A Non-Visual Sensor Triggered Life Logging System Using Canonical Correlation Analysis

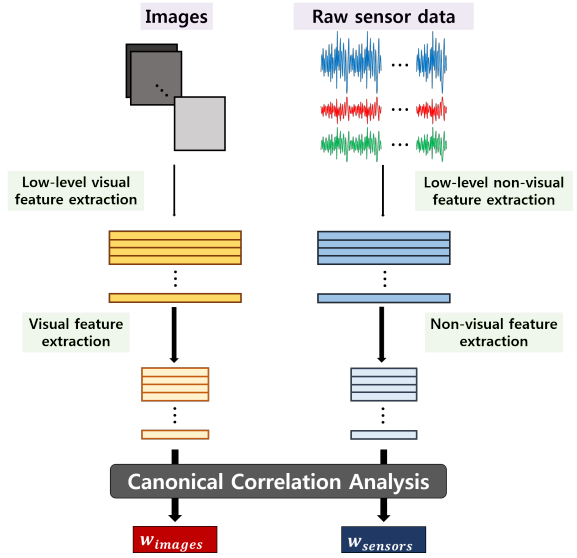
Inhwan Hwang and Songhwai Oh
CPSLAB, ASRI
Department of Electrical and Computer Engineering, Seoul National University, Seoul, Korea
Email: inhwan.hwang@cpslab.snu.ac.kr and songhwai@snu.ac.kr

***Abstract*—Life logging is one of the key service in modern life as wearable devices are forming an emerging market. Life logging has advantages to enlarge the human memory and even can help patients who suffer from memory impairment. Periodical picture taking is the simplest and the most widely used method but inefficient in both energy and memory side. In this paper, we suggest a novel capturing points decision method using the combination of visual information and non-visual information. In order to merge visual information and non-visual information, we adopted canonical correlation analysis (CCA) which is a statistic technique to find the mapping function for two different domain data into a highly correlate domain. In this way, we showed the new possibility of life logging system with minimum heuristic methodology. Moreover, we tested our approach on restricted real life situation and evaluated the result based on an image diversity measure using a determinantal point process (DPP) and showed better results than conventional methods.**

I. Introduction

As smartphones become the essential part of modern life, we can call this world as an era of smartphones. After the smartphones become key equipments for people today, people got interest in new types of devices such as wearable computers. Wearable computing is not a quite recent notion, only today it became more familiar with the help of smartphones and there are various types of wearable devices being invented and commercially sold nowadays.

Among them, a camera equipped type which makes it possible to log the day of the user is one of the most popular design of wearable devices after the advent of *Google Glass*. Life logging devices can help the user to memorize, summarize, and share her life. In the field of life logging, several researches are preceded to log the user's day effectively and efficiently. A *SenseCam* [5] from Microsoft Research is invented to help people who suffer from memory impairment such as the Alzheimer's disease. It provides not only periodic picture taking but also context based picture taking using information such as proximity, movement detected by an IR sensor, and temperature. Another life logging system, *InSense* [3], takes pictures based on the pre-defined interest and importance based on the user's activity and environmental characteristics. *Narrative Clip* [1] is one of the commercially produced life logging device which is comfortably small. Even though a number of life logging techniques are proposed, none of them relates the logging criteria to the characteristics of the image itself. The reason is that visual information itself holds the largest information, but at the same time it is impractical to use the visual information for a real-time capturing point decision due to its high computational cost for images processing.



(a) Weights learning part

(b) Points detecting part

Figure 1. Overview of the proposed method

In this paper, we propose a novel method which enables to log the day of the users according to the reasonable capturing criteria extracted from visual information. However, our proposed system does not make use of visual feature itself at points detecting part. Instead, we utilized the sensor input from the wearable devices and smartphones. Our proposed method is similar to *InSense* in the sense that the user's behavioral and environmental characteristics are used to find the capturing points but there is one distinct difference. While *InSense* merely relies on the user's experience to decide the current point is worthwhile taking pictures, our method utilizes the visual similarity extracted from the given two successive images from the stream of pictures taken with an egocentric camera designed for life logging. Although capturing points are extracted based on the image similarity, our proposed

