

## Interactive Visual Pattern Recognition

George Nagy and Jie Zou

*Electrical, Computer, and System Engineering, Rensselaer Polytechnic Institute, Troy, NY*  
[nagy@ecse.rpi.edu](mailto:nagy@ecse.rpi.edu) and [zouj@rpi.edu](mailto:zouj@rpi.edu)

### Abstract

*Camera Assisted Visual Interactive Recognition (CAVIAR) draws on sequential pattern recognition, image database, expert systems, pen computing, and digital camera technology. It is designed to recognize wild flowers and other families of similar objects more accurately than machine vision and faster than most laypersons. The novelty of the approach is that human perceptual ability is exploited through interaction with the image of the unknown object. The computer remembers the characteristics of all previously seen classes, suggests possible operator actions, and displays confidence scores based on already detected features. On 80 test images of wildflowers, 10 laypersons averaged 80% recognition accuracy at 12 seconds per flower.*

### 1. Introduction

The goal of visual pattern recognition during the past fifty years has been the development of automated systems that rival or even surpass human accuracy, at higher speed and lower cost. Human interaction is considered, if at all, only to deal with “rejects” in the final step. However, there are many well-designed interactive systems, like word processors, computer-aided drafting, photo-editors and spreadsheets that help laypersons to achieve near-expert performance.

There are pronounced differences between human and machine cognitive abilities. Humans apply to recognition a rich set of contextual constraints and superior noise filtering abilities to excel in gestalt tasks, like object-background separation. Computers, however, can store thousands of images and associations between them, never forget a name or a label, and compute geometric moments and probability distributions.

These differences suggest that a system that combines human and machine abilities can, in some situations, outperform both. Wholly automatic systems do not scale up to large heterogeneous databases. The '92 US NSF workshop stated that “computer vision researchers should identify features required for interactive image understanding, rather than their discipline’s current

emphasis on automatic techniques” [1]. However, the current emphasis on interactive retrieval is more on query refinement than on operator-assisted feature extraction [2], [3]. Aside from content-based image retrieval [4], [5], [6], we have found little research published on the analysis and optimization of interactive systems for visual pattern recognition. We describe work in progress on a decision-theoretic model for interactive recognition of visual objects, and the construction and evaluation of a self-contained prototype system.

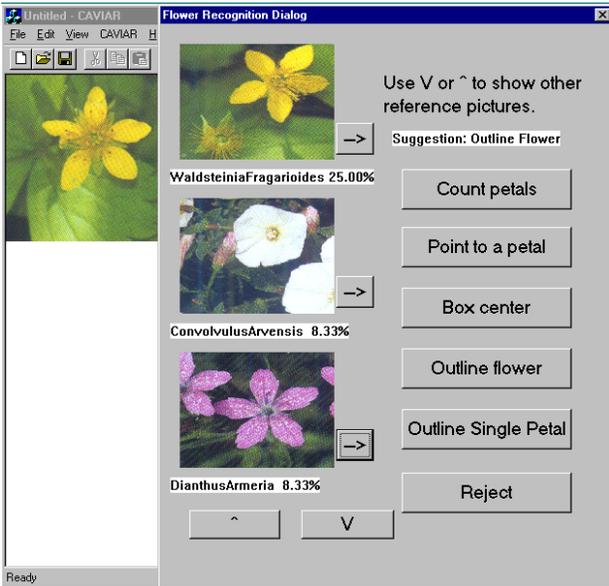
### 2. System architecture

The system includes, in addition to the interactive classifier, modules to adapt it to new families of objects, and to train the recognition engine. There is also a logging subsystem to record time-stamped user actions and extract statistics of interest. Although we intend to port the system to a digital-camera mounted on a pocket computer, or to a webcam mounted on a PC, the prototype was implemented on a Windows platform with mouse instead of stylus interaction, and we used “canned” pictures.

#### 2.1. Interactive classification

The interaction is based on the few primitive actions that can be executed easily with a stylus and a small, touch-sensitive display. The richness of the interaction results from its *interpretation*. The system is aware that the operator is pointing at a petal, a stamen, a blemish, or the tip of a leaf. (If the proposed interaction has already been executed on the reference image, that result will be displayed to help the operator. An operator who is unsure of what a *stamen* is can glance at the expected target highlighted on the currently displayed reference images.) A bounding box can be interpreted, as appropriate, as that of the whole flower, of a leaf, or of a distinctive secondary color. When automated segmentation fails, the operator need only point to the incorrect part of the boundary. Standard color, shape and texture features will be instantly computed on the designated part of the image. The top candidates, based on the new confidence values, are displayed. The operator action leading to the

potentially most discriminating feature is suggested (Figure 1).



