MyPlaces: Detecting Important Settings in a Visual Diary

Michael Blighe and Noel E. O'Connor
Centre for Digital Video Processing, Adaptive Information Cluster
Dublin City University, Ireland
{blighem, oconnorn}@eeng.dcu.ie

ABSTRACT

We describe a novel approach to identifying specific settings in large collections of passively captured images corresponding to a visual diary. An algorithm developed for *setting detection* should be capable of detecting images captured at the same real world locations (e.g. in the dining room at home, in front of the computer in the office, in the park, etc.). This requires the selection and implementation of suitable methods to identify visually similar backgrounds in images using their visual features. We use a *Bag of Keypoints* approach. This method is based on the sampling and subsequent vector quantization of multiple image patches. The image patches are sampled and described using Scale Invariant Feature Transform (SIFT) features. We compare two different classifiers, K Nearest Neighbour and Multiclass Linear Perceptron, and present results for classifying ten different settings across one week's worth of images. Our results demonstrate that the method produces good classification accuracy even without exploiting geometric or context based information. We also describe an early prototype of a visual diary browser that integrates the classification results.

Categories and Subject Descriptors

I.4.9 [**Image Processing and Computer Vision**]: Applications; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*clustering*

General Terms

Algorithms

Keywords

Visual Diary, Setting Detection, SIFT, Bag of Keypoints

1. INTRODUCTION

Many people keep a journal or a diary in order to help them remember important aspects of their daily life. Diaries can help people recall what they did as well as how they were feeling at a particular place and time. The explosion of online blogging sites can be viewed as an evolution of the hand written diary in the Internet age. The growing ubiquity of media capture devices means that it is now possible to augment the traditional text-based diary with other content such as images and video clips. This is attractive since the inclusion of such content can potentially aid us in reliving important events, more so than is possible with plain text. A parallel can be drawn to the way we create and arrange photograph albums to help us remember a family holiday or wedding, for example. The proliferation of digital cameras and camera-phones means that taking pictures has never been easier thereby fueling this growing trend of multi-media life logging. Providing tools to help automate content organisation and management is thus increasingly important in order to help users tame the inevitable information overload. However, the development of methods for managing digital photos (and video) has not kept pace with acquisition technology, thereby severely degrading the practical usefulness of visual diaries that rely on these photos.

Significant research effort is currently being invested in the capture and retrieval of life logs in order to automatically generate a record of a user's daily life [24] [15]. Much of the work focuses on using context and content information in order to infer details about one's daily activities [31]. Context information is usually generated using location-based sensing from a mobile phone, GPS device, or other similar sources. Content information is usually derived from the analysis of passively captured audio/visual data, most often in the form of video or digital photos. Using photos, for example, one can construct a visual diary of an individual's life. For a single day, this might consist of a sequence of images providing a visual summary of the most important aspects of the day. The underlying challenge is to be able to manage, organise, and search large volumes of photos to judiciously select and present representative samples in a visually coherent manner. Within this broad challenge, a key objective is to be able to identify these representative samples in the first place – they typically need to be selected from thousands of images representing an individual's day (and ultimately from millions over a lifetime) and they should correspond to images that are somehow 'important' to the owner.

In this work, we focus on personal image collections captured via a passive image capture device – Microsoft's Sense-

CIVR '08 Niagara Falls, Canada

Cam. We have developed an algorithm to perform *setting detection*. A *setting* in this context refers to those images taken at the same location in the real world (e.g. in the dining room at home, in front of the computer in the office, in the park, etc.) that have been flagged by a user as being important to him/her for some reason. Examples of two distinct settings can be seen in Figure 1. Detecting such settings is a key enabling technology to allow us to structure the large numbers of images that passive capture devices collect that in turn allows us to help the user in constructing and maintaining a visual diary.

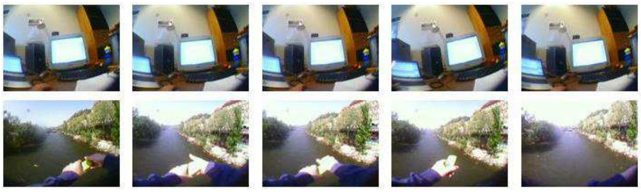


Figure 1: Sample SenseCam images showing two distinct settings

To perform setting detection, it is necessary to select and implement suitable methods to identify visually similar backgrounds in SenseCam images using visual features. Note that in this paper we constrain ourselves to using only visual SenseCam data, as this represents a technology component that can be deployed to other devices (e.g. a mobile phone running the Campaignr software [19] – see Section 2.2). Setting (or more generally location) detection using a combination of image and other sensor data is underway and will be reported elsewhere. Similar work has been carried out by [36] where the authors automatically detected similar locations in movies. In our application, however, the images produced by the SenseCam are of a low resolution and suffer from a number of quality issues such as blurring and distortion. Our algorithm was developed using SIFT features as they have proven their usefulness in a variety of object recognition tasks [26]. SIFT image features are not affected by many of the limitations of other interest point detection methods, such as changes in scale and rotation. Therefore, they provide an extremely useful method to detect similar settings in different SenseCam images, even if the background has been displaced or distorted.

This paper significantly extends our initial preliminary experiments in this area, reported in [5] (see Section 2.4 for details). The key contributions of this paper are the provision of a novel method of detecting the settings in a Visual Diary application, and the description of how this can be integrated with our other diary tools. Our setting detection algorithm can be viewed as one of a number of tools which combine to generate an intuitive and simple method of browsing a large volume of SenseCam images. In addition, by detecting unusual or important settings, we can provide the user with a novel method of reviewing events which occurred at places they have deemed to be important in the context of their day to day activities.

The rest of this paper is organised as follows. In Section 2, we review related research in this area. In section 3, we outline an approach to setting detection. Section 4 describes the experiments we performed and results obtained. This is followed by a discussion in Section 5, whilst future work and conclusions are discussed in Section 6.