method is not directly using the visual characteristics for deciding whether it takes pictures or not. Ideally, using visual characteristics to find capturing points, but it requires a large amount of system resources and computational cost. In this reason, our proposed method searches for capturing points on behavioral and environmental sides which have high relationship with visual characteristics. In order to relate these two different domain information, we applied canonical correlation analysis (CCA) [6] which enables to find projection vectors to a new domain which guarantees the maximum correlation between visual characteristics and non-visual characteristics. In this paper, we thoroughly explore the relationship between visual information and non-visual information and propose a possible method to relate these two different domain information.

As mentioned above, CCA is the key method to find the maximum correlation guaranteeing domain. Suppose two different domain information are expressed in the form of a vector as $(\mathbf{x}, \mathbf{y})$ and we are given a sample $S = ((\mathbf{x_1}, \mathbf{y_1}), \ldots, (\mathbf{x_n}, \mathbf{y_n}))$. Let S_x denotes $(\mathbf{x_1}, \ldots, \mathbf{x_n})$ and S_y denotes $(\mathbf{y_1}, \ldots, \mathbf{y_n})$. The objective of CCA is to find directional vectors $\mathbf{w_x}$ and $\mathbf{w_y}$ which maximizes the correlation between two projected vectors as follows:

$$\rho = \max_{\mathbf{w_x}, \mathbf{w_y}} corr(S_x \mathbf{w_x},\ S_y \mathbf{w_y}) \tag{1}$$

$$= \max_{\mathbf{w_x}, \mathbf{w_y}} \frac{< S_x \mathbf{w_x},\ S_y \mathbf{w_y} >}{||S_x \mathbf{w_x}||\ ||S_y \mathbf{w_y}||}. \tag{2}$$

Obtaining the projection vectors, $\mathbf{w_x}$ and $\mathbf{w_y}$, is done offline at a weights learning part so that the only things that we have to do at the points detecting part are simple linear projection and evaluating projected vectors in a new maximum correlation domain.

II. System Overview

Our proposed method is basically divided into two parts, a weights learning part and a capturing points detecting part. A weights learning part (Figure 1(a)) extracts weights for both images and sensors which associate the non-visual characteristics and visual characteristics using CCA. These weights are utilized to decide whether each point is worthwhile taking picture or not without using visual information at a point detecting part. For visual characteristics extraction, we adopt computer vision technique to calculate the similarity and flows of the visual elements between two successive images taken from a life logging device. On the other hand, non-visual characteristics are extracted from Wi-Fi fingerprints and an inertial measurement unit (IMU) sensor which is a combination of accelerometer, gyroscope, and magnetometer from the user carrying smartphone and wearable devices including a life logging device. Once the information extraction is successfully done on both sides, finding comparison domain is performed by applying CCA. A points detecting part (Figure 1(b)) finds capturing points merely relies on the non-visual behavioral and environmental information projected on the highly correlated domain obtained at a weights learning part. In this way, our method can decide whether taking picture or not without using visual information although the points are chosen reflecting visual importance.

III. System Explanation

Following subsections explain the detail of the process separated into two parts, the weights learning part and the capturing points detecting part. The weights learning part is again, separated into two phases, visual feature extraction and non-visual feature extraction.

A. Weights Learning Part

The main purpose of weights learning part is to extract projection vectors out of visual characteristics and non-visual characteristics embracing behavioral and environmental traits. In order to compute projection vectors, it is necessary to extract diverse visual characteristics out of low-level visual features such as SIFT (scale-invariant feature transform) feature [10] and GIST feature [14]. It is also crucial to extract non-visual features containing various type of user's behavioral and environmental information which can have high correlation with visual information at the same time.

1) Visual Feature Extraction: The visual feature extraction part computes low-level visual feature and then extracts high-level visual features, image similarity and visual flow.

