

AI Goggles: Real-time Description and Retrieval in the Real World with Online Learning

Hideki Nakayama Tatsuya Harada Yasuo Kuniyoshi
 Dept. of Mechano-Informatics, Grad. School of Information Science and Technology,
 The University of Tokyo, 7-3-1, Hongo, Bunkyo-ku, Tokyo, Japan
 {nakayama, harada, kuniyosh}@isi.imi.i.u-tokyo.ac.jp

Abstract

In this paper, we present the AI Goggles system, which can instantly describe objects and scenes in the real world and retrieve visual memories about them using keywords input by the users. This is a stand-alone wearable system working on a tiny mobile computer. Also, the system can quickly learn unknown objects and scenes by teaching and learn to label and retrieve them on site, without loss of recognition ability for previously learnt ones.

As the core algorithm of the system, we propose and implement a new method of multi labeling and retrieval of unconstrained real-world images. Our method outperforms the current state-of-the-art method, in terms of both accuracy and computation speed on the standard benchmark dataset. This is a major contribution to development of visual and memory assistive man-machine user interface.

1. Introduction

Recently, an increasing number of studies have been conducted on “life logs.” This research field aims to constantly record and analyze real-world information (typically visual information) that we observe in daily life autonomously [6]. The realization of such a system is an extremely important challenge, not only from a scientific standpoint concerned with human behavior, but also from a practical one. It would have widespread applications, such as memory assistance/organizer and vision aid (visual dictionary).

In order to realize such a system, it is rational to extract exactly the same images of user’s view and record them (Fig. 1) because our eyesight reflects our intention. Also, it is appropriate to mount the system on a daily-used item such as glasses or goggles because these systems are expected to be implemented into a wearable system and natural enough not to disturb daily life. Considering the current improvements in computers and hardware, it is not difficult

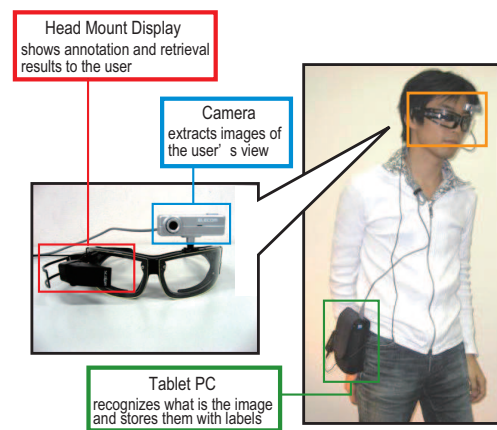


Figure 1. Implementation of the AI Goggles system.

to assume that the whole system will be embedded in normal glasses in the future.

However, a mere image log is awkward and difficult for end users to handle effectively. In order to recall some information efficiently, it is desirable that images are tagged with some labels so that the user can retrieve necessary information by using the labels, as in an internet search query. In other words, the system needs to automatically recognize the semantic content of the images and store them with appropriate symbols. This is an extremely difficult problem due to the ambiguity of real world images, and the complex process used by humans to assign symbols to them.

Humans have a powerful ability to near-instantaneously recognize very complex scenes and objects in the visual field. We act in a range of differing situations and need to recognize an enormous number of generic objects. In addition to identifying names of objects and places (noun), it is also necessary to obtain descriptive and impressionistic (adjective) information about images. In order to realize this

ambiguous processing on computers, we need a highly versatile and sophisticated algorithm of generic object recognition, which can extract multiple meanings from images. Furthermore, because recognition is a subjective process, different people may provide very different labelings of the same scene. Therefore, desirable symbolization depends on each user's life and is not apparent. Thus the labeling policy of the system should be able to adapt to each user. Also, we cannot implement all necessary knowledge to the system in advance because there are an infinite number of objects and scenes in the real world. For these reasons, online learning function is extremely important for user-adaptive systems.

Today, numerous studies are under way to realize automatic image recognition. Probably the most promising framework for generic object recognition with multi labeling is that of the image annotation and retrieval research field [11, 5, 4, 2], which is based on a statistical machine learning framework. Its goal is to attach multiple labels to unknown input images (annotation), and to search for unlabeled images that best correspond to text queries input by user (retrieval). This framework allows images to be multiply labeled, with any set of symbols desired by the users. In this sense, this scheme best meets the need of our system.