2. RELATED RESEARCH

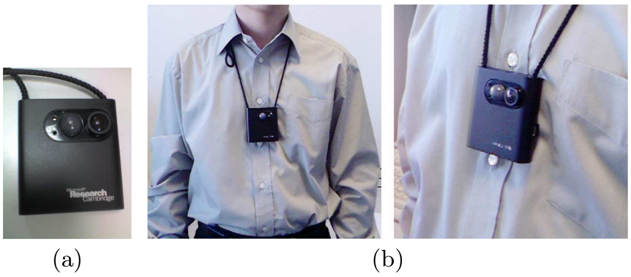
2.1 Lifelogging

Many studies have been undertaken which examine how people spend their time and how this is related to daily experiences, but there is no generally accepted method for gathering this data. Studies generally focus on the well-being of the population at large and attempt to analyse this using surveys or time-budget studies [21] [14]. Other studies rely on global reports of happiness or satisfaction with life in general, or within specific domains such as work and family [8] [1]. Although there is no universally agreed approach to data gathering, a number of methods have been proposed. The *Day Reconstruction Method* (DRM) assesses how people spend their time and how they experience the various activities and settings of their lives by combining features of time-budget measurement and experience sampling [22]. Participants systematically reconstruct their activities and experiences of the previous day by constructing a diary consisting of a sequence of episodes. They then describe each episode by answering questions about the situation and about the feelings they experienced. The Experience Sampling Method (ESM) [9] is designed to measure the quality of people's lives by prompting them to record where they are, what they are doing, and how they feel several times throughout the day. The technique is reported to provide a rich description of a sample of moments in respondents' lives, while avoiding the distortions that affect delayed recall and evaluation of experiences. However, experience sampling is expensive, involves high levels of participant burden, and provides little information about uncommon or brief events, which are rarely sampled. The DRM involves a similar burden on the particpants and faces similar problems in practice.

Although these studies are difficult to carry out, their utility is not in question. Kahneman et al. [22] describe how this information is useful to medical researchers for assessing the onset and development of different illnesses and the health consequences of stress; to epidemiologists interested in social and environmental stressors; to economists and policy researchers for evaluating policies and for valuing non-market activities; and to anyone who wishes to measure the well-being of society. We believe that the next logical step in overcoming the difficulties associated with traditional methods of gathering the required data is the use of passive capture devices such as the SenseCam.

2.2 Passive Image Capture

Many researchers have started work on developing passive capture devices - cameras which automatically take pictures without any user intervention. Passive capture lets people record their experiences without having to operate recording equipment, and without having to give recording a conscious thought. The advantages of this are increased coverage of, and improved participation in, the event itself. Detailed technical information about SenseCam can be found in [16]. We use version 2.3 of the SenseCam shown in Figure 2(a). To facilitate the capture of images in a passive manner, the SenseCam is worn around the neck as shown in Figure 2(b). It takes pictures automatically every fifty seconds (this is the default setting and it can be changed to a minimum of every five seconds). It also has a number of sensors onboard the device which trigger capture more frequently if necessary. The sensors include a passive infra-red detector, similar to that used in home alarm systems, which can detect people or other warm objects directly in front of the individual wearing the camera, an accelerometer which captures data in the X, Y & Z directions, a digital light sensor and a temperature sensor. In a typical day, the SenseCam will capture 2,000-3,000 photos. Other examples of passive capture include StartleCam which is a wearable video camera, computer, and sensing system which also passively captures images depending on certain events detected by the sensors on the device [18]. Similarly, the Campaignr project [19] is a software framework for mobile phones that enables owners of smartphones (specifically Symbian Series 60 3rd edition phones) to participate in data gathering campaigns including automatic image capture.



(a) (b)

Figure 2: (a) Microsoft SenseCam; (b) User wearing the SenseCam

2.3 Image Collection Management

Passive capture of photos presents new problems in terms of how to manage and organise the massively increased volume of images captured [4]. Traditional systems for content-based image retrieval are not up to this task. Naaman et al. [29] describe how the photo collection management problem can be categorised into tools which enable easy annotation of photos, tools which allow fast visual scanning of the images and content-based tools. They also identify the problems associated with each of these types of systems, such as difficulties for consumers with annotation, inability of tools to allow fast visual scanning to scale to many thousands of images and the semantic gap in relation to content based tools. As we have found in the MediAssist [7] project, the manual annotation of approximately 11,000 images required the work of up to 10 individuals over an extended period of time. When we consider that the SenseCam produces approximately 14,000 images per week, we can see that this approach is not scalable in practice. Tools allowing fast visual scanning are not sufficient in isolation. A software interface has been provided by Microsoft to allow the fast visual playback of a day's worth of SenseCam images, but it does not allow sufficient interaction and very often the user spends extended periods of time watching repetitive events and images. We believe a better solution is to provide the user with tools to construct a visual diary based on images selected using content based analysis tools. In this way, the user controls the process but his/her effort is minimised. In previous work in our group, reported in [11], we described a method of segmenting lifelog data into events using low-level MPEG-7 features extracted from the image and sensor data from the SenseCam. However, this made no attempt to recognise similar locations or settings that have been identified by the user as important, our key objective in this work.

2.4 Object & Scene Detection

The earliest work on appearance-based object recognition mainly utilized global descriptions such as colour or texture histograms [13]. The main drawback of such methods is their sensitivity to real-world sources of variability such as viewpoint and lighting changes, clutter, and occlusions. For this reason, global methods were gradually replaced by methods which utilised local features and SIFT-based approaches have emerged as one of the most popular approaches. For example, in [40], a generative probabilistic approach using a Gaussian Mixture Model is presented to improve the results of Lowe's original work. In [37], a combination of SIFT keypoints and MPEG-7 features extracted from the same interest point is used to obtain better results than either descriptor on their own. Low quality images, large view and scale changes, and blur negatively influence these results.

Regarding scene detection, most works use color and texture information to perform classification/retrieval. Vailaya et al. [39] used histograms of different low-level cues to perform scene *classification*. Different sets of cues were used depending on the two-class problem at hand: global edge features were used for city vs landscape classification, while local color features were used in the indoor vs outdoor case. This approach is not really scalable to the multi-class approach required for setting detection. Boutell et al. [6] use only LUV colour moments in a 7 × 7 block layout to perform multi-label scene classification, but the use of colour means that their system is not very robust to viewing angle or lighting changes. Of more interest in the context of our work is [36] where the authors describe a system to match camera shots which are images of the same real world location in a film. They use two features: one based on interest point neighbourhoods, the other based on the Maximally Stable Extremal Regions of Matas et al. [27]. In both cases an elliptical image region is used to compute the invariant descriptor. Their system also employs semi-local and global contraints (e.g. using epipolar geometry) to boost matching accuracy. However, processing time in this system is of the order of hundred's of hours and significant tuning of separate processes is necessary in order to create a working system.

