SPELLA - A Voice Interactive Object Recognition System

Sreejith C*
Sreeshma K*

Abstract

SPELLA is a user interactive speech enabled learning and recognition system. This paper presents the design of a recognition system that is able to identify, store and recognize objects. The system hardware consists of web camera, a headphone with a microphone and interface software. In this study we aim to devise a system that identifies and memorizes the object in a given environment under similar conditions using voice commands. For object recognition we employ various image processing techniques and Eigen method. Voice interaction is done with Microsoft speech application programming interface (SAPI). Future works will involve on the designing and implementing this user interacting system in the field of robotics.

.

Keywords: Eigen, Object recognition, SAPI, SPELLA

1. Introduction

As the holy grail of computer vision research is to tell a story from a single image or a sequence of images, object recognition has been studied for more than four decades. Significant efforts have been paid to develop representation schemes and algorithms aiming at recognizing generic objects in images taken under different imaging conditions (e.g., viewpoint, illumination, and occlusion). Within a limited scope of distinct objects, such as handwritten digits, fingerprints, faces, and road signs, substantial success has been achieved. Object recognition is also related to content- based image retrieval and multimedia indexing as a number of generic objects can be recognized. In addition, significant progress towards object categorization from images has been made in the recent years. Note that object recognition has also been studied extensively in psychology, computational neuroscience and cognitive science.

Object recognition is concerned with determining the identity of an object being observed in the image from a set of known labels. Oftentimes, it is assumed that the object being observed has been detected or there is a single object in the image. Object recognition in computer vision is the task of finding a given object in an image or video sequence. Humans recognize a

*Department of Computer Science and Engineering, MEA Engineering College, Kerala, India

multitude of objects in images with little effort, despite the fact that the image of the objects may vary somewhat in different viewpoints, in many different sizes / scale or even when they are translated or rotated.

Speech recognition (also known as automatic speech recognition or computer speech recognition) converts spoken words to text. The term "voice recognition" is sometimes used to refer to recognition systems that must be trained to a particular speaker-as is the case for most desktop recognition software. Recognizing can simplify the task of translating speech.

Speech recognition is a broader solution that refers to technology that can recognize speech without being targeted at single speaker-such as a call system that can recognize arbitrary voices Speech synthesis is the artificial production of human speech. A computer system used for this purpose is called a speech synthesizer, and can be implemented in software or hardware. A text-to-speech (TTS) system converts normal language text into speech; other systems render symbolic linguistic representations like phonetic transcriptions into speech

2. Proposed System

Evaluation of object detection systems requires a set of test images with objects in heterogeneous scenes. Unfortunately, existing publicly available object databases provide few, if any, test images suitable for evaluating object detection systems. Here we present the an Object Recognition system, a software package for creating patterns suitable for evaluating object detection systems. These are created by superimposing objects from existing publicly available object databases onto heterogeneous backgrounds. It is capable of creating various patterns focusing, gray scale conversion. This software package is being made publicly available to aid the computer vision community by providing various patterns which will allow object detection systems to be systematically compared and characterized.

Appearance-based object recognition methods have recently demonstrated good performance on a variety of problems. However, many of these methods either require good whole-object segmentation, which limits their performance in the presence of clutter, occlusion, or background changes; or utilize simple conjunctions of low-level features, which cause crosstalk problems as the number of objects is increased. To recognize an object, that is to answer the question "what object is in this image?" key features together with their local contexts are extracted from the image, and fed into the associative memory. All matches are retrieved, and for each match, the associated information is used to compute a hypothesis about the identity, view, and configuration of a possible object. This hypothesis is fed to a second, "working" associative memory, where current hypotheses are stored. If any matches are found, the evidence associated with them is updated to reflect the new information. Otherwise a new hypothesis is entered. The accumulation is not a flat voting process, but depends on the frequency of occurrence of the feature, with uncommon features providing more evidence. SPELLA represents speech enabled learning system which provides simple and efficient user interactive recognition system.

