Compass-Based Automatic Picture Taking using SenseCam

Ignas Budvytis
Microsoft Research Cambridge/University of Cambridge

James Scott
Microsoft Research Cambridge

Alex Butler
Microsoft Research Cambridge

Abstract
SenseCam is a wearable digital camera that automatically takes pictures due to timers or sensor-based triggers. We propose using a compass sensor to trigger picture-taking, and detail our implementation of this idea as an add-on module for the SenseCam. We also describe a method for comparing triggering algorithms for SenseCam using a video recorder and video annotation software. We use this method to conduct a preliminary evaluation of compass-based triggering, and show a significant increase in the number of "independent views" captured.

1. Motivation

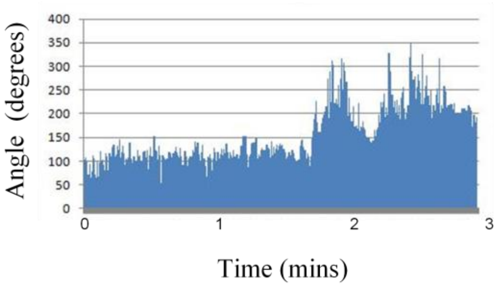
SenseCam [4] is a wearable camera designed to take pictures and log sensor data automatically throughout a user's day. It has a number of applications, in the treatment of memory loss due to brain injury [1] or degenerative diseases such as Alzheimer's, or in life-logging [3]. The built-in sensors include an accelerometer, passive infra red sensor, temperature sensor and a light sensor. In its default mode, in a single day it will take around 3000 photos.

In previous work with patients with memory impairment, one observation has been that it is not necessarily the "obvious" pictures that trigger recall of an event. To use an example from ongoing clinical work [2] in which a subject was taken to the cinema, it was surprising that images such as the cinema entrance cued recall less powerfully than a picture of the car park where a particular building happened to be in view, and a picture of a ticket machine in the cinema, since the building and ticket machine were particularly noticed by the subject at the time. It is difficult to see how physical sensors can predict when such surprising memorable views are present. (We did not try to use physiological sensors; this is an interesting direction for future work). Instead, we focus on the use of sensors to trigger capture of as many different "views" of the world as possible, in order to maximise the likelihood that a particular memorable object or scene is captured.

2. Compass-based triggering

We added a compass sensor (Honeywell HMR3300) to the SenseCam using an expansion port as shown in Figure 1(a). Figure 1(b) shows a typical compass trace. The compass provides a more direct "view change" trigger than the existing sensors in SenseCam, as a change in the compass often implies that the lens is facing new objects. However, simply triggering when the compass changes past a given threshold may result in repetitive pictures if the user is, for example, simply conversing with more than one person in a room and turning to face each in turn. We therefore explored more detailed sensing algorithms using historical data to predict when a particular direction faced was likely to contain a new "view".

(a) SenseCam with added compass sensor



(b) Example output from compass sensor

Figure 1: Adding a compass to SenseCam

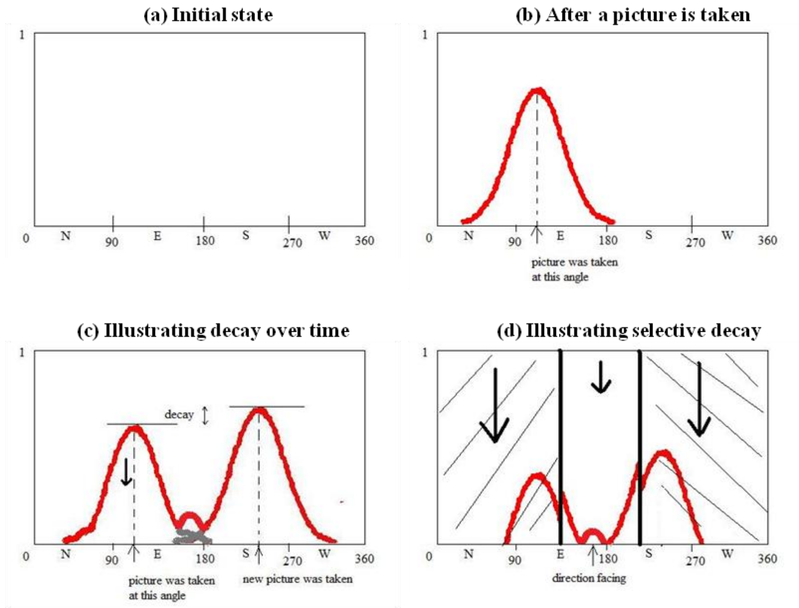


Figure 2: Compass-based triggering methods

2.1 Compass-Based Triggering Algorithms

In storing and using historical data, we were careful to design data structures and algorithms that can run on the SenseCam's limited processor (a PIC 18F8722) along with the other processing requirements of a SenseCam. Our basic data structure (see Figure 2(a)) is an array corresponding to the 360 degrees of compass directions, currently quantised into 1 degree chunks, where for each direction a value is stored signifying the level of recent picture-taking in each direction (represented as between 0 and 1, but actually stored as a byte). The total overhead of our algorithm is then 360 bytes (with the current size parameters), which is appropriate for the PIC implementation.