In our previous preliminary work [5], the X-means algorithm was used to cluster SIFT descriptors extracted from user generated training data. Image signatures were then created from the cluster centres and the Earth Mover's distance used to calculate the distance between these signatures. However, the number of clusters produced by X-means did not provide enough discriminative power to sufficiently model the settings in question. The work reported in this paper extends this initial investigation by building upon the work of [10], using different classifiers and more classes (ten) than was used in their work. Finally, to the best of our knowledge, the method we have used has not been applied in this area before.

3. SETTING DETECTION ALGORITHM

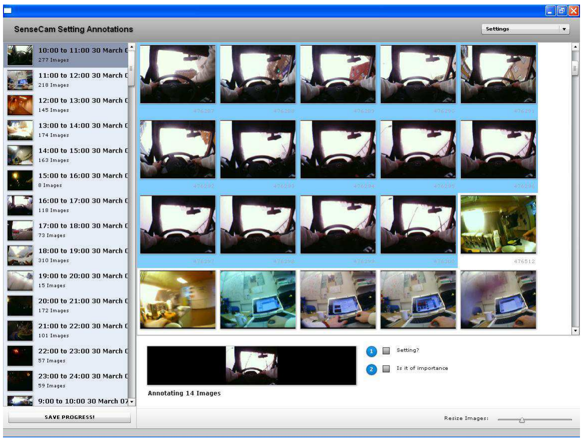
In order to perform setting detection, we utilize an approach similar to that outlined by [10]. The basic idea is that a set of local image patches is sampled using some method (e.g. densely, randomly, using a keypoint detector) and a vector of visual descriptors is evaluated on each patch independently. The resulting distribution of descriptors in descriptor space is then vector quantized against a pre-specified codebook to convert it to a histogram of votes for codebook centres and the resulting global descriptor vector is used as a characterisation of the image (e.g. as a feature vector on which to learn an image classification rule based on a multi-class classifier). A summary of the main steps used in our approach are as follows:

- Annotation of images into pre-defined settings;
- Sample multiple image patches from each image;
- Extract patch feature vectors from all the points using the SIFT descriptor;
- Generate codebooks with k-means clustering over extracted patch feature vectors;
- Assign all patch feature vectors to the nearest codebooks, and convert a set of patch feature vectors for each image into one histogram vector of assigned codebooks;
- Train a multi-class classifier with all the histogram vectors in the training data;
- Classify all the histogram vectors of the test images into the appropriate setting by applying the trained clasification rules.

This approach is designed to maximise classification accuracy while minimising computational effort. The vocabulary used should be large enough to distinguish relevant changes in image parts, but not so large as to distinguish irrelevant variations such as noise. Our goal is to use a vocabulary that allows good categorisation performance on a given training dataset. Each of these steps is described in more detail below.

3.1 Setting Annotation

In the first step of our approach, the user reorganizes a single week of SenseCam images to reflect the real settings depicted that are particularly important to him/her. This is performed using a simple annotation tool, see Figure 3, which allows the user to update the setting information for each image. The tool is simple and intuitive to use. The user can visually scan over all images very quickly, easily identifying collections of images which constitute an important setting. Note that we asked the user to provide an importance score between 0 (not very important) - 5 (very important) for each setting. We do not use this score yet in our work except to draw some preliminary anecdotal evidence of the nature of important settings in Section 5. In these experiments, a week's images consisting of 14,965 images were annotated in this way in approximately 30 minutes. The objective here is to provide the user with a low-overhead mechanism for organising his/her visual diary in terms of specific settings of interest. Given this user generated training data, we train a multi-class classifier using the bags of keypoints as feature vectors.


SenseCam Setting Annotations

10:00 to 11:00 30 March
277 Images

11:00 to 12:00 30 March
218 Images

12:00 to 13:00 30 March
145 Images

13:00 to 14:00 30 March
174 Images

14:00 to 15:00 30 March
163 Images

15:00 to 16:00 30 March
8 Images

16:00 to 17:00 30 March
118 Images

17:00 to 18:00 30 March
73 Images

18:00 to 19:00 30 March
310 Images

19:00 to 20:00 30 March
15 Images

20:00 to 21:00 30 March
172 Images

21:00 to 22:00 30 March
101 Images

22:00 to 23:00 30 March
57 Images

23:00 to 24:00 30 March
59 Images

9:00 to 10:00 30 March 07

Annotating 14 Images

1 □ Setting?

2 □ Is it of importance

Figure 3: SenseCam Setting Annotation Tool. In this particular instance, the user is annotating a setting where the individual wearing the SenseCam is driving.

3.2 Feature Extraction

Similar to terms in a text document, an image has local interest points, or keypoints, defined as salient image patches (small regions) that contain rich local information of the image. Denoted by small crosses in the three images in Figure 4, keypoints are usually around the corners and edges of image objects. Mikolajczyk et al. [28] have compared several descriptors for matching and found that SIFT descriptors perform best so we continue with SIFT on this basis. The SIFT descriptor computes a gradient orientation histogram within the support region. For each of 8 orientation planes, the gradient image is sampled over a 4×4 grid of locations, thus resulting in a 128-dimensional feature vector for each region. A Gaussian window function is used to assign a weight to the magnitude of each sample point. This makes the descriptor less sensitive to small changes in the position of the support region and puts more emphasis on the gradients that are near the centre of the region. To ensure robustness to illumination changes, the descriptors are made invariant to illumination transformations of the form $aI(x) + b$ by scaling the norm of each descriptor to unity [26].