However, previous methods in image annotation field are not sufficiently accurate, and furthermore, they need tremendous amounts of computational resources (e.g., SML [2], which is at the top on Corel image dataset [4] in 2007, needs a cluster of 3,000 PCs for computation). For the applications described in this paper, instantaneous recognition and retrieval is crucial because the system must deal with an environment that varies from hour to hour in real time. This requirement is difficult to meet, as the computational resources of wearable systems are often strictly limited. Also, previous methods do not consider the case where additional symbols are added to the system after the initial training period. Therefore, it has so far been effectively impossible to apply previous methods of image annotation and retrieval to the real-world applications mentioned above.

2. AI Goggles

In this paper, we present the AI Goggles system, which is a wearable system capable of describing generic objects in the environment and retrieving the memories of them using visual information in real time without any external computation resources. Our system is also capable of learning new objects or scenes taught by users. This ability further enhances man-machine interaction and makes our system practical. As the core of the system, we develop a high-accuracy and high-speed image annotation and retrieval method supporting online learning. We show that our method is superior to previous methods in quantitative evaluation experiments.

Figure 1 shows the implementation of the AI Goggles system. This system is composed of a web camera mounted on the goggles, a head mount display (HMD), and a tablet PC (CPU: Core2Duo 1.2 GHz; Memory: 2 GB). C++ language is used for implementation. Figure 2 shows the overall view of the system. The system continuously extracts an image of the user's view from the web camera, annotates the image using the proposed annotation method, and then displays the annotation result on the HMD. At the same time, the system accumulates the image and corresponding annotation result to a vision log. When it receives a search query from the user, it finds appropriate movies in the vision log and displays them on the HMD. In addition, it can incrementally learn objects and scenes taught by the user on sight. Figure 3 shows the GUI of the system. A large image at top-left is the camera image (user's view), and red texts in center are the corresponding annotation. Five small images at the bottom are thumbnails of retrieved images from memory, corresponding to the query showed above. When the user clicks them, the system shows a sequence of images around the time they were recorded.

We review some closely-related previous studies based on generic image recognition techniques. Schiele *et al.* [13] had proposed a similar memory-assistive system using visual triggers. They built a museum's guide as a practical application. However, their interface is built on *query-by-example* scheme and does not provide keyword search. Also, it has been argued that it is difficult to overcome semantic-gap [14] by *query-by-example* scheme. For these reasons, it can be said that semantic aspects of their system are not enough.

Torralla *et al.* [15] had proposed a wearable system that performs versatile objects and scene recognition. Their system can recognize scenes such as "room," "corridor," "road," and representative objects from the scenes. Nevertheless, because they need to train a large-scale graphical model, it is expected that it will face practical issues when the number of target objects increases; for example, computational time for learning will be excessively large. In addition, their system does not support online learning.

In contrast, our method is extremely light, in terms of both learning and recognition. Our system can recognize a wide range of generic objects in real time and enables re-experiencing of the memory through our image annotation and retrieval method. Besides, it supports online learning of unknown objects, which is a crucially important function for a practical application.

3. Proposed Image Annotation and Retrieval Method

In this section, we describe our method of image annotation and retrieval [12], which is the core of our system.

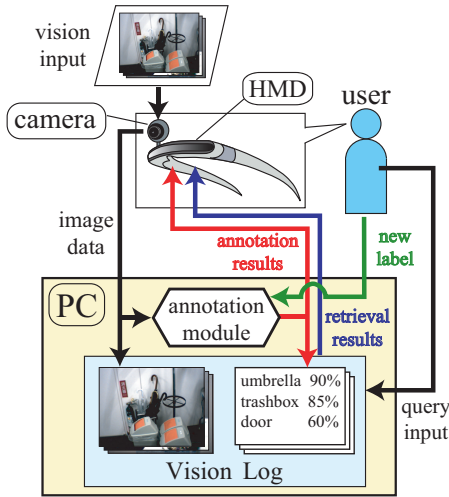


Figure 2. Overall view of the AI Goggles system.

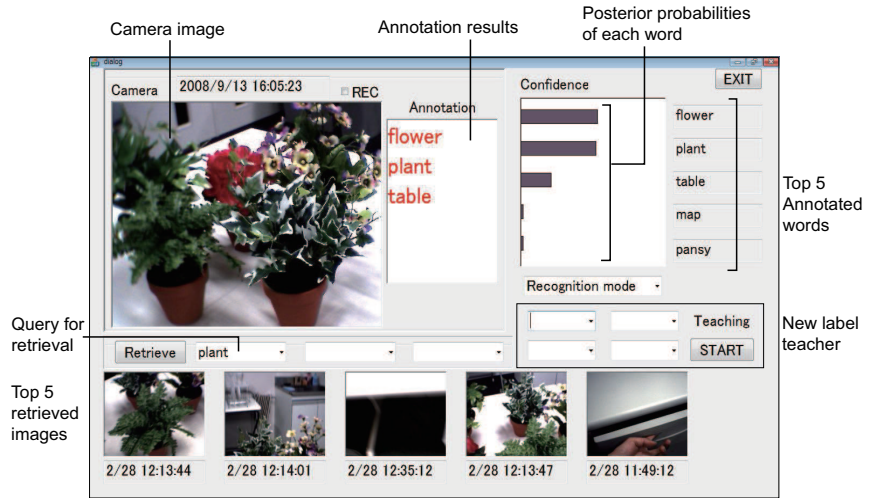


Figure 3. Graphical User Interface of the AI Goggles system.