In order to compute image similarity, we adopted the bag-of-words (BoW) model [4] which is widely used in the field of computer vision. Based on the BoW model, we generate a general codebook and extracting a histogram of codebook out of every picture. A histogram of a specific picture holds the scene information so that we can compare the scene similarity of given pictures by computing the distance among pictures. The detail of codebook generation is done in following order.

1) Collecting images with GoPro action camera
2) Extracting SIFT features out of each image
3) Quantizing the SIFT features in order to reduce the computational cost for clustering
4) Clustering given SIFT features using K-means algorithm [12]
5) Saving the center of each cluster as the codeword (the number of codewords is the size of codebook)

After the codebook generation, it is easy to extract a histogram out of a single image. Considering the number of collected images and the computational cost, we set the size of a codebook to be 1,000 which is a half of the well known appropriate codebook size for image processing. When generating a codebook, we quantized the SIFT features to 100,000 different subsets to help clustering the extracted features. In order to boost the performance of image comparison, we added color histogram obtained from Lab color space whose size is 600 ($= 200 \times 3$ (dimension of Lab color space)). By concatenating these two vectors and 960 dimensional GIST feature vectors which is commonly used to describe scene structure, a representative vector for each image can be generated. Before concatenating into one scene representative vector, we normalized each vector separately.

From the normalized representative image vectors, we can get the diversity measure out of every successive pair of images by calculating the distance between two representative vectors. When the distance is big, two different images are far from similar so that the similarity measure between two successive

State	Size	State
Wi-Fi	1	Number of access points
	1	Average Wi-Fi signal strength
	1	Percentage overlapped access points
Sensors	3	Average magnitudes of IMU from the smartphone
	6	Average magnitudes of IMU from the wearable devices
	3	Standard deviations of magnitude of IMU from the smartphone
	6	Standard deviations of magnitude of IMU from wearable devices
	3	Angular velocity of head in 3 axis (x, y, z)
Activity	4	User's current activity

Table I. STRUCTURE OF BEHAVIORAL AND ENVIRONMENTAL CHARACTERISTIC VECTOR

images, I_k and I_{k+1} can be expressed as the inverse of the diversity and it can be defined as follows:

$$similarity(I_k,\ I_{k+1}) = \frac{1}{||\mathbf{H_k} - \mathbf{H_{k+1}}||_\mathbf{2}}, \tag{3}$$

where $\mathbf{H_k}$ is a k^{th} image vector consists of a BoW histogram ($\mathbf{h_k^{BoW}}$), color histogram ($\mathbf{h_k^{color}}$), and GIST feature ($\mathbf{h_k^{GIST}}$), so that $\mathbf{H_k} = \{\mathbf{h_k^{BoW}},\ \mathbf{h_k^{color}},\ \mathbf{h_k^{GIST}}\}$.

After obtaining a diversity, it is necessary to generate visual characteristics in order to find the relationship between images and sensor readings. We set a visual characteristic vector (V) as the concatenation of image similarity and flow vectors of the visual elements so that $V_k = \{similarity_{k,\ k+1},\ \mathbf{flow_{k,\ k+1}}\}$. The first step to extract $\mathbf{flow_{k,\ k+1}}$ is calculating SIFT flow [9] out of two successive images (I_k and I_{k+1}). After SIFT flow is obtained, averaging SIFT flow vectors in a separated section. Each image is evenly divided into eight different sections so that there are eight different direction vectors out of two successive images. By calculating the mean value of eight different direction vectors, we can get the moving direction and amount of visual elements. And also, we can compute the consistency of two successive pictures by calculating the standard deviation of eight different direction vectors' magnitude and angle. If two images are consistent, the standard deviation of eight different direction vectors should be small.

2) Non-visual Feature Extraction: Non-visual features mainly focus on the traits of behavioral and environmental information like user's movements and locations. Behavioral characteristics are extracted from the IMU sensors and environmental characteristics are obtained from Wi-Fi fingerprint. Our proposed method assumes that the user is carrying a smartphone and wearing a wrist band type and a head mounted type smart devices equipped with IMU sensors at the same time so that we can track the user's movements and coarse locations.

