Firstly, two people may be combined within the person database. If the system has imported the same person from two sources which are not explicitly linked, the user may manually combine the properties of both versions to form one entity. There is also a location manager in which entities with spatial properties may be integrated into the user's manually entered locations list, either by adding them or by combining them with existing locations. As with the automatic annotation, the link is not integral to the internal data and can therefore be removed with no side-effects if the user makes a mistake.

5.3 Provenance

The problem with importing linked data from the web automatically is that it may not all be reliable. Accidentally or maliciously, information that is not helpful or true may be imported into the Imouto system and treated as factual. For this reason, every triple within the internal database references the RDF document from which it came, as well as the date the document was last imported. Consequently the viewer application is able to list all the documents that have been imported, and the user may decide to delete or re-import each document individually should they go out of date or later be deemed unreliable.

6. PRIVACY

In general, lifelogging has raised privacy concerns [3, 2, 16] based around the potential for exploitation of information which has been conveniently (for the snooper) collected in one place. In this section we begin by reviewing some of well-known privacy and surveillance concerns, before moving on to concerns about linked data in section 6.2. In section 6.3 we consider additional factors associated with life annotation as opposed to lifelogging per se.

6.1 Privacy and Lifelogging

To invade the informational privacy of a person, a snooper needs to collect information about that person (an obvious truism). A serious breach of privacy may occur if the snooper manages to find a single juicy piece of information (for example a criminal record), which may be picked up from a number of sources, including government open data, blogs, online news sites and aggregators, Tweets or social networking sites. A lifelog is unlikely to add a great deal to this risk, as these other sources are themselves already extremely deleterious to privacy.

More problematic is the case where the snooper has to piece together individually innocuous pieces of information to provide an unwonted insight into someone's life (for example, an anonymised criminal record, an anonymised record of a medical treatment undertaken within a prison, and a medical record may place a data subject in prison on a particular date). The cost of such data fusion is relatively high, but the greater part of the snooper's job may be done for him by the lifelogger collecting information about him- or herself.

This discussion is premised on the snooper's gaining access to the lifelogging data. This will of course be trivial if the lifelogger has posted it to the web, or made it available in some other way. Yet this need not be the case. Bell and Gemmell [3] explicitly advise that if you have serious privacy concerns, then don't put your lifelog on the web; Bell keeps the MyLifeBits information to himself for personal use. As noted above, Imouto focuses its security/privacy strategy on the client side, and no data are held on an external server. Hence for the unauthorised snooper, the privacy problem is equivalent to the anti-hacking/security issue.

The second risk is that one's lifelog will actually contain details not only of one's life, but of that of other people. The record of one's own life is therefore also a partial view of those others' lives whose privacy is therefore put at risk through no fault of their own. Nevertheless, the risk still depends on a number of behavioural norms being crossed. The lifelogger would have to elect to share the data concerning others' conduct, either by posting on the web or sending it to interested individuals. Sharing is not implicit in the practice of lifelogging, so it is not clear how large a risk this would be. It is certainly the case that it would entail that the lifelogger was no longer keeping the data for his or her "domestic purposes" (to use the key term from the UK's Data Protection Act 1998), and therefore the data transfer would fall under data protection principles, as mandated in a number of countries including all those in the OECD. This would mean, for instance, that data could only be used for the specific purposes for, which it was collected, and that it must not be disclosed to other parties without the expressed consent of the data subject.

This is not to say that abuses could not occur (they do under the current imperfect data protection regime), but the area is covered by law already; it is not clear what an extra law regulating lifelogging would add.

6.2 Privacy and Linked Data

As Bizer et al [5] point out, privacy is one of the major research challenges to the linked data web. Datasets are published in the Linked Data cloud without metadata expressing privacy preferences or restrictions, thereby putting everything automatically into the public domain. It is certainly the case that languages and tools for expressing preferences will be important for the linked data web, especially with respect to the social web [17].