3.1. Approaches

In previous studies on image annotation and retrieval, there have been mainly two approaches. Supervised Multiclass Labeling (SML) [2], which achieves the best performance on the Corel dataset [4] as of 2007, estimates the relevance of image and word class directly as Fig. 4 (a). However, because the image feature distribution of a certain word has a complicated structure in general, their probability density functions become also highly complex. For this reason, it takes a lot of time to learn the model. Also, it is difficult to control generalization.

On the other hand, another approach has been studied which assumes an intermediate latent node between images and words as Fig. 4 (b). This node expresses the essential hidden state responsible for generating both the image and the words. It has been shown that this model can perform learning and annotation efficiently. Translation-model [4], CRM [11], and MBRM [5] are typical works. However, the problem is that these methods basically estimate the latent space only from image features and ignore the semantic aspects.

In this study, we use Canonical Correlation Analysis (CCA) to learn the latent variable. Although there is some previous work using CCA for image annotation [7], mere use of CCA loses the information of non-linear distribution in the latent space because CCA is basically a linear approximation technique and does not provide a probabilistic scheme. Our proposed method can efficiently exploit the non-linear structure by sample-based approach with a probabilistic basis [1]. One of the other approaches to exploit the non-linearity is the use of kernelised methods [7, 8]. How-

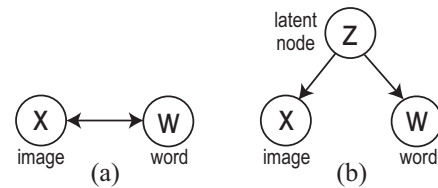


Figure 4. Two approaches to the annotation/retrieval problem.

ever, scalability of kernelised methods are barely tractable because they need to solve eigenvalue problems whose dimensions are the number of training samples. Also, it is difficult to control generalization.

3.2. Latent Variable Learning between Image and Labels via CCA

Here, we have p dimensional image features $\mathbf{x} = [x_1, x_2, \dots, x_p]^T$ and q dimensional label features $\mathbf{w} = [w_1, w_2, \dots, w_q]^T$. We describe the training data as $\{(\mathbf{x}_i, \mathbf{w}_i) | i = 1, \dots, N\}$ and the covariance matrix as $C = \begin{pmatrix} C_{xx} & C_{xw} \\ C_{wx} & C_{ww} \end{pmatrix}$. CCA finds the linear transformation $\mathbf{s}_i = A^T \mathbf{x}_i$, $\mathbf{t}_i = B^T \mathbf{w}_i$ to maximize the correlation between the new values (canonical values), \mathbf{s}_i and \mathbf{t}_i . Optimal projection matrices A and B can be obtained explicitly as the solution of the following generalized eigenvalue problems.

$$C_{xw} C_{ww}^{-1} C_{wx} A = C_{xx} A \Lambda^2 \quad (A^T C_{xx} A = I_d), \quad (1)$$

$$C_{wx} C_{xx}^{-1} C_{xw} B = C_{ww} B \Lambda^2 \quad (B^T C_{ww} B = I_d), \quad (2)$$

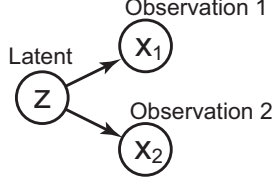


Figure 5. Graphical model for canonical correlation analysis.

Here, Λ^2 is a diagonal matrix of eigenvalues. The square root of the eigenvalues is the correlation between the canonical values \mathbf{s} and \mathbf{t} . d is the dimension of \mathbf{s} and \mathbf{t} . We can obtain d eigenvectors in the descending order of the eigenvalues.