For finding the relationship between behavioral characteristics from sensor based data and visual characteristics from image based data, it is the most important part to determine which factor of sensor data should be compared. The best scenario is that the user's body movement and the locational variation always influence the view of the user. However, the the specific sensor fluctuation does not always guarantee the transition of images at every moment. In this reason, it is necessary to thoroughly consider the factors to influence the visual changes of ego-centric life logging devices. After our elaborating examination on the relationship between logged pictures and the user's state, we drew a conclusion as follows:

1) When the user's state is consistent, the point of view usually remains unchanged.
2) User's behavioral characteristics affects the viewing perspective the most especially the rotational movement of the head.
3) User's activity can affect the sight of the user a lot.
4) User's locational change is one of the most crucial factor of logged pictures.

For these reasons, we categorized behavioral and environmental characteristic vectors into three different types according to the location and type of sensors as shown in Table I. First, Wi-Fi information offers the locational and environmental characteristics. Wi-Fi information basically provide the user's rough location. Furthermore, it even provides information whether the user is inside or outside of the building. Second, information from the IMU sensors reflects the user's behavioral traits such as the user is moving fast or slow, moving consistently or randomly, and with the help of head mounted type of wearable device, the detail movement of the user's head. IMU sensors sometimes provide not only the behavioral characteristics but also environmental characteristics when the user passes some area where abnormal magnetic field exist like near the elevator or huge electrical devices. Last, the classified user's activity out of smartphone is used to offer the user's behavioral characteristics. Our system simply classifies the user's activity into four different classes, staying, walking, taking a bus, and a subway. This classification result can also be used to obtain more sophisticated projection vectors specialized for a specific activity class.

Wi-Fi provides three different information, which is the number of Wi-Fi access points, Wi-Fi signal strength, and the ratio of overlapped access points. The number of Wi-Fi access points gives the hint about the location of the user. From the number of Wi-Fi access points, it is possible to infer whether the user is in the place which provides numerous Wi-Fi access points like public place or not. Wi-Fi signal strength can reflect the blockage degree of the user's current position. Even though two locations have similar Wi-Fi finger print distribution, Wi-Fi signal strength can separate these two locations. One with weak signal strength might have in condition that surrounded by several obstacles, which can affect the view of the life logging device. The last element of Wi-Fi characteristic vector is the ratio of overlapped access points. It is acquired simply store all mac addresses of access points at time k and compare them to the addresses at time $k+1$. This directly indicates the locational transition of the user.

Sensor type of characteristics provides the information about the user's activeness. Wearable devices and a user carrying smartphone are equipped with 3-axis accelerometer, gyroscope, and magnetometer so that they can analyze the detail movement of the user. A head mounted device manly focuses on the movement of the head which can indicate the viewing perspective of the user. Unlike a wrist worn type wearable device and a smartphone, a head mounted type of device collects not only the average and standard deviation of magnitude of IMU but also the angular velocity of a head in three axes (x, y, z). By doing so, it is possible to finely analyze the rotational movement of the user's head which gives great influence on the viewing perspective of the user.