However, using linked data for lifelogging need not be a public activity. As noted above, the lifelogger imports important and useful contextual data to help provide a rich picture of his or her own life. If the data he or she generates is not published as linked data, then (a) the individual lifelogger's privacy is protected to the extent that the data store is secure from hacking, and (b) the lifelog is only used for domestic purposes, and therefore does not fall under the regulatory regime for data protection of others' data.

6.3 Privacy and Life Annotation

If, on the other hand, the lifelogger does wish to publish his or her own datasets as linked data, then it may be that methods for generating relevant metadata could be developed by systems such as Imouto. Life annotation systems could be crafted that are positive boons for privacy by providing metadata automatically for lifelog datasets that could help indicate privacy preferences semi-automatically, or be used to infer how the data should be treated in accord with current privacy norms, perhaps leveraging location-sensitive models of privacy.

For instance, Toch et al [21] exploit the notion of location entropy to develop models of privacy in location-sharing systems. A location has a high entropy when it has many visitors who each visit with similar frequency, and low entropy when it has few visitors who visit with highly uneven frequencies. Hence a typical high entropy place would be a public area, such as a university campus or a railway station; a typical low entropy place would be a private home, whose residents are present vastly more often than anyone else. Toch et al argue that, all things being equal, people are more willing to share their location (are less concerned about privacy) when they are in higher-entropy locations.

As noted above, our system exploits Bluetooth data for both person and location identification, and hence, over a period of time, could generate a parameter analogous to location entropy for a particular location. If the empirical results of Toch et al were confirmed, then the user of Imouto would be able to treat information differently depending on the entropy of the location. Default privacy preference metadata could be inferred from the characteristics of a location gathered by Imouto itself.

7. THE SOCIAL VALUE OF LIFELOG INFORMATION

Although privacy is the main concern of lifelogging's critics, there are other concerns too. Some worry that the information gathered will be valueless. "Rather than unfocused efforts to 'capture everything', system designers should channel their efforts more fruitfully by identifying the situations where human memory is poor or targeting the things users most want to remember. These situations are where the systems would provide their greatest utility" [18]. Why bother capturing a load of stuff that humans can already remember, or alternatively aren't interested in remembering?

Imouto goes some way to alleviating the concerns expressed here, as its focus is on sensemaking and data organisation. Generating narratives from the raw data provides an intermediate stage between the data and the descriptions of most utility. The problem with Sellen and Whittaker's criticism is that, though well-made, it is extremely hard to predict exactly which data will prove valuable. Memory works associatively, and so although it may be possible to make a guess at which data will be found to be important in the future (even this is doubtful, but let us allow it for the sake of argument), it will be completely impossible to identify in advance the subordinate pieces of data that will lead us by association to the important stuff.

Lifelogging is also criticised for contributing to information overload, but the problem of associative indexing shows why such criticisms are inappropriate. "Capturing vast arrays of data might overwhelm end users maintaining and retrieving valuable information from large archives; it also ignores the burden huge amounts of data impose on system designers and developers" [18]. Viktor Mayer-Schönberger [15] has also argued that we should be more prepared to delete information because there is too much of it available for comfort, while Nicholas Carr maintains that tools like the web are bad for our health, because "the influx of competing messages that we receive whenever we go online not only overloads our working memory; it makes it much harder for our frontal lobes to concentrate our attention on any one thing. The process of memory consolidation can't even get started" [8]. However, the association problem means that the value of the lifelog is proportional to the amount of information gathered and stored.

Given that it is not possible to filter the lifelog's information to preserve all and only the useful stuff, it is clear that intermediary representations are required; these are partly created by Imouto's opportunistic analyses, to suggest people, locations and events that have probably been encountered. These are key concepts in our own understanding of our lives, and key components of life narratives.