According to probabilistic CCA [1] viewpoints, the graphical model for CCA has a structure shown in Fig. 5; note the similarity to Fig. 4(b). \mathbf{z} in this figure represents a latent variable, and is intimately related to \mathbf{s} and \mathbf{t} . The relationship between \mathbf{s} , \mathbf{t} and the latent variable \mathbf{z} can be obtained as follows using the posterior expectation of \mathbf{z} on the probabilistic CCA: $E(\mathbf{z}|\mathbf{x}) = G_x^T \mathbf{s}$, $E(\mathbf{z}|\mathbf{w}) = G_w^T \mathbf{t}$, where G_x, G_w are arbitrary matrices such that $G_x G_w^T = \Lambda$. Here, we select an identity matrix as G_x . That is, we take \mathbf{s} as the latent variable \mathbf{z} .

The optimal annotation \mathbf{w} for an image \mathbf{x} is the one that maximizes posterior probability $p(\mathbf{w}|\mathbf{x})$. By using $\{\mathbf{s}\}_{i=1}^N$ as the latent variable, the posterior probability can be marginalized as follows;

$$\begin{aligned} p(\mathbf{w}|\mathbf{x}) &= \frac{\sum_{i=1}^N p(\mathbf{x}|\mathbf{w}, \mathbf{s}_i) p(\mathbf{w}|\mathbf{s}_i) p(\mathbf{s}_i)}{\sum_{i=1}^N p(\mathbf{x}|\mathbf{s}_i) p(\mathbf{s}_i)}, \\ &= \frac{\sum_{i=1}^N p(\mathbf{x}|\mathbf{s}_i) p(\mathbf{w}|\mathbf{s}_i)}{\sum_{i=1}^N p(\mathbf{x}|\mathbf{s}_i)}. \end{aligned} \quad (3)$$

The simplification $p(\mathbf{x}|\mathbf{w}, \mathbf{s}_i) = p(\mathbf{x}|\mathbf{s}_i)$ can be obtained using the assumption of the conditional independence. In the absence of any task knowledge we use a uniform prior $p(\mathbf{s}_i) = 1/N$, so we obtain Eq. (3).

Given a new image \mathbf{x}_{new} , we calculate the posterior probabilities $p(w_i|\mathbf{x}_{new})$ of all candidate words $\{w_i\}_{i=1}^q$. We then annotate the new image \mathbf{x}_{new} as w_i in descending order of these probabilities.

In retrieval, we use the maximum likelihood estimation. We let $\{I_i\}$ denote the candidate images and $\{\mathbf{x}_i\}$ their image features. Given a query \mathbf{w}_{new} as an input, we calculate the likelihood $p(\mathbf{w}_{new}|\mathbf{x}_i)$ of all I_i according to Eq. (3), and then output the candidate images in descending order of likelihood. In this way, we can obtain a ranked retrieval of the query.

3.3. Density Functions

The density function of posterior probability $p(\mathbf{x}|\mathbf{s}_i)$ in Eq. (3) can be calculated using the distance in canonical space. We denote $\mathbf{s} = A^T \mathbf{x}$ as the projected point in canonical space of the image \mathbf{x} . We define the posterior probability as

$$p(\mathbf{x}|\mathbf{s}_i) = \frac{\exp\left(-\frac{1}{2}(\mathbf{s} - \mathbf{s}_i)^T \Sigma^{-1}(\mathbf{s} - \mathbf{s}_i)\right)}{\sqrt{2\pi}^d \sqrt{|\Sigma|}}. \quad (4)$$

Here, $\Sigma = \beta I_d$. We can control the smoothness of the posterior density distribution by setting the band width β appropriately. Moreover, compared to the image-features space, the dimension of canonical space is highly compressed. Thus, we can reduce computational costs considerably.

We design $p(\mathbf{w}|\mathbf{s}_i)$ in a top-down manner using language models. In previous research, CRM [11] and MBRM [5] have used such language models. Here, we use a model that builds on these previous models.

$$p(\mathbf{w}|\mathbf{s}_i) = \prod_{w \in \mathbf{w}} p_W(w|\mathbf{s}_i), \quad (5)$$

$$p_W(w|\mathbf{s}_i) = \mu \delta_{w, \mathbf{s}_i} + (1 - \mu) \frac{N_w}{N_W}, \quad (6)$$

where, scalar w denote each element of \mathbf{w} . Also, N_W is the total number of labels in the training data set, N_w is the number of the images that contain w in the training data set, δ_{w, \mathbf{s}_i} is one if the label w is annotated in the image \mathbf{s}_i otherwise zero. μ is a parameter between zero and one.

3.4. Online Learning

Here we describe the algorithm of incremental learning of new training samples. The core of our algorithm is CCA, so the main problem is how to derive incremental CCA.

One obvious method is to use the classical perceptron algorithm [10]. However, to get discriminative power in this approach, we need to sequentially input all target objects into the system evenly. In a practical situation it is natural to give the system a large batch of samples of a certain object all at once, so this assumption is invalid. Also, it is quite difficult to use this approach if the dimension of the feature vector increases. Thus, it is almost impossible to learn a new label.