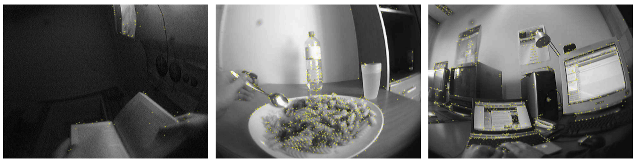


Figure 4: Sample images from 3 settings showing detected keypoints

3.3 Visual Vocabulary Construction

Csurka et al. [10] describe the construction of a visual vocabulary as a way of constructing a feature vector for classification that relates new descriptors in query images to descriptors previously seen in training. An extreme example of this approach would be to compare each query descriptor to all of the training descriptors in the database.

For most applications this is not feasible due to the huge number of training and test descriptors involved (approx. 5,000,000 keypoint descriptors in our experiments) and the large amount of processing time this would require.

Instead, we use the vector quantization technique which clusters the keypoint descriptors in their feature space into a large number of clusters using the K-means clustering algorithm [12] and encodes each keypoint by the index of the cluster to which it belongs. The algorithm proceeds by partitioning the input points into k initial sets, either at random or using some heuristic. It then calculates the centroid of each set and constructs a new partition by associating each point with the closest centroid. Then the centroids are recalculated for the new clusters, and the algorithm repeated until convergence, which is obtained when the points no longer switch clusters (or alternatively centroids are no longer changed). The choice of k-means is justified because the Euclidean distance is meaningful in the SIFT-descriptor space. One problem with the k-means algorithm is that the number of clusters, k, is an input parameter. Methods do exist to facilitate the estimation of the number of clusters [33], however, in this scenario we are not really interested in a correct clustering in the sense of feature distributions, but rather in accurate categorisation into the correct settings.

Each cluster generated is representative of a visual word which represents a specific local pattern shared by the keypoints in that cluster. The clustering process, therefore, generates a visual-word vocabulary which describes different local image patches in the images. The number of clusters generated via the k-means clustering determines the size of the vocabulary, which can vary from hundreds to over tens of thousands. We can then represent each image in the data set as a histogram of visual words drawn from the vocabulary. This representation is analogous to the bag-of-words document representation in terms of form and semantics. Both representations are sparse and high-dimensional, and just as words convey meanings of a document, visual words reveal local patterns characteristic of the whole image.

The bag-of-keypoints representation can be converted into a visual-word vector similar to the term vector of a document. The visual-word vector may contain the presence or absence information of each visual word in the image, the count of each visual word (i.e., the number of keypoints in the corresponding cluster), or the count weighted by other factors. Visual-word vectors are used in our image classification approach.

3.4 Classification

Once descriptors have been assigned to clusters to form feature vectors, we can use different classification methods in the image descriptor space. The problem is effectively reduced to that of multi-class supervised learning, with as many classes as defined visual categories. We have chosen to use two classification algorithms in this work - the K Nearest Neighbour (KNN) classifier and the Multiclass Linear Perceptron (MLP) algorithm [12].

In the KNN algorithm, a setting is classified by a majority vote of its neighbours, with the setting being assigned to the class most common amongst it's k nearest neighbours. The neighbours are taken from the training data for which the correct classification is known. In order to identify neighbours, the settings are represented by position vectors in a multidimensional feature space. It is usual to use the Euclidean distance, though other distance measures, such as the Manhattan distance could in principle be used instead.

Settings annotated by user	Total No. of Images	Training Total	Testing Total
Reading in bed	62	12	50
Having dinner	46	9	37
At a restaurant	118	24	94
Sitting in the park	37	7	30
Eating ice cream	44	8	36
Working on computer	130	24	106
At a cafe	108	22	86
Reading in the castle grounds	100	20	80
On an aeroplane	173	24	149
On a train	142	22	120

Table 1: Three databases containing different images were created. This table show's the total number of images used for training and testing for each manually annotated setting in each of the three databases.

The Perceptron algorithm [34] is a well studied and popular classification learning algorithm. Despite its age and simplicity it has proven to be quite effective in practical problems. The Perceptron maintains a single hyperplane which separates positive instances from negative ones. In [12], this binary perceptron algorithm was extended to construct a MLP algorithm considering all classes at once. In this case, c linear discriminant functions have to be defined, i.e.,

$$f_i(x) = w_i^T x + b_i \; i = 1, ..., c, \tag{1}$$

where w and b denote the weight vector and threshold of the i^{th} discriminant function. Now, for some input vector x, if $f_i(x) > f_j(x)$ for all $j \neq i$, this vector is assigned to the i^{th} class.

4. EXPERIMENTAL RESULTS

The experiments were carried out on a total of 14,965 images taken by the SenseCam over a period of one week. Using the annotation tool, these images were classified into different settings by the owner of the images. The user was not given any strict instructions as to how to should perform the annotation. The concept of a Visual Diary and the definition of settings was explained and it was left up to him/her to judge what they considered an important setting in this context. A total of ten different settings were found over that particular week and these are listed in Table 1. Sample images from each setting can be seen in Figures 6 and 8. The total number of images for all ten settings is 2,880. These images were then divided into three sets of testing and training images labeled *T1*, *T2*, and *T3*. Each set contained a total of 960 images, consisting of 172 training images and 788 testing images. The training images were randomly chosen across all ten settings. An experiment was then run, using the procedure outlined above, on each of the three sets of images. In order to determine the appropriate vocabulary size, a preliminary experiment was performed on the first database of images, *T1*. Here, we examined the overall error rate for both classifiers as a function of the number of clusters k. The results can be seen in Figure 5. Based on these results, we used a value of $k = 1900$, as this was the value which minimised the error rate for both classifiers.

Our final experiment used the same approach but with lower dimensional projections of the original SIFT feature descriptors. Principal Component Analysis (PCA) [20] was used, and an experiment performed to determine the number

% Variance	75%	80%	85%	90%	95%
KNN	0.632	0.6282	0.6028	0.5888	0.5812
MLP	0.5381	0.5083	0.5025	0.5711	0.5343
Dimensions	23	29	36	46	63

Table 2: This table show's the overall error rate for each classifier for descriptors in the reduced PCA space. It also shows the number of dimensions required to retain a certain percentage of the variance.

of components to keep for different percentages of variance. The SIFT keypoints extracted from the 14,965 images were projected into Principal Component space and the components required to retain a certain percentage of variance was examined. The same experiment was then run in the lower dimensional space on database *T1* and the error rate examined to estimate the usefulness of the lower dimensional descriptors. The results can be seen in Table 2. Finally, in order to evaluate our multi-class classifiers, precision and recall figures were calculated based on the ground-truth generated using the annotation tool. The number of images used for training and testing for each setting are also shown in Table 1.