Lastly, the classification result of the user's activity can

Domain	Feature	Formula
Time	Mean	$\mu_{X^T} = \frac{1}{l}\sum_{i=1}^{l} X_i^T$
	Variance	$\sigma_{X^T} = \frac{1}{l}\sum_{i=1}^{l}(X_i^T - \mu)^2$
	Minimum	$min_{X^T} = minimum(X_i^T)$
	Maximum	$max_{X^T} = maximum(X_i^T)$
	Range	$max_{X^T} - min_{X^T}$
	Mean Crossing Rate	$mcr_{X^T} = \frac{1}{l-1}\sum_{i=1}^{l-1}\delta\{(X_i^T - \mu)(X_{i+1}^T - \mu) < 0\}$ $(where\ \delta\ is\ the\ indicator\ function)$
	Root Mean Square	$rms_{X^T} = \sqrt{\frac{1}{l}\sum_{i=1}^{l} X_i^{T\,2}}$
	Skewness	$skew_{X^T} = \frac{1}{l}\sum_{i=1}^{l}(\frac{X_i^T - \mu}{\sigma})^3$
	Kurtosis	$kurt_{X^T} = \frac{1}{l}\sum_{i=1}^{l}(\frac{X_i^T - \mu}{\sigma})^4$
	Average Magnitude Area	$AMA_X = \frac{1}{l}\sum_{i=1}^{l}(\lvert X_i^1\rvert + \lvert X_i^2\rvert + \lvert X_i^3\rvert)$
	Average Energy Expenditure	$AEE_X = \frac{1}{l}\sum_{i=1}^{l}\sqrt{X_i^{1\,2} + X_i^{2\,2} + X_i^{3\,2}}$
Frequency	Minimum Frequency	$\omega_{min} = \{\omega \lvert X_i^F(\omega) = minimum(X_i^F)\}$
	Maximum Frequency	$\omega_{max} = \{\omega \lvert X_i^F(\omega) = maximum(X_i^F)\}$
	Spectral Energy	$E_{X^F} = \frac{1}{l}\sum_{i=1}^{l} X_i^F$

Table II. FEATURES USED FOR ACTIVITY RECOGNITION

affect the properties of the user's viewing perspective. Activities are classified into four different types, staying, walking, taking a bus, and a subway. Assuming that a user is in a subway, then the scene around the user is relatively more consistent than walking around in outdoor circumstance. This means that the user's activity can be the prior knowledge for the user's viewing perspective and it can provide a room to flexibly modify the criteria to decide whether given situation is worthwhile taking a picture.

Activity recognition is performed using the data from accelerometer, gyroscope, and magnetometer from user's smartphone. Features for classification are shown in Table II which are commonly used for activity recognition using IMU sensors. We utilized existing features from [11] and [13], and chose the minimum number of features to classify only four activities efficiently and confidently. In order to minimize the affect of noise, we adopted L_1-ARG algorithm proposed in [8]. It is not only robust against noisy sensory input but also gives a soft classification result so that the classification scores can be weight factors to combine multiple CCA projection weight vectors obtained individually out of different activities.

Above mentioned behavioral and environmental characteristic vectors can be both continuous and discrete, so that it is hard to be integrated into one system. It is more reasonable to be integrated into continuous value because all the elements of the behavioral and environmental characteristic vector is continuous except activity. For scale consistency of characteristic vector, we set all the value to have a value between zero to one and the activity class is encoded into four digits binary code whose each single entry is one or zero. In order to normalize the continuous values, we fit them into normal distribution so that the values of the vector's elements are between zero and one.

3) Weights Learning: When both visual and non-visual characteristic is extracted, weights can be calculated using CCA. CCA finds a domain to maximize the correlation between visual characteristic vectors ($\mathbf{V}$) and behavioral and environmental characteristic vectors ($\mathbf{S}$) and calculates projection vectors for visual characteristic vectors and behavioral and environmental characteristic vectors ($\mathbf{w_V}$, $\mathbf{w_S}$). After these two projection vectors are obtained, both the visual vectors and non-visual vectors can be projected to an identical domain whose correlation between two characteristic vectors, $\mathbf{V}$ and $\mathbf{S}$, is maximized. As a result, reference vectors which are the projected visual characteristic vectors at capturing points are expressed as $\mathbf{V^{new}_{picPoint}} = \mathbf{w_V} \cdot \mathbf{V_{picPoint}}$ on maximum correlation guaranteeing domain where $\mathbf{V_{picPoint}}$ is obtained according to the image similarity (equation 3).

B. Points Detecting Part

Unlike the weights learning part, this part is performed on a user carrying device in real-time. When the sensor data came, the first thing should be done is extracting behavioral and environmental characteristic vectors. Then, extracted characteristic vectors are projected to the new domain which maximizes correlation between two vectors by simply multiplying them with the weight vector obtained in the weights learning part. Consequently, the projected non-visual vectors are expressed as $\mathbf{S^{new}_{test}} = \mathbf{w_S} \cdot \mathbf{S_{test}}$. Projected characteristic vectors, $\mathbf{S^{new}_{test}}$, are then compared to the projected visual characteristic vectors at capturing points, $\mathbf{V^{new}_{picPoint}}$, by simply doing inner product two vectors.