3. Eigen Faces

Eigenfaces are a set of eigenvectors used in the computer vision problem of human face recognition. The approach of using eigenfaces for recognition was developed by Sirovich and Kirby (1987) and used by Matthew Turk and Alex Pentland in face classification. It is considered the first successful example of facial recognition technology. These eigenvectors are derived from the covariance matrix of the probability distribution of the high-dimensional vector space of possible faces of human beings.

3.1 Eigen face generation

A set of eigenfaces can be generated by performing a mathematical process called principal component analysis (PCA) on a large set of images depicting different human faces. Informally, eigenfaces can be considered a set of "standardized face ingredients", derived from statistical analysis of many pictures of faces. Any human face can be considered to be a combination of these standard faces. For example, one's face might be composed of the average face plus 10% from eigenface 1, 55% from eigenface 2 and even -3% from eigenface 3. Remarkably, it does not take many eigenfaces combined together to achieve a fair approximation of most faces. Also, because a person's face is not recorded by a digital photograph, but instead as just a list of values (one value for each eigenface in the database used), much less space is taken for each person's face.

3.2 Practical Implementation

To create a set of eigenfaces, one must:

1. Prepare a training set of face images. The pictures constituting the training set should have been taken under the same lighting conditions, and must be normalized to have the eyes and mouths aligned across all images. They must also be all resample to the same pixel resolution. Each image is treated as one vector, simply by concatenating the rows of pixels in the original image, resulting in a single row with r × c elements. For this implementation, it is assumed that all images of the training set are stored in a single matrix T, where each row of the matrix is an image.
2. Subtract the mean. The average image a has to be calculated and then subtracted from each original image in T.
3. Calculate the eigenvectors and eigenvalues of the covariance matrix S. Each eigenvector has the same dimensionality (number of components) as the original images, and thus can itself be seen as an image. The eigenvectors of this covariance matrix are therefore called eigenfaces. They are the directions in which the images differ from the mean image. Usually this will be a computationally expensive step (if at all possible), but the practical applicability of eigenfaces stems from the possibility to compute the eigenvectors of S efficiently, without ever computing S explicitly, as detailed below.
4. Choose the principal components. The D x D covariance matrix will result in D eigenvectors, each representing a direction in the $r \times c$-dimensional image space. The eigenvectors (eigenfaces) with largest associated eigenvalue are kept.

4. SAPI and its Applications

SAPI (Speech Application Programming Interface) was first introduced to Windows 95. This API provides a unified interface for dynamic speech synthesis and recognition. Over the years new versions were developed and now it is version 4.0 with WinXP. Unfortunately the API wasn't really maturated and supported only C++ (later Visual Basic and COM), so it was quite widely used. Microsoft redesigned the version 5.0 from scratch and changed critical parts of the interface. However the latest stable version 5.1 is still a native code DLL, but with the next one, which is considered to be part of Windows Vista (a complete redesign again), A full support for managed .NET code will be expected [7]. Right now it is only possible to take advantage of the current SAPI interface via C# by using COM Interop, which is .NET technique to use native COM objects.