Figure 2(b) illustrates adding a set of values to the existing array if a picture is taken in a particular direction. The shape of the set of values can be determined by both the field of view of the lens and its shape (SenseCam uses a 119° fisheye lens so there are fewer pixels devoted to angles further off-centre), and the fact that pictures which have objects at their periphery may be regarded as badly capturing those objects. For these reasons we found it convenient to use a normal distribution as a "sensible default" here, and did not experiment further with different distributions.

The distribution is then used to judge when to take a new picture. We use a 10Hz timer to read the compass sensor and, using the array above, decide whether to take a picture or not (N.B. pictures can only be taken at a maximum rate of 0.2Hz). The decision is made simply on the basis of the level of the array in the current direction indicated by the compass sensor; if the level is less than half, a picture is taken and the array is modified. Of course, the values in the array must also be decreased over time or picture-taking would stop. This is illustrated in Figure 2(c). The decay is arranged so that if the camera is stationary, it would take pictures at a period of around 30 seconds, which is similar to the current SenseCam's picture-taking period while still.

A final modification was to use accelerometer readings as an indication of motion. Rather than directly triggering picture-taking from this as in the original SenseCam algorithm, we instead used this to decay the array faster during consistent movement. This is because if a user is moving, then the view in any particular compass direction changes so the fact that a picture was taken previously

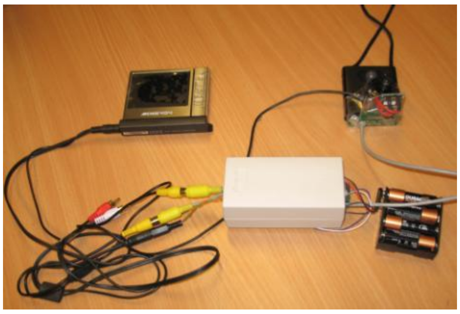


Figure 3: SenseCam with video logging hardware

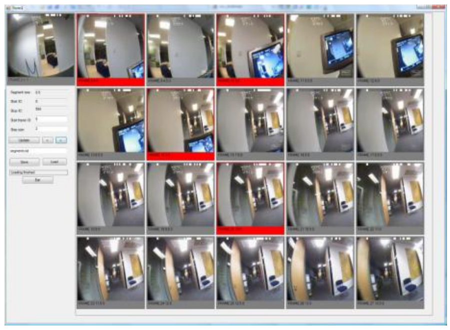


Figure 4: Software allowing users to annotate a video

in that direction does not preclude it being a new "view" which we wish to record. Furthermore, we implemented a direction-specific decay such that the directions to the left and right of the direction the user is facing decay even more quickly during movement, as illustrated in Figure 2(d). The rationale for this can be illustrated by the example of a user walking down a street and then pausing to briefly look in a shop window on their left. While the pause may only be for a few seconds, this compass-based algorithm will highly prioritise taking that picture.

3. Emulating SenseCam operation for repeatable experiments

In order to determine the usefulness of the above algorithm, we wanted to be able to conduct repeatable experiments comparing the algorithm with others. Unfortunately this is difficult to do using a SenseCam alone, since the SenseCam can only record pictures at a maximum rate of 0.2Hz (one each 5 seconds), and therefore cannot log pictures using multiple algorithms simultaneously.

Instead, we used the equipment shown in Figure 3, including a portable video recorder to gather video rather than pictures, with the same type of fisheye lens as the SenseCam, the lens being attached to the SenseCam itself at approximately the same position as the SenseCam's own lens. We configured the SenseCam to record sensor data only (no images) at a highly sped-up rate of 50Hz. To synchronise the video and sensors recorded, we built a custom device (the white box in Figure 3) which uses a timestamp signal from the SenseCam's expansion port to superimpose barcodes in the top region of the video signal sent to the video recorder.