Figure 2 shows the the relationship between two different characteristic vectors at capturing points, $\mathbf{V_{picPoint}}$ and $\mathbf{S_{picPoint}}$. Figure 2(a) shows the relationship between two different types of vectors after principal component analysis (PCA) [7], which is one of the most popular dimension reduction method is applied. PCA basically finds the axis which maximizes the variance within data so that it does not consider the relationship between two different types of data, visual and behavioral and environmental vectors in our case. On the other hand, Figure 2(b) are 1-D projected vectors using CCA which considers the relationship between two different types of vectors. In this reason, Figure 2(b) shows better correlated result than Figure 2(a) which does not consider the relationship between two different domain features. Figure 2(c) shows the distance histogram in each projected space. The distance in CCA space is rather concentrated at near zero, while the distance in PCA space is relatively evenly distributed. This indicates that the CCA provides better projection vectors to connect the visual and non-visual information at each point.

From the result here, we set the threshold whether input projected characteristic vectors are similar to the capturing points or not. When visual and behavioral and environmental characteristic vectors at capturing points are projected into one identical domain, then the inner product value of two vectors can be interpreted as the similarity criteria. If the inner product value with the i^{th} capturing point is within the similarity bound, which means the inner product value is less than threshold, then the given input point i can be selected as a capturing point as follows:

$$\delta(i,\ k) = \begin{cases} 1 & \text{if } \mathbf{V^{new}_{picPoint}} \cdot \mathbf{S^{new}_{test}} \leq \sigma \\ 0 & \text{othrewise} \end{cases} \quad (4)$$

where σ is the threshold.

Total similarity score can be computed by summing up all the δ obtained from all visual based capturing points as follows:

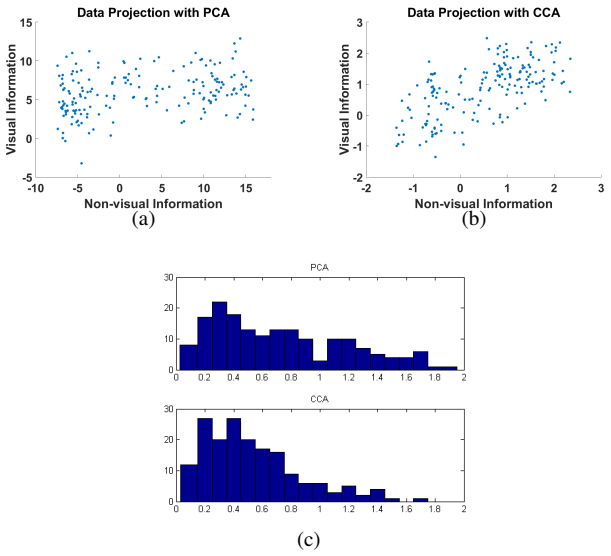


Figure 2. (a) Relationship between original visual information vector and behavioral and environmental information vector projected on 1-dimensional space with PCA. (b) Data projection with CCA and the relationship between visual information vector and behavioral and environmental information vector. (c) Histogram of distances between visual information and non-visual information at each point.

$$score(k) = \sum_{picPoints} \delta(i,\ k). \tag{5}$$

Deciding capturing points is performed based on the scores obtained above. There are a number of the projected visual characteristic vectors at capturing points which are obtained in weights learning part and it is reasonable to compare all of these reference points when the new test input comes.

IV. Diversity

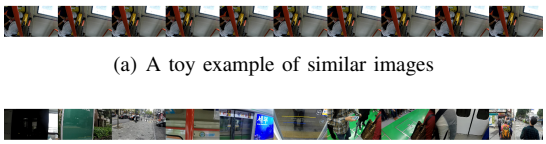
In this paper, we propose two different types of diversity measure of image stream to evaluate the performance of our proposed method. Diversity can be easily considered as the inverse of similarity measure introduced in Section III-A1.