5. System Design

The System design of SPELLA is as shown below

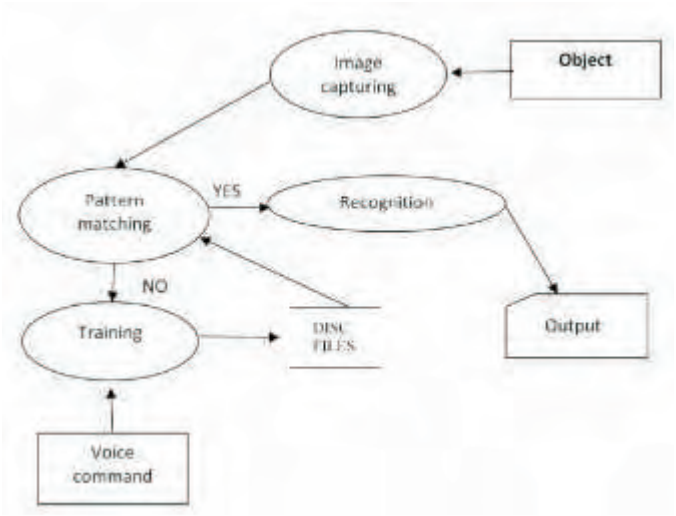

Fig. 1: System Design

The image of the object that is to be recognized is being captured using a webcam. The pattern of that image is compared with the existing patterns already in the disk files. If the pattern is matching with any of the existing ones, the object is recognized. If not, the system is trained about the new object and stored. Here commands are given through voice.

The figure below shows the interface of the system:

Fig. 2: SPELLA-The user interface

The Proposed system consists of four modules:

5.1 Speech recognition and synthesis

Speech recognition is the process of converting an acoustic signal, captured by a microphone, to a set of words. Speech synthesis is the artificial production of human speech. A text-to speech (TTS) system converts normal language text into speech.

This system is completely under the control of voice. And also it interacts to the user through voice. A text file named *Grammar.txt* is used to store the words/sentences used in the system. If we need to use a word during the interaction we have to store those words in this text file. It should be made sure that there are no extra vacant lines or spaces in between. All commands are predefined.

For e.g. if you want the system to answer the question, "How are you", we have to define the question and the response. While adding new commands, it should follow a common syntax. For e.g.:

Question: "How are you?"
Response: "I am fine"

The general syntax is as:

```
If (command = "How are you?")
        {
                        Speak ("I am fine.")
        }
```

5.2 Video Mode

The real time video is captured by any video capturing device.

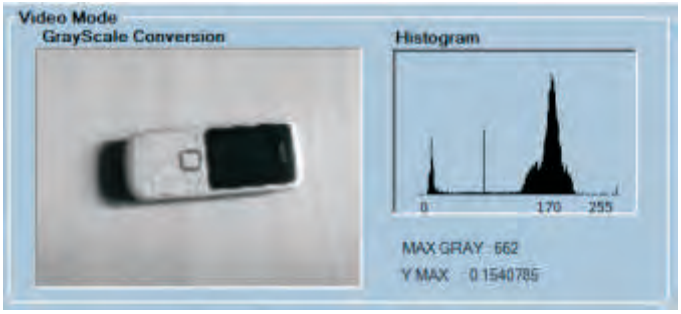

Fig. 3: Video mode

5.3 Training Mode

Prepare a training set of objects. The pictures constituting the training set should have been taken under same lighting conditions.

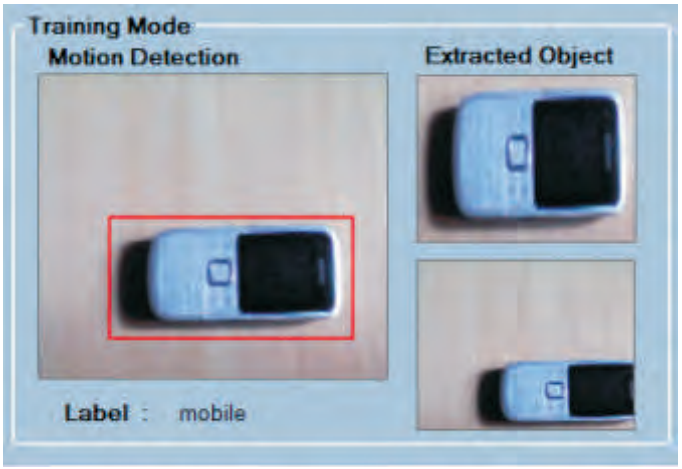

Fig. 4: Training mode

5.4 Recognition Mode

Object recognition in computer vision is the task of finding a given object in an image or video sequence.