Therefore, we implement a simple form of incremental CCA that incrementally estimates only covariance matrices. This method needs to solve an eigenvalue problem every time. However, unlike kernelised methods [7, 8], the dimension of our eigenvalue problem is that of image and labels features. Therefore, computational cost of solving the

eigenvalue problem is constant against the number of samples and is generally quite smaller than the cost of calculating covariance matrices (see Sec. 5.2). Also, this approach can deal with the case in which the dimension of features increases.

Suppose we already have t training samples, and mean vector, correlation matrices, covariance matrices of them. We let $\mathbf{m}(t), R(t), C(t)$ denote them respectively. When a new training sample $\{\mathbf{x}_{t+1}, \mathbf{w}_{t+1}\}$ is given, we update mean and covariance as follows.

$$\begin{aligned} \mathbf{m}_x(t+1) &= \frac{t-l}{t+1}\mathbf{m}_x(t) + \frac{1+l}{t+1}\mathbf{x}_{t+1}, \\ R_{xx}(t+1) &= \frac{t-l}{t+1}R_{xx}(t) + \frac{1+l}{t+1}\mathbf{x}_{t+1}\mathbf{x}_{t+1}^T, \\ C_{xx}(t+1) &= R_{xx}(t+1) - \mathbf{m}_x(t+1)\mathbf{m}_x^T(t+1), \end{aligned} \quad (7)$$

where l is a positive number that determines the weight of a new sample. We update C_{ww} and C_{xw} likewise. When $l = 0$, the resultant covariance matrices and CCA solution become exactly the same ones obtained from batch process.

When a new label is added, the update algorithm slightly changes as follows. A new label feature corresponding to the new label is added to the tail of original label features vector.

$$\begin{aligned} \mathbf{m}_w(t+1) &= \frac{t-l}{t+1} \begin{pmatrix} \mathbf{m}_w(t) \\ 0 \end{pmatrix} + \frac{1+l}{t+1}\mathbf{w}_{t+1}, \\ R_{ww}(t+1) &= \frac{t-l}{t+1} \begin{pmatrix} R_{ww}(t) & \mathbf{0} \\ \mathbf{0} & 0 \end{pmatrix} + \frac{1+l}{t+1}\mathbf{w}_{t+1}\mathbf{w}_{t+1}^T, \\ R_{xw}(t+1) &= \frac{t-l}{t+1} (R_{xw}(t) \mathbf{0}) + \frac{1+l}{t+1}\mathbf{x}_{t+1}\mathbf{w}_{t+1}^T. \end{aligned} \quad (8)$$

After updating covariance matrices, we can obtain new projection matrix $A(t+1)$ by solving Eq. (1). In incremental learning, finding a good value of d (dimension of the canonical space) is difficult because the structure of the canonical space changes dynamically. Here, we set a threshold for eigenvalues, and employ eigenvectors whose eigenvalues are greater than the threshold.

After updating the canonical space, we need to calculate the hidden variable \mathbf{s} again using $A(t+1)$. That is, $\mathbf{s}_i = A^T(t+1)\mathbf{x}_i$ for all $i \leq t+1$.

It is worth noting that, in spite of the algorithm's simplicity, it is guaranteed to provide the same solution as the batch process. This property makes our system highly stable, which is crucially important for realistic situation.

3.5. Image and Label Features

As the image feature, we use the color higher-order local auto-correlation (Color-HLAC) features [9]. This is a powerful global image feature for color images. In general,

global image features are suitable for realizing scalable systems because they can be extracted quite fast. Also, they are well suited for weak labeling problem where we cannot predict the location and the number of objects in input images. The HLAC features enumerate all combinations of mask patterns that define autocorrelations of neighboring points. In this paper we use at most the 1st order correlations. Also, we extract Color-HLAC features from the original images and weakly binarized images as described in [12].

We use the word histogram as the labels feature. In this work, each image is simply annotated with a few words, so the word histogram becomes a binary feature [12].

4. Benchmark Test

We use the Corel5K dataset [4] to evaluate the performance of the proposed method. This dataset contains 5000 pairs of the image and the labels, and has become the de facto standard for the problem of image annotation with multiple words. Each image is manually annotated with one to five words. There are 4500 images in the training data set and 500 images in the test data set. The training data has 371 words. 260 words among them appear in the test data. We search the optimal parameters with 10-fold cross validation over the 4500 training data as in [12]. The experiment is conducted on a commercially available PC (dual Xeon 2.66 GHz, eight cores) with C++ implementation.