The evaluation criteria is total diversity of the image stream, and it is computed in two different ways. First criteria is measuring the pairwise diversity of all images in a given stream. Second criteria is using determinantal point process (DPP) [2] to measure diversity. When DPP is applied, diversity is obtained by simply calculating the determinant of generated kernel matrix based on the image features. In our proposed method, there are three different types of histogram, $\mathbf{h_k^{BoW}}$, $\mathbf{h_k^{color}}$, and $\mathbf{h_k^{GIST}}$ to measure similarity and diversity between images. The kernel matrix is defined as follows:

$$L_{i,j} = exp\left(-\sum_{feat} \frac{||\mathbf{h_i^{feat}} - \mathbf{h_j^{feat}}||_2^2}{\sigma_{feat}}\right), \tag{6}$$

where i and j are the index of all images in the image stream and $feat \in \{BoW,\ color,\ GIST\}$.

To verify proposed two different types of diversity measure, we examined it on simple toy example as shown in Figure 3.



(a) A toy example of similar images

(b) A toy example of diverse images

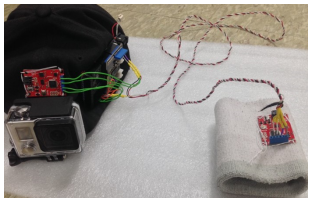
Figure 3. Example of two different image stream

The diversity measured simply calculating pairwise distance is 8.099×10^{-2} for a similar image set (Figure 3(a)) and 1.457×10^{-1} for a diverse image set (Figure 3(b)). The diversity results using DPP are 7.666×10^{-21} for a similar image set (Figure 3(a)) and 8.505×10^{-17} for a diverse image set (Figure 3(b)).

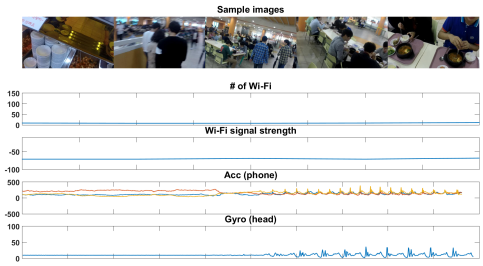
V. Experimental Results

We use the data collected with sensor and Wi-Fi data from Google Nexus 4 and 9-DoF IMU data out of Arduino which is installed on the front of a hat and a wrist as shown in Figure 4(a). The sensor mounted on the front of a hat can detect the movement of the head and also can infer the direction of the user's view.

For training purpose, we need pictures with good resolution and fast sampling rate (at least one picture per one second) so that we set up a GoPro action camera on the forehead with a head-mounted sensor. A GoPro action camera collects visual data as a video clip instead of still pictures and extracted pictures out of video clips afterwards.



(a) A prototype of a life logging device



(b) Having lunch in a campus cafeteria

Figure 4. A prototype and sample data out of a prototype device

We gathered data with above hardware setting for seven days in normal daily life scenarios. Figure 4(b) shows examples of gathered images and sensor readings. Data gathering is performed by one subject in order to minimize the effect of unique pattern from a specific user's daily activity. Total length of collected data is about eight hours of data so that about 29,247 pictures (one picture per one second) are gathered. We separated the set into training set and test set according to the

(a) Non-visual information based point detection

(b) Periodic image selection method

Figure 5. The resulting images on going to campus cafeteria for lunch scenario. Two different types of capturing points selecting methods are examined. The time sequence of resulting images is from left to the right and from top to the bottom.