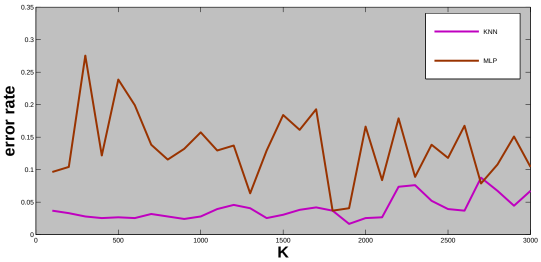


Figure 5: The overall error rate found for different choices of k for both classifiers.

The figures for precision and recall can be seen in Tables 3 & 4. These tables show the precision and recall figures for each setting across all three databases. All ten settings were detected by the system using both algorithms. Both systems performed well, with the lowest precision value being 58.56% for the *Reading in the castle grounds* setting in database *T2* and the lowest recall figure being 50% for the *Reading in bed* setting in database *T2* using the MLP classifier. Using the KNN classifier, the lowest precision value is 59.52% for the *Reading in bed* setting in database *T2* and the lowest recall figure is 44.44% for the *Eating ice cream* setting in database *T2*.

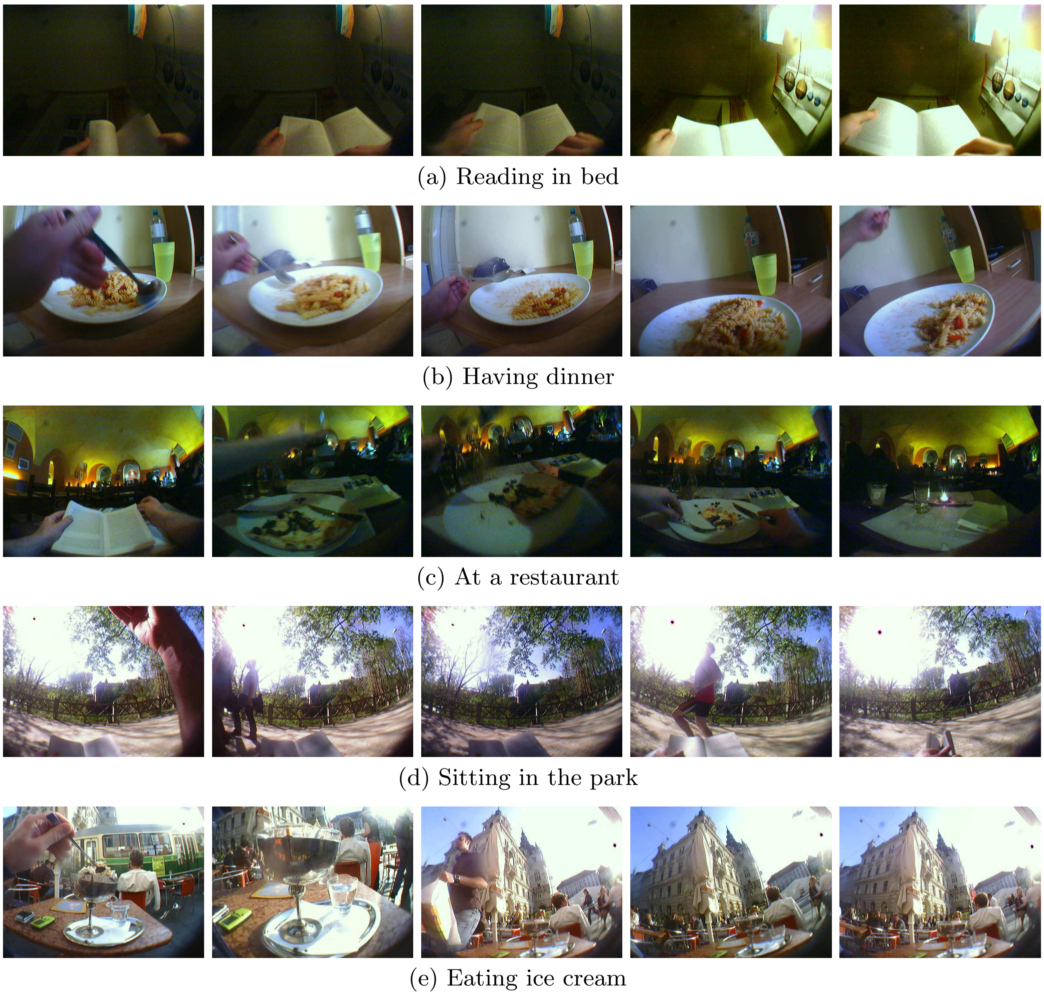
5. DISCUSSION

When one considers the challenging nature of the dataset, the results obtained are very encouraging. The images used contain significant viewpoint, lighting, blur and affine changes. However, the system was able to find matches for all ten settings with high rates of precision and recall. This would seem to justify the overall approach taken in this work, however, a number of points are open to discussion.

With a bag-of-keypoints approach, we are faced with a number of implementation choices. These include how to sample image patches, what visual patch descriptor to use, and how to classify images based on the resulting global image descriptor. In this work, we used the SIFT features to sample (using Difference of Gaussian's) and describe the image patches. SIFT features have been used in many applications for object detection and recognition [32] [17]. However, they have not very often been used as a tool to detect settings across the entire image. The very nature of SenseCam images themselves means that they are inherently of poor quality, with many blurry shots, significant changes in lighting, etc. Therefore, it was important that the training images used in these experiments provided a realistic data set with which to describe the settings in question. Indeed, the variation in results between the different databases in our experiments would seem to confirm this. This is in stark contrast to most object detection systems using SIFT, where the use of high quality training, or *model*, images is crucial [2] [35]. We believe the use of SIFT is justified in our work due to the excellent results achieved and the large body of existing work in similar areas. However, despite this it would be naive to ignore other algorithms, such as Speeded Up Robust Features (SURF), which are reported to give better performance at greater speeds than SIFT [3]. In addition, it has been suggested that randomly sampled image patches are more discriminant than keypoint based ones and this should be further investigated in our work.