4.1. Evaluation Protocol

For evaluation, we follow the methodology of previous works. The evaluation method of annotation is as follows. The recognition system annotates 500 test images with 5 words each. These words are then compared with the original ones. We obtain Mean-Recall (MR) and Mean-Precision (MP). Because of the trade-off between these two scores, we evaluate the total performance using the F-measure: $F\text{-measure} = \frac{2 \times MR \times MP}{MR + MP}$.

As for the evaluation of retrieval, the retrieval system ranks all 500 test images for each testing word. We evaluate this performance with Mean Average Precision (MAP) in two cases: the MAP over all 260 test words, and over the words in which recall > 0 on annotation (MAP-RP).

4.2. Results

Table 1 shows the results obtained by the proposed method and various previously proposed methods using Corel5K. The proposed method outperforms the previously published methods in both annotation and retrieval.

Table 1. Performance comparison on Corel5K.

| | MR | MP | F-measure | MAP | MAP-RP |
|----------|------|------|-----------|------|--------|
| CRM [11] | 0.23 | 0.22 | 0.23 | 0.26 | 0.30 |
| MBRM [5] | 0.25 | 0.24 | 0.25 | 0.30 | 0.35 |
| SML [2] | 0.29 | 0.23 | 0.26 | 0.31 | 0.49 |
| Proposed | 0.32 | 0.25 | 0.28 | 0.32 | 0.58 |

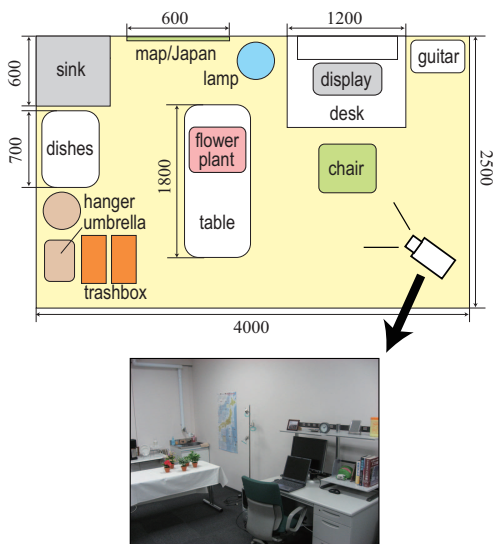


Figure 6. Layout of the experimental room.

4.3. Computational Costs

Here we compare the computational costs on the Corel5K data set. SML [2], which got the best result, used 3000 state-of-the-art Linux machines in 2005, and takes one hour for learning [2] and 280 seconds for the annotation [3]. Our method takes just 5 seconds for learning (solving CCA), and takes 10 seconds for annotation over all 500 test images on an eight-core desktop PC. Though we cannot propose clear comparison because hardware configurations are different, it can be said that our method is faster and more accurate than the previously published methods. In particular, the improvement in computation speed is immense. In this sense, our proposed algorithm is well suited for the mobile application we aim at.

Also, the computational costs of our recognition method and incremental learning method both increase in proportion to the number of training samples N . However, we emphasize that they can be perfectly parallelized because they are instance-based methods and do not need sequential estimation. This property is suitable for the trend of current computer hardware.



Figure 7. Examples of training data sets.

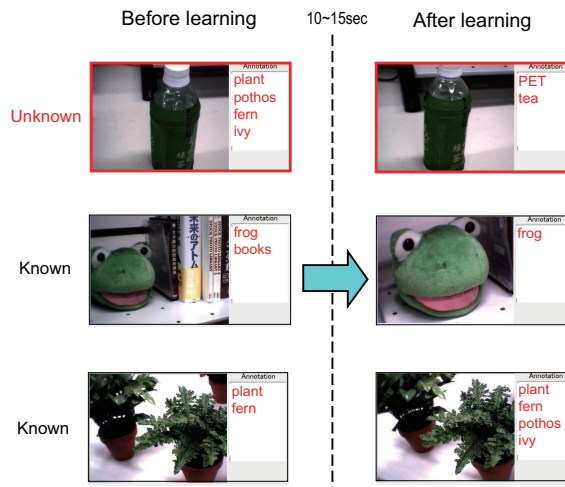


Figure 8. Example of incremental learning. The system newly learns PET bottle of green tea.

5. Demonstration of AI Goggles

In this section, we test the performance of our system in the real world. It is notable that the teaching phase is conducted mainly through online learning of new objects. After teaching the system, we actually wear the system and perform real-time annotation and retrieval.