In order to capture the ground truth of when "independent views" occurred, we also wrote a software tool (illustrated in Figure 4) which shows frames from the video in such a way that the user can quickly advance through the video and tag frames in which the scene differs enough that it is a "new view" compared to the previously tagged frame. The metric we asked the user to apply is that at least 50% of the view must be shared by every image that is included in the same "view". The software then presents the user with an opportunity to combine "views" that were close together in time, so that (e.g.) a sequence looking left, right, left, right ends up as two views.

4. Results

Using the video recorder, we gathered a data set comprising two hours in a normal office day including walking to a neighbouring building for lunch. We used this to test four algorithms:

- PERIODIC – simply taking a picture every 30 seconds
- CURRENT – the current SenseCam's algorithm, without a compass, triggering both periodically and also based on the accelerometer and passive infra-red sensor
- COMPASS – the compass-based algorithm described above.
- IDEAL – a manually constructed "ideal" trace as a comparison point only

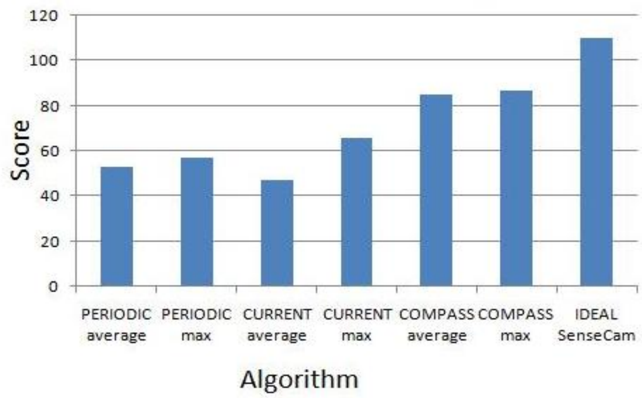


Figure 5: Comparison of picture-triggering methods

All the algorithms were tuned to limit picture-taking to an *average* of 1 picture every 30 seconds to make the total number of pictures taken even and the results comparable. The algorithms were scored by comparing their trigger times with the ground truth data, with 1 “point” per independent view that was captured at least once. Since the start time may make significant difference (particularly for PERIODIC), we also offset the start time by up to 30 seconds in 1 second intervals.

The results are shown in Figure 5. The COMPASS-based algorithm is the most robust to the time offset (max is similar to average), and the average score is 30% better than the best that the CURRENT algorithm can produce. The CURRENT algorithm actually performs worse than PERIODIC against our metric of independent views.

5. Conclusions

We have shown that the a compass can be a useful trigger for picture taking, particularly in the case of trying to maximise the number of independent “views” that are taken. While inspired by the needs of memory loss patients, this can also be valuable for the general user, particularly for life-logging applications e.g. to maximise the objects/scenes “logged”. Future work includes a more rigorous evaluation with more data sets across multiple users, and the evaluation of the compass with a normal SenseCam rather than in the emulation environment. The emulation environment, comprising the video logger hardware and the tagging software, are useful tools themselves that we also intend to deploy in other experiments, e.g. for sensor-based activity or location estimation.

6. References

[1] BERRY, E., KAPUR, N., WILLIAMS, L., HODGES, S., WATSON, P., SMYTH, G., SRINIVASAN, J., SMITH, R., WILSON, B. AND WOOD, K., "The use of a wearable camera, SenseCam, as a pictorial diary to improve autobiographical memory in a patient with limbic encephalitis". Special issue of Neuropsychological Rehabilitation 2007, 17 (4/5), 582-681, “Encephalitis: Assessment and Rehabilitation Across the Lifespan (Special Issue of Neuropsychological Rehabilitation)”. ISBN: 978-1-84169-836-6.

[2] BERRY, E. AND BROWNE, G. The role of SenseCam in memory rehabilitation . Presentation at the Workshop on Intelligent Systems for Assisted Cognition, New York, October 2007, htttp://research.microsoft.com/sendev/projects/sensecam/pdf_files/8%20Workshop%20on%20Intelligent%20Systems%20.pdf

[3] GEMMELL, J., WILLIAMS, L., WOOD, K., LUEDER, R. AND BELL, G., "Passive Capture and Ensuing Issues for a Personal Lifetime Store", Proceedings of the the 1st ACM workshop on continuous archival and retrieval of personal experiences. New York, USA October 2004.

[4] HODGES, S., WILLIAMS, L., BERRY, E., IZADI, S., SRINIVASAN, J., BUTLER, A., SMYTH, G., KAPUR, N. AND WOOD, K., “SenseCam: a Retrospective Memory Aid” In Proceedings of UbiComp 2006, LNCS 4206, 2006, Springer-Verlag.