Another issue which can impact performance is the size of the visual-word vocabulary. This is controlled by the number of clusters generated. Two contradictory considerations are at work here – the discriminative nature of the descriptor versus it's ability to generalise – so choosing the right vocabulary size involves a trade-off. With a small vocabulary, the visual-word feature is not very discriminative because dissimilar keypoints can map to the same visual word. As the vocabulary size increases, the feature becomes more discriminative, but meanwhile less generalisable and forgiving to noise, since similar keypoints can map to different visual words. Using a large vocabulary also increases the cost of clustering keypoints, computing visual-word features, and running supervised classifiers. There is no consensus as to the appropriate size of a visual-word vocabulary. The vocabulary size used in existing works varies from several hundreds [25] [40], to thousands and tens of thousands [38] [41]. Csurka et al. [10] found no significant improvement in performance as they moved from $k = 1000$ to $k = 2500$, so they used $k = 1000$ as it provided a good trade off between speed and accuracy. In our experiments a value of $k = 1900$ was used to minimise the overall system error for both classifiers. However, it is difficult to directly compare the two methods due the different corpus and classification methods used. The error rate for the MLP classifier was somewhat erratic across all values of k so it is difficult to determine an appropriate value using this classifer. Using the KNN classifer, the error rate tended to fall until we reached values of $k = 1900/2000$, before it began to rise slightly. Further testing is necessary to determine if improved results can be obtained using a different vocabulary size, however, due to the lower overall error rate and it's more predictable nature, these experiments would seem to indicate that the KNN classifier is the better choice.

Regarding the use of PCA, the high error rates indicate that this technique is not suitable for use in this approach. This is an interesting finding as [30] obtained good results using PCA in a similar fashion. In their work, they used the



(a) Reading in bed

(b) Having dinner

(c) At a restaurant

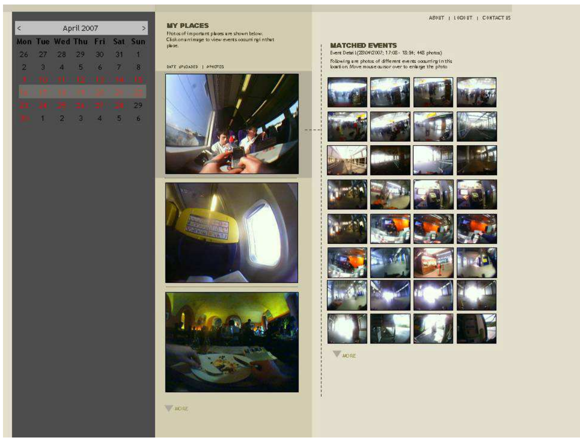
(d) Sitting in the park

(e) Eating ice cream

Figure 6: Sample Images from each of the 10 settings

Setting	**Precision (T1)**	**Recall (T1)**	**Precision (T2)**	**Recall(T2)**	**Precision (T3)**	**Recall (T3)**
Reading in bed	98.04%	100%	59.52%	100%	60.98%	100%
Having dinner	100%	100%	100%	64.87%	100%	89.19%
At a restaurant	89.52%	100%	82.72%	96.8%	86.92%	98.94%
Sitting in the park	100%	96.66%	100%	56.66%	100%	46.66%
Eating ice cream	100%	97.22%	100%	44.44%	100%	47.22%
Working on computer	100%	95.28%	100%	75.47%	100%	100%
At a cafe	100%	97.67%	98.75%	91.86%	100%	90.7%
Reading in the castle grounds	100%	100%	100%	75%	100%	90%
On an aeroplane	99.33%	100%	69.9%	96.64%	91.41%	100%
On a train	100%	96.66%	91.89%	85%	98.28%	95%

Table 3: Precision and Recall figures for the KNN Classifier



Mon	Tue	Wed	Thu	Fri	Sat	Sun
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	1	2	3	4	5	6

MY PLACES

Photos of important places are shown below. Click on an image to view events occurring in that place.

DATE UPLOADED | #PHOTOS

MATCHED EVENTS

Event Detail (22/04/2007; 17:08 - 18:54; 448 photos)

Following are photos of different events occurring in this location. Move mouse cursor over to enlarge the photo.

Figure 7: Weekly Summary of Visual Diary: Once important settings have been detected, the interesting events which occurred while the user was in these locations can be highlighted. These images have been detected using a separate event detection process and have been matched to particular settings based on their timestamps.

first 50 components, however, no information is provided as to how this number was determined. Other authors [23] have performed PCA on the 41 × 41 pixel patches that are passed through the SIFT interest point detector, instead of on the descriptor itself. Again, the results achieved here using very low-dimensional descriptors (e.g. 20) were good. Further investigations are necessary to determine if PCA can be successfully used with SenseCam images. As mentioned previously, the lower dimensional SURF descriptor will also be investigated.

The main novelty of our work is the provision of a facility to aid a user in browsing a Visual Diary. An ancillary benefit of performing setting detection is that once settings have been determined over a long period of time, infrequently occurring settings can be given more importance in the diary, as they are probably of more interest to the user. In general terms, analysing the importance scores provided, we determined that the important settings are the ones that don't occur on a regular or routine basis. The *On an aeroplane* or *At a restaurant* settings are examples of settings which were deemed to be unusual or important to the user. The ability to detect these important settings across a weeks images allows us to highlight events occurring within those settings to the user in a simple and efficient manner. Figure 7 shows a prototype system currently under development to facilitate a user in browsing important settings and the events that occur within these settings across their image collection. By selecting a particular day, or a whole week as in the example shown, the user is presented with settings they have deemed to be important from that particular day or week. In addition, by selecting a particular setting, the events which were detected using the approach in [4], are highlighted to the user. Mousing over an image enlarges it for easier viewing. This facility should provide a much more interesting Visual Diary for the user to browse through the settings and important events of their daily life.