Type of scenario	Visual information	Non-visual information	Periodic
Commuting 1	2.230×10^{-39}	8.061×10^{-58}	9.152×10^{-63}
Commuting 2	1.245×10^{-2}	3.163×10^{-8}	3.051×10^{-9}
Lunch 1	3.192×10^{-2}	4.147×10^{-9}	4.957×10^{-10}
Lunch 2	7.093×10^{-10}	3.918×10^{-20}	1.545×10^{-19}

Table III. DIVERSITY OF EACH IMAGE STREAM

day of data is collected, i.e., if the first day is set to be the test set then the rest of the days are designated to be a training set. We set the capturing points to be 10 % of the total images and run the proposed algorithm on four scenarios to select the capturing points out of collected data based on three different methods, using visual information, non-visual information (proposed method), and periodic picturing selecting. Each method selects same number of images in order to compare the diversity in fair condition. We evaluated the resulting image stream by calculating the diversity measure we proposed in Section IV. Table III shows the diversity of four different scenarios out of seven days. Except the scenario Lunch 2, non-visual information based image capturing shows higher diversity than periodic image capturing. Figure 5 shows one of the resulting image stream (Figure 5(a)) and comparison group (Figure 5(b)) of scenario for going to lunch at campus cafeteria (Lunch 1). From the resulting images, we can conclude that the capturing points detection using sensor input reasonably selects images with little overlap. As shown in Figure 5, a sensor based points detection method (Figure 5(a)) selects less overlapped images than a periodic image selection method (Figure 5(b)) when having a lunch in a campus cafeteria.

VI. FURTHER WORK

Even though we showed the possibility of efficient life logging system without using visual information, there are two major works should be followed. First, the experiment is done in relatively restricted condition so that there still have some questions on how reliably this system works in real scenarios. So, the first issue on our proposed method should be focused on sufficient data collecting and stable performance on various types of daily activity regardless of the users. Moreover, the sensors and a camera used for data collection and evaluation is just a prototype so that it is hard to be used in real life. Designing rather complete life logging device with IMU will be the next thing should be done.

VII. CONCLUSION

In this paper, we proposed a novel life logging method using non-visual information with the help of visual information. Even though our method is basically defining the capturing points based on the visual information, real time capturing points detection is possible because there are no image processing is performed. And also we proposed a measure for image diversity so that it can be used to evaluate the quality of resulting image stream. Unlike other life logging method, we showed the new possibility of life logging system with minimum heuristic methodology.

ACKNOWLEDGMENTS

This work was supported by the Institute for Information and Communications Technology Promotion (IITP) grant funded by the Korea government (MSIP) (B0101-15-0557, Resilient Cyber-Physical Systems Research).

REFERENCES

[1] http://getnarrative.com.

[2] R. H. Affandi, E. B. Fox, R. P. Adams, and B. Taskar. Learning the parameters of determinantal point process kernels. *arXiv preprint arXiv:1402.4862*, 2014.

[3] M. Blum, A. S. Pentland, and G. Troster. Insense: Interest-based life logging. *IEEE Multimedia*, 13(4):40–48, 2006.

[4] L. Fei-Fei, R. Fergus, and A. Torralba. Recognizing and learning object categories. *CVPR Short Course*, 2, 2007.

[5] S. Hodges, E. Berry, and K. Wood. Sensecam: A wearable camera that stimulates and rehabilitates autobiographical memory. *Memory*, 19(7):685–696, 2011.

[6] H. Hotelling. Relations between two sets of variates. *Biometrika*, pages 321–377, 1936.

[7] I. Jolliffe. *Principal component analysis*. Wiley Online Library, 2002.

[8] E. Kim, M. Lee, C.-H. Choi, N. Kwak, and S. Oh. Efficient l_1-norm-based low-rank matrix approximations for large-scale problems using alternating rectified gradient method. *IEEE Trans. on Neural Networks and Learning Systems*, 26(2):237–251, 2015.

[9] C. Liu, J. Yuen, and A. Torralba. Sift flow: Dense correspondence across scenes and its applications. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(5):978–994, 2011.

[10] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[11] H. Lu, J. Yang, Z. Liu, N. D. Lane, T. Choudhury, and A. T. Campbell. The jigsaw continuous sensing engine for mobile phone applications. In *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, pages 71–84. ACM, 2010.

[12] J. MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, page 14. California, USA, 1967.

[13] C. McCall, K. K. Reddy, and M. Shah. Macro-class selection for hierarchical k-nn classification of inertial sensor data. In *PECCS*, pages 106–114, 2012.

[14] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International journal of computer vision*, 42(3):145–175, 2001.