Furthermore, Imouto's use of linked data helps raise the value of the lifelog data still further by contextualising it. The linked data stores which Imouto can tap into will also provide further metadata for life annotation, making greater (and more objective) sense of the individual experience. The exploitation of the linked data web will also help address the problem of "overwhelm[ing] end users" and "overload[ing] our working memory"; the acquisition, storage and annotation efforts for the linked data will of course be distributed.

Hence Imouto's contribution is not only to the capture, storage and representation of life data, but also to its usability, relevance and contextualisation. The linked data web will provide a growing resource for the interpretation and retrieval of relevant data for the individual.

8. CONCLUSIONS AND FUTURE WORK

In this paper, we have discussed several lifelogging systems, including the latest version of our Imouto system. Imouto both collects and organises lifelogs and life annotations, as well as combining them with both external contextual data and open linked data from the web. We have discussed some of our design and implementation decisions, particularly within the areas of person detection and event detection, and suggested ways the software can be improved. We have addressed various issues with regards to privacy and social issues and explained how they apply to Imouto.

In section 3.2 we hypothesised that Bluetooth could be used for location detection on a large scale, and the testing of this hypothesis is important future work. On the face of it, collecting training data for location detection by Bluetooth will be similar to the way Google, Skyhook and others have collected the MAC addresses of wireless access points to determine location based on local wifi. The main difference is that most Bluetooth devices are portable and we can only use static devices for location detection. This is not as problematic as it sounds. For example, we know from the registry of Bluetooth device manufacturers that any device whose hexadecimal MAC address begins with `00:03:81` is manufactured by Ingenico, a company that produces card payment devices for pubs and restaurants. It is unlikely that such devices will ever move, and when collecting the addresses of static devices we can be reasonably sure that these devices are the most likely to be static. Likewise any device beginning with `00:19:2D` is made by Finnish mobile phone manufacturer Nokia, meaning it is unlikely to be useful for location detection, and can therefore be removed from our training data.

In section 3.3 we discussed the different methods we have already tried when attempting to split a user's life into events. We have opted to use Bluetooth, but there are other ways of doing this and a study would be worthwhile. In section 3.1 we suggested that face detection is a more effective method of person detection than Bluetooth device logging, but only for events in which many photographs exist. With the existance of wearable cameras such as the Sensecam, it seems sensible to experiment with a constant stream of images in order to evaluate the effectiveness of wearing a camera.

Finally we propose more work in the field of open linked data. Our ultimate goal is for Imouto to be completely automatic, and despite the ability to annotate using public data it is still up to the user to actually locate this information. For example, by looking for common paths and destinations in the GPS logs it is possible to identify locations relevant to the user's life without any prior knowledge. Using techniques outlined in section 6.3 it is theoretically possible to identify 'public' locations relevant to the user, and with a simple search, identify these known locations by name and type, all with no user input.

9. ACKNOWLEDGEMENTS

The work reported in this paper was partly supported by the projects LiveMemories (Active Digital Memories of Collective Life), Bando Grandi Progetti 2006, Provincia Autonoma di Trento, and the EU FET project Living Knowledge, (`http://livingknowledge-project.eu/`), contract no. 231126.

10. REFERENCES

[1] LifeLog Proposer Information Pamphlet. Defense Advanced Research Projects Agency Information Processing Technology Office document PIP 03-30, 2003. http://web.archive.org/web/20030603173339/ http%3A//www.darpa.mil/ipto/Solicitations/ PIP%5F03-30.html.

[2] Anita L. Allen. Dredging-Up the Past: Lifelogging, Memory and Surveillance. *University of Chicago Law Review*, 2008.

[3] Gordon Bell and Jim Gemmell. *Total Recall: How the E-Memory Revolution Will Change Everything*. Dutton Adult, 2009.

[4] Emma Berry, Narinder Kapur, Lyndsay Williams, Steve Hodges, Peter Watson, Gavin Smyth, James Srinivasan, Reg Smith, Barbara Wilson, and Ken Wood. The use of a wearable camera, SenseCam, as a pictorial diary to improve autobiographical memory in a patient with limbic encephalitis: a preliminary report. *Neuropsychol Rehabil*, 17(4-5):582–601.