6. CONCLUSIONS

We presented a novel approach to *Setting Detection* in SenseCam images in order to help construct a useful visual diary. We developed a simple annotation tool to allow the user to quickly and efficiently annotate settings considered important. Using the bag-of-keypoints approach, we created an image descriptor for each of the training images for each setting and then learned a classification model using two different multiclass classification algorithms. The classification algorithms used were KNN and MLP. Finally, using the learned model, we classified the test images. Much future work remains. As previously mentioned, other methods of sampling and describing the image patches will be investigated. In addition, by integrating our work on setting detection with other work on event detection in SenseCam images, such as [4] & [11], we hope to create a really useful visual diary capable on fulfilling the vision described in the *Discussion* section.

Acknowledgments

The research leading to this paper was supported by the European Commission under contract FP6-027026 (K-Space), Microsoft Research and Science Foundation Ireland under grant number 03/IN.3/I361.

7. REFERENCES

[1] F. M. Andrews and S. B. Whithey. *Social Indicators of Well-Being - Americans Perceptions of Life Quality*. Plenum, New York, 1976.

[2] A. Baumberg. Reliable feature matching across widely separated views. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 774–781, 2000.

[3] H. Bay, T. Tuytelaars, and L. V. Gool. Surf - speeded up robust features. In *9th European Conference on Computer Vision*, May 2006.

[4] M. Blighe, H. L. Borgne, and N. O'Connor. Exploiting context information to aid landmark detection in sensecam images. In *2nd International Workshop on Exploiting Context Histories in Smart Environments - Infrastructures and Design (ECHISE)*, September 2006.

[5] M. Blighe, N. O'Connor, H. Rehatschek, and G. Kienast. Identifying different settings in a visual diary. In *9th International Workshop on Image Analysis for Multimedia Interactive Services*, May 2008.

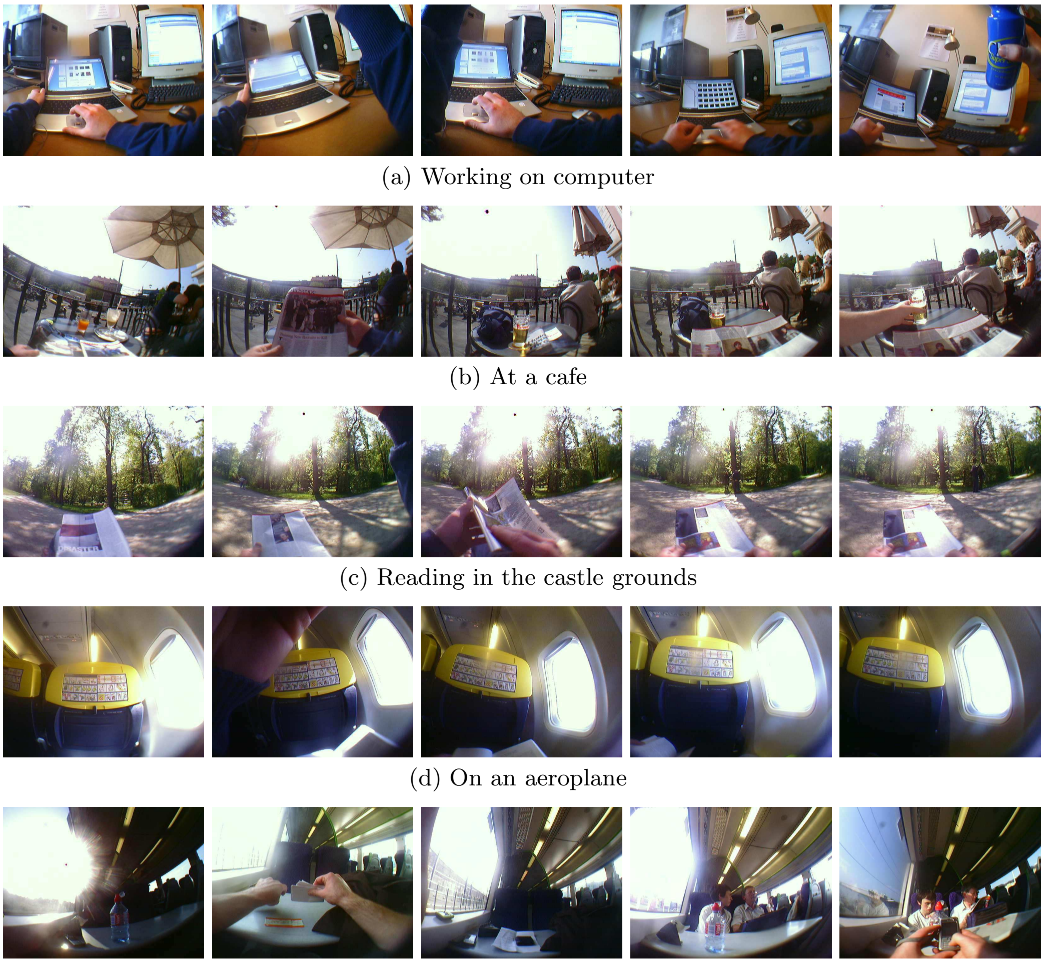
[6] M. Boutell and J. Luo. Bayesian fusion of camera metadata cues in semantic scene classification. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 623 – 630, June 2004.

[7] N. M. C. Gurrin and G. Jones. Mediassist: Managing personal digital photo archives. ERCIM News, July 2005. No. 62.

[8] A. Campbell. *The Sense of Well-Being in America.* McGraw-Hill, New York, 1981.

[9] M. Csikszentmihalyi and R. E. Larsen. The experience sampling method. *New Directions for Methodology of Social and Behavioral Science*, 15:41–56, 1983.

[10] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray. Visual categorization with bags of keypoints.