**Figure 1. CAVIAR interaction. The unknown object is shown on the left, the current top candidates on the right.**

The recognition engine is a sequential k-nearest-neighbors classifier [7], [8]. The system ranks the candidate classes according to a confidence level based on all the features extracted so far:

$$P^n(\omega_i | x^1; f^1, \dots, x^n; f^n) = \frac{k_i + 1}{k + c}$$

where  $\omega_1, \dots, \omega_c$  are the candidate classes,  $x^1, \dots, x^n$  are the values of the feature vectors associated with user actions  $f^1, \dots, f^n$ , and  $k_i$  is the number of  $k$  nearest neighbours closest to the unknown among the training patterns in feature space. This Bayesian multinomial estimator does *not* assume class-conditional independence among the features. The next “best” interactive action is computed by minimizing the expected cost:

$$f_i^{n+1} = \arg \min_i R(f_i^{n+1})$$

$$R(f_i^{n+1}) = \alpha P(\text{error} | f_i^{n+1}) + (1 - \alpha) T_i(f_i^{n+1}),$$

where the cost function (*risk*  $R$ ) is a convex combination,  $\alpha \times \text{error} + (1 - \alpha) \times \text{time}$ , of the expected error and the normalized average duration  $T_i$  of each possible user action  $f_i$  at the  $(n+1)$ <sup>th</sup> stage.

The system always displays exemplars of the three currently most-probable classes. With a single click, the operator can request either other reference pictures of the same class, or of a different class. When she finds the reference picture most similar to that of the unknown

object, she clicks on the reference picture to assign its class to the unknown. If no acceptable match is found, she marks the image as *reject*. The purpose of the automated image processing and classifier functions is to save time by steering the operator to the most likely candidates.

In developing CAVIAR, we observe the following guidelines:

- It should take very little time and expertise to adapt the system to a new family of objects.
- Interaction should be intuitive, fast, simple, and *consistent*.
- Only a few reference images of each class (currently 2) should be necessary to prime the system.
- The human is always in charge: the computed suggestions can be heeded or discarded. The operator’s decision is final.
- Every interaction and calculation is logged and time-stamped to allow experimentation to track improvement in the performance of the system and of the operator, and to reveal weak points.

## 2.2. New Families of Objects

CAVIAR makes use of only a few primitive means of interacting with a picture: pointing, tracing, dragging, dialog boxes. New families of objects require only the assignment of new *interpretations* to these primitives. The operator needs to know what to point at or trace, or what multiple-choice question to answer. A domain expert and a CAVIAR analyst must choose the appropriate interpretations jointly.

Some of the tools provide input to automated feature extraction methods, like area or contour moments, colour histograms, and texture indicators. Eventually, we expect that CAVIAR will include most of the common image analysis operators. We currently make use of the comprehensive, public domain Intel computer vision library. Here again, a domain expert must select the most discriminating features for a new family of objects.

So far we have applied CAVIAR only to flowers, produce, and Chinese characters. We have not yet constructed the interface required for a domain expert without extensive computer knowledge to select tools and features for a new family of objects.

## 2.3. Training and Learning

Initially, a human *trainer* extracts all possible features from a few samples of each class of the current family.

The features extracted from newly classified objects can be added to the reference database to improve the estimates of the class-conditional feature probabilities. At the same time, the operator gains familiarity with the most discriminating features for each type of object. The objective of the system is to minimize a weighted

combination of classification error and operator time. We expect that a layperson will, after sufficient experience with the system, classify objects as accurately and as fast as a domain expert. At that time, the system will no longer be needed for that family of objects. This opens up whole families of educational applications.

### 2.4. Logging subsystem

Subjects begin a session by clicking on the CAVIAR icon on the Windows Desktop, and end the session with the QUIT button. The beginning and end of each user action and automated computation is time stamped and recorded. So are the values of the computed features, which can be used to improve subsequent estimates of the class-conditional feature probabilities that the system used for computing confidence measures and suggestions to the operator.

The log files are reformatted after each session to determine the classification accuracy and the average times for user decisions and operations. The session statistics are saved under the Subject's pseudonym and the date and time of the session in an Excel format.

Averages for multiple sessions by the same or different subjects are computed analogously. The log files also allow determination of both short term (within-session) and long term (over several sessions) improvement in the subject's performance.

### 3. Preliminary experiments

Our prototype is still rudimentary, but we have conducted some experiments on Chinese characters, fruit, and flowers. We report here only the latter: we classify ideographs only to demonstrate the versatility of the approach: it is not a sensible application for CAVIAR.

One hundred pictures of 10 classes of wildflowers (mostly white) were divided at random into 20 training pictures (two pictures per class) and 80 test pictures (eight pictures per class). The photographs were collected from botanical Web sites or scanned from flower guides. Most sites and books offer only one picture of each species. The images exhibit a great deal of variation in size, colour, and resolution. Figure 2 shows two examples of easily confused species. We must wait till spring to photograph flowers *in situ* or in a botanical garden. (