[5] Christian Bizer, Tom Heath, and Tim Berners-Lee. Linked data - the story so far. *Int. J. Semantic Web Inf. Syst.*, 5(3):1–22, 2009.

[6] Vannevar Bush. As we may think. *The Atlantic Monthly*, July 1945.

[7] Daragh Byrne, Barry Lavelle, Aiden R. Doherty, Gareth J.F. Jones, and Alan F. Smeaton. Using bluetooth and GPS metadata to measure event similarity in SenseCam Images. 2007.

[8] Nicholas Carr. *The Shallows: How the Internet is Changing the Way We Think, Read and Remember*. Atlantic Books, London, 2010.

[9] Alan Dix. The ultimate interface and the sums of life. *Interfaces*, Issue 50:16, 2002.

[10] Aiden R. Doherty, Alan F. Smeaton, Keansub Lee, and Daniel P. W. Ellis. Multimodal segmentation of lifelog data. In *Proceedings of Recherche d'Information Assistée par Ordinateur (RIAO) 2007*, Pittsburgh, PA, USA, 2007.

[11] Jim Gemmell, Gordon Bell, Roger Lueder, Steven Drucker, and Curtis Wong. Mylifebits: Fulfilling the memex vision. In *Proceedings of ACM Multimedia 2002*, pages 235–238, Juan-les-Pins, France, December 2002.

[12] Ramesh Jain. Eventweb: Developing a human-centered computing system. *IEEE Computer*, 41(2):42–50, February 2008.

[13] Barry Lavelle, Daragh Byrne, Cathal Gurrin, Alan F Smeaton, and Gareth J F Jones. Bluetooth familiarity: Methods of calculation, applications and limitations. In *Mobile Interaction with the Real World (MIRW 2007), Workshop at the 9th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileMCI07)*, Singapore, September 2007.

[14] Steve Mann and Hal Niedzviecki. *Cyborg: Digital Destiny and Human Possibility in the Age of the Wearable Computer*. Random House, 2001.

[15] Viktor Mayer-Schönberger. *Delete: The Virtue of Forgetting in the Digital Age*. Princeton University Press, Princeton, 2009.

[16] Kieron O'Hara, Mischa Tuffield, and Nigel Shadbolt. Lifelogging: Privacy and empowerment with memories for life. *Identity in the Information Society*, 1(2), 2009.

[17] Owen Sacco and Alexandre Passant. A privacy preference ontology (PPO) for linked data. In *Proceedings of the Linked Data on the Web Workshop (LDOW2011), 20th International World Wide Web Conference*, Hyderabad, India, March 2011. CEUR-WS.

[18] Abigail Sellen and Steve Whittaker. Beyond total capture: a constructive critique of lifelogging. *Communications of the ACM*, 53(5):70–77, 2010.

[19] Ashley Smith. *Who Controls the Past Controls the Future - Life Annotation in Principle and Practice*. PhD thesis, University of Southampton, July 2008.

[20] Ashley Smith, Leslie Carr, and Wendy Hall. An opportunistic approach to adding value to a photograph collection. In *Proceedings of the Fourth International Semantic Web Conference*, Galway, Ireland, November 2005.

[21] Eran Toch, Justin Cranshaw, Paul Hankes Drielsma, Janice Y. Tsai, Patrick Gage Kelley, James Springfield, Lorrie Cranor, Jason Hong, and Norman Sadeh. Empirical models of privacy in location sharing. In *Proceedings of the 12th ACM international conference on Ubiquitous computing*, Ubicomp '10, pages 129–138, New York, NY, USA, 2010. ACM.

[22] Mischa Tuffield. *Telling Your Story: Autobiographical Metadata and the Semantic Web*. PhD thesis, School of Electronics and Computer Science, University of Southampton, July 2010.