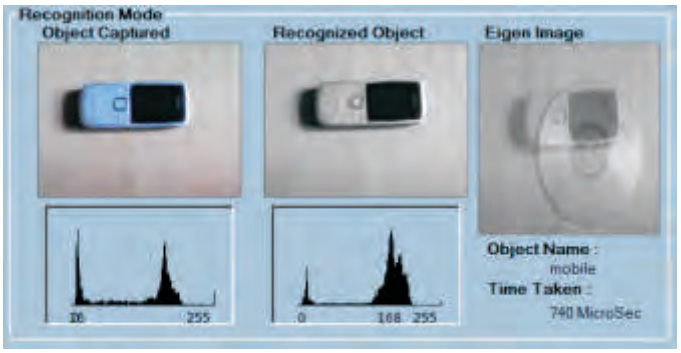

Fig. 5: Recognition mode

The flowchart of training and recognition phase is as shown below.

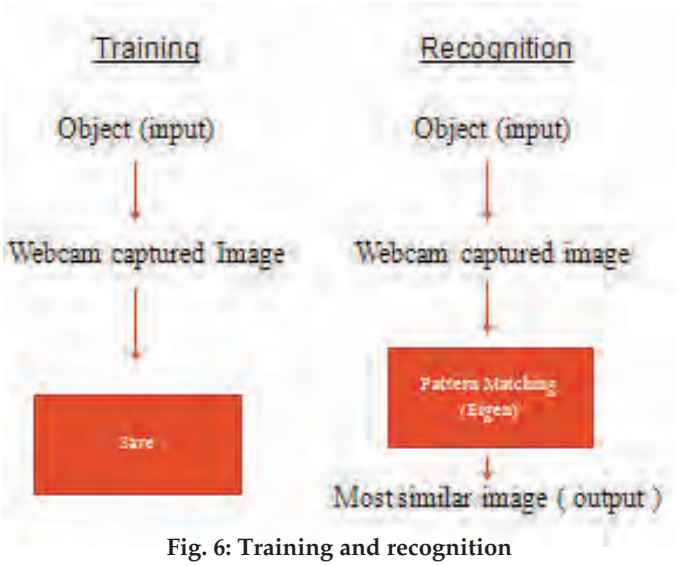

Fig. 6: Training and recognition

6. Experimental Study

We now consider some simple experiments which illustrate the matching performance of eigenface technique for object recognition. We compute the performance of these techniques when the background is known and the lightning conditions remains almost unchanged.

The experiment consist of two phases

1. Training phase
   Here we prepare a training set of objects.
2. Recognition phase
   Here the trained objects are recognized.

The experiment consist a trial set of 5 different objects. The experiment was conducted with 100 trials and the accurate trials are noted down. We considered five different sample objects including a pen, a toy car, a disc a cell phone, and a flash drive. Three tables are given here with the experiment results. The Table 1 records the details in which training set is eight.