5.1. Experimental Environment

As the main environment, we set up an experimental room simulating common life space (Fig. 6). We placed various objects ranging from large ones such as a desk, chair, and table, to small ones such as books, clock, and photograph (Figure 7 shows some representative ones). Also, we test our system outside of the room and teach some landmark objects and scenes. Overall, we teach 106 labels in the current setup.

As the initial training dataset, we capture some image samples using the goggles in advance and hand-label each of the samples with a few words. Figure 7 shows some examples. This dataset contains about 5000 samples of just 40 labels in the experimental room. We teach a large part of objects and scenes through online learning algorithm. The


| | | | | | |
|-------------------|---|---|---|--|---|
| Captured Images |  |  |  |  |  |
| System Annotation | books 0.99 guitar 0.99 frog 0.86 | sink 0.74 dishes 0.70 | lamp 0.67 desk 0.62 PC 0.46 | glass 0.80 flower 0.50 | map 0.99 Japan 0.92 |
| Captured Images |  |  |  |  |  |
| System Annotation | flower 0.96 plant 0.82 table 0.79 | phone 0.98 keyboard 0.79 mouse 0.78 | ball 0.89 | trashbox 0.62 umbrella 0.62 | clock 0.95 |
| Captured Images |  |  |  |  |  |
| System Annotation | bicycle 0.68 | street 0.65 building 0.65 | tree 0.87 forest 0.86 lake 0.86 | tree 0.87 stone 0.86 statue 0.86 | fountain 0.90 |

Figure 9. Example of annotation by the system.

objective here is to test how our system can adapt to a whole new environment (outside of the room).

5.2. Online Learning of New Objects

We show an example of teaching an unknown object to the system on site (Fig. 8). First, we show the system a plastic bottle of tea having green label. The system recognizes it as “plant”, which has the most similar color in known objects. Next, we define new labels “PET” and “tea” for this plastic bottle, and perform incremental learning of image and label. Just after the user clicks the “START” button on the GUI, the system calculates the proposed incremental algorithm every frame until “STOP” button is clicked. As a result, the system learns to recognize the plastic bottle in just about 10 seconds. It is also able to distinguish it from “plant” and “frog” learnt before, which have similar appearance and are thought to be confusing.

Our incremental algorithm takes just 0.1 seconds on an average for each frame to update the system in current setup, where normal CCA (batch process) takes 2.87 seconds. As these results show, our incremental algorithm is quite fast and effective. This function enforces man-machine interaction and leads to user-adaptive systems. It is also important to overcome the high intra-class variation of known objects.

5.3. Real-time Image Labeling and Retrieval

After completing learning, we actually wear the goggles and attempt real-time recognition of objects in the environment. The vision log, which contains the image itself and

the posterior probability (value of Eq. (3)) of each word, is saved at 1 fps. Figure 9 shows examples of the recognition results. The upper two rows show the results in the experimental room, and the last row shows those taken outside. The posterior probabilities of each word are shown next to the corresponding words. We set the threshold of posterior probability to 0.40. The recognition result is the set of objects whose posterior probability exceeds this threshold. As shown, the system succeeds in recognizing various generic objects both in the experimental room and outside. It is remarkable that though these two environments are considerably different, the incremental learning in the outside environment does not deteriorate the recognition performance in the experimental room. These results indicate the powerful and flexible recognition ability of the system.

We also attempt to retrieve images containing certain objects from the previously recorded vision log. Figure 10 shows some examples of retrieval, where the proposed system could retrieve target images correctly from a vision log containing many images. As “face” example shows, our system can recognize various objects as “face”. In other words, it can flexibly deal with the high intra-class variation. Also, our system permits multi-label retrieval. If we input “face” and “frog” as the query, the system correctly retrieves the target. Figure 11 illustrates a practical situation of retrieval. Suppose we cannot find a pair of pliers, and want to know where it is now. Then we let the system retrieve images of the pliers that we must have seen somewhere before. The system shows us a sequence of images of the last time the pliers were seen. This informs us that the pliers are currently in a drawer.

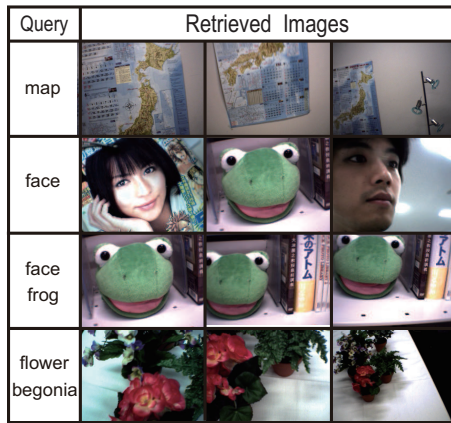


Figure 10. Example of retrieval in the experimental room.