(a) Working on computer

(b) At a cafe

(c) Reading in the castle grounds

(d) On an aeroplane

(e) On a train

Figure 8: Sample Images from each of the 10 settings

Setting	**Precision (T1)**	**Recall (T1)**	**Precision (T2)**	**Recall(T2)**	**Precision (T3)**	**Recall (T3)**
Reading in bed	100%	94%	100%	50%	100%	58%
Having dinner	90.24%	100%	100%	78.38%	100%	97.3%
At a restaurant	100%	96.8%	97.75%	92.55%	98.92%	97.87%
Sitting in the park	100%	96.66%	83.87%	86.66%	100%	46.66%
Eating ice cream	100	94.44%	95%	52.77%	57.14%	100%
Working on computer	99.04%	98.11%	100%	91.51%	94.64%	100%
At a cafe	100%	84.88%	100%	86.04%	88.17%	95.35%
Reading in the castle grounds	80.8%	100%	58.56%	98.75%	96.2%	95%
On an aeroplane	97.96%	96.64%	98.55%	91.28%	96.75%	100%
On a train	95.9%	97.5%	77.48%	97.5%	98.26%	94.17%

Table 4: Precision and Recall figures for the MLP Classifier

In *European Conference on Computer Vision*, May 2003.

[11] A. Doherty, A. Smeaton, K. Lee, and D. Ellis. Multimodal segmentation of lifelog data. In *RIAO 2007 - Large-Scale Semantic Access to Content (Text, Image, Video and Sound)*, Pittsburgh, PA, USA, 30 May - 1 June 2007 2007.

[12] R. Duda, P. Hart, and D. Stork. *Pattern Classification*. Wiley, 2001.

[13] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker. The qbic project: Querying images by content using color, texture and shape. In *SPIE Storage and Retrieval of Image and Video Databases*, pages 171–181, 1993.

[14] L. Flood. Household, market, and nonmarket activities - procedures and codes for the 1993 time-use survey. Technical Report vol. VI., Uppsala Univ. Dept. Economics, Uppsala, Sweden, 1997.

[15] J. Gemmell, R. Lueder, and G. Bell. Living with a lifetime store. In *ATR Workshop on Ubiquitous Experience Media*, September 2003.

[16] J. Gemmell, L. Williams, K. Wood, R. Lueder, and G. Bell. Passive capture and ensuing issues for a personal lifetime store. October 2004.

[17] K. Grauman and T. Darrell. Efficient image matching with distributions of local invariant features. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 627–634, 2005.

[18] J. Healey and R. Picard. Startlecam: A cyberbetic wearable camera. October 1998.

[19] A. Joki, J. Burke, and D. Estrin. Campaignr - a framework for participatory data collection on mobile phones. Technical Report 770, Centre for Embedded Network Sensing, University of California, Los Angeles, October 2007.

[20] I. Jolliffe. *Principal component analysis*. Springer series in statistics. Springer, 2nd edition, 2002.

[21] F. Juster and F. Stafford. Time, goods, and well-being. *Institute for Social Research*, pages 397–414, 1985.

[22] D. Kahneman, A. B. Krueger, D. A. Schkade, N. Schwarz, and A. Stone. A survey method for characterizing daily life experience - the day reconstruction method. *Science*, 306:1776–1780, December 2004.

[23] Y. Ke and R. Sukthankar. Pca-sift - a more distinctive representation for local image descriptors. In *Computer Vision and Pattern Recognition*, volume 2, pages 506–513, 2004.

[24] M. Lamming and M. Flynn. Forget-me-not: Intimate computing in support of human memory. In *FRIEND21, Int. Symp. Next Generation Human Interface*, pages 125–128, February 1994.

[25] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features - spatial pyramid matching for recognizing natural scene categories. In *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, volume 2, pages 2169–2178, 2006.

[26] D. Lowe. Distinctive image features from scale-invariant keypoints. In *International Journal of Computer Vision*, volume 60(2), pages 91–110, 2004.

[27] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *The British Machine Vision Conference*, pages 384–393, 2002.

[28] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, February 2005.

[29] M. Naaman, S. Harada, Q. Wang, H. Garcia-Molina, and A. Paepcke. Context data in geo-referenced digital photo collections. *MM'04*, pages 10–16, October 2004.

[30] A. Noulas and B. J. A. Kröse. Unsupervised visual object class recognition. In *Advanced School of Computing and Imaging Conference*, Lommel, Belgium, 2006.

[31] N. O'Hare, C. Gurrin, G. Jones, and A. Smeaton. Combination of content analysis and context features for digital photograph retrieval. *2nd IEE European Workshop on the Integration of Knowledge, Semantic and Digital Media Technologies*, November 2005.

[32] A. Opelt, M. Fussenegger, A. Pinz, and P. Auer. Weak hypotheses and boosting for generic object detection and recognition. In *European Conference on Computer Vision*, volume 2, pages 71–84, 2004.

[33] D. Pelleg and A. Moore. X-means - extending k-means with efficient estimation of the number of clusters. In *17th International Conference on Machine Learning*, pages 727–734, 2000.

[34] F. Rosenblatt. The perceptron - a probabilistic model for information storage and organization in the brain. *Psychological Review*, 65:386Ű407, 1958. (Reprinted in Neurocomputing (MIT Press, 1988)).

[35] F. Schaffalitzky and A. Zisserman. Multi-view matching for unordered image sets, or "how do i organize my holiday snaps?". In *7th European Conference on Computer Vision*, volume 1, pages 414–431, 2002.

[36] F. Schaffalitzky and A. Zisserman. Automated location matching in movies. *Computer Vision and Image Understanding*, 92:236–264, 2003.

[37] P. Schugerl, R. Sorschag, W. Bailer, and G. Thallinger. Object re-detection using sift and mpeg-7 color descriptors. In *International Workshop on Multimedia Content Analysis and Mining*, pages 305–314. Springer, July 2007.

[38] J. Sivic and A. Zisserman. Video google - a text retrieval approach to object matching in videos. In *9th IEEE Int'l Conf. on Computer Vision*, volume 2, 2003.

[39] A. Vailaya, M. Figueiredo, A. Jain, and H. Zhang. Image classification for content-based indexing. In *IEEE Trans. on Image Processing*, volume 10 of *1*, pages 117–130, 2001.

[40] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid. Local features and kernels for classifcation of texture and object categories: An in-depth study. Technical Report RR-5737, INRIA Rhone-Alpes, November 2005.

[41] W. Zhao, Y. Jiang, and C. Ngo. Keyframe retrieval by keypoints - can point-to-point matching help? In *5th Int'l Conf. on Image and Video Retrieval*, pages 72–81, 2006.