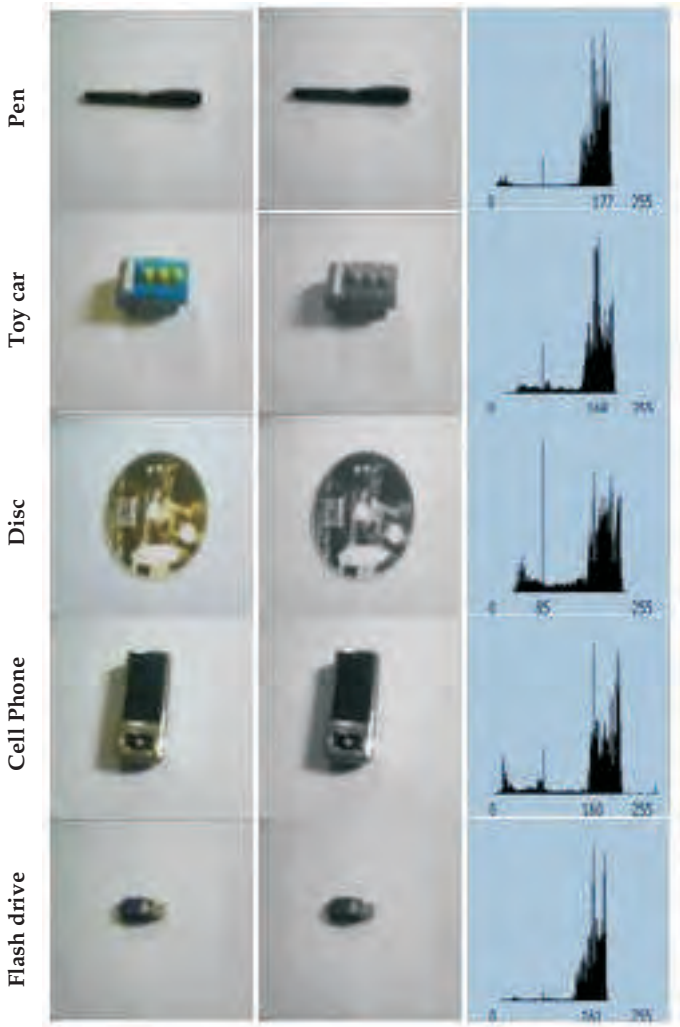

Fig. 7: A view of object used for trial along with grayscale converted image and histogram

Fig. 8: sample training of pen with 8 training set

The table 1 illustrates the experiment with 5 objects containing 8 training set. Of the 100 trials conducted, in the case of pen, 96 trials succeed in recognizing the object giving 96% of accuracy. Considering the other objects, toy car, disc, cell phone, flash drive, they deliver 92%, 100%, 100% and 93% of accuracy respectively.

Table 1: Result of trial, where number of training performed is eight

Object name	Number of training performed	Number of trials Conducted	Number of success trial	Percentage of accuracy
Pen	8	100	96	96%
Toy Car	8	100	92	92%
Disc	8	100	100	100%
Cell Phone	8	100	100	100%
Flash Drive	8	100	93	93%

7. Conclusion

In this work we tried to design a voice controlled user interactive system for object recognition. We experiment object recognition using Eigen method and used windows speech programming interface for speech recognition and synthesis. In brief this method can be utilized for accurate and efficient object recognition. SPELLA is simple and efficient and produce high rate of accuracy.

Future works will involve implementing vision system in the field of interactive robotics for the purpose of object recognising.

8. References

1. *Wilhelm Burger and Mark J. Burge Digital Image Processing: An Algorithmic Approach Using Java. Springer. ISBN 1846283795 and ISBN 3540309403, 2007.*
2. *Pedram Azad, Tilo Gockel, Rüdiger Dillmann Computer Vision - Principles and Practice. Elektor International Media BV. ISBN 0905705718, 2008.*
3. *M. Turk, A. Pentland. "Eigen faces for Recognition". Journal of Cognitive Neuroscience. Vol 3, No. 1. 71-86, 1991.*
4. *Kuttler, Kenneth (PDF), An introduction to linear algebra, Online e-book in PDF format, Brigham Young University, 2007.*
5. *Roth, Peter M. and Winter, Martin "Survey of Appearance-Based Methods for Object Recognition", Technical Report ICG-TR-01/08, Inst. for Computer Graphics and Vision, Graz University of Technology,Austria; January 15, 2008.*
6. *Speech-enabled windows application using Microsoft SAPI, IJCSNS International Journal of Computer Science and Network Security, VOL.6 No.9A, September 2006.*
7. *Harrington, M, 'Giving Computers a Voice', http://msdn.microsoft.com/coding4fun/inthebox/ttshw/default.aspx, 2006.*
8. *Shivani Agarwal, Aatif Awan, and Dan Roth," Learning to Detect Objects in Images via a Sparse, Part-Based Representation".*