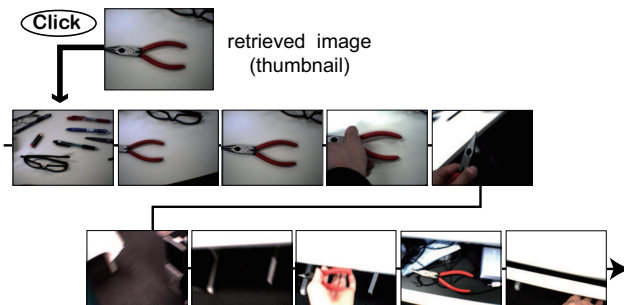


Figure 11. An example of memory assistance. The system retrieves a record of the last time the pliers were seen.

6. Conclusion

In this paper, we presented the AI Goggles system, which recognizes and retrieves various generic objects and scenes present in the environment. Using the goggles, we can retrieve what we have seen in the past using a search query, just like an internet search. Also, it enables instant learning of new objects and scenes. This ability further enhances man-machine interaction. We can optimize the labeling policy of the system flexibly according to the user's preference. Also, the system can adapt to a new environment as our experiment showed. This system may have many practical applications, such as memory assistance, visual dictionary, automatic blog writer, and so on.

This system is implemented by the proposed image annotation and retrieval method. Our method outperforms previous methods in terms of both accuracy and computational speed in multi-labeling task. In particular, the significant improvement in speed enables image annotation and retrieval even in a tiny notebook computer. This makes it

possible for mobile systems to label and retrieve real-world visual information instantly. We verified the effectiveness of the method through experiments involving use of the AI Goggles system in the real world.

References

- [1] F. R. Bach and M. I. Jordan. A probabilistic interpretation of canonical correlation analysis. Technical Report 688, Department of Statistics, University of California, Berkeley, 2005.
- [2] G. Carneiro, A. B. Chan, P. J. Moreno, and N. Vasconcelos. Supervised learning of semantic classes for image annotation and retrieval. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 29(3):394–410, 2007.
- [3] G. Carneiro and N. Vasconcelos. Formulating semantic image annotation as a supervised learning problem. In *Proc. IEEE Conf. CVPR*, volume 2, pages 163–168, 2005.
- [4] P. Duygulu, K. Barnard, and D. F. N. Freitas. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *Proc. ECCV*, pages 349–354, 2002.
- [5] S. Feng, R. Manmatha, and V. Lavrenko. Multiple bernoulli relevance models for image and video annotation. In *Proc. IEEE Conf. CVPR*, volume 2, pages 1002–1009, 2004.
- [6] J. Gemmell, G. Bell, R. Lueder, S. Drucker, and C. Wong. Mylifebits: Fulfilling the memex vision. In *Proc. ACM Multi Media*, pages 235–238, 2002.
- [7] D. R. Hardoon, C. Saunders, S. Szedmak, and J. Shawe-Taylor. A correlation approach for automatic image annotation. In *Proc. International Conference on Advanced Data Mining and Applications*, 2006.
- [8] D. R. Hardoon, S. Szedmak, and J. Shawe-Taylor. Canonical correlation analysis; an overview with application to learning methods. 16(12):2639–2664, 2004.
- [9] T. Kato, T. Kurita, N. Otsu, and K. Hirata. A sketch retrieval method for full color image database –query by visual example–. In *Proc. ICPR*, volume 2, pages 213–216, 1992.
- [10] P. L. Lai and C. Fyfe. A neural implementation of canonical correlation analysis. *Neural Networks*, 12:1391–1397, 1999.
- [11] V. Lavrenko, R. Manmatha, and J. Jeon. A model for learning the semantics of pictures. In *Proc. NIPS*, 2003.
- [12] H. Nakayama, T. Harada, Y. Kuniyoshi, and N. Otsu. High-performance image annotation and retrieval for weakly labeled images using latent space learning. In *Proc. PCM*, pages 601–610, 2008.
- [13] B. Schiele, T. Jebara, and N. Oliver. Sensory augmented computing: Wearing the museum's guide. *IEEE Micro*, 21:44–52, 2001.
- [14] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-based image retrieval at the end of the early years. *IEEE Trans. Pattern Analysis and Machine Intelligence*, pages 1–32, 2000.
- [15] A. Torralba, K. P. Murphy, W. T. Freeman, and M. A. Rubin. Context-based vision system for place and object recognition. In *Proc. IEEE ICCV*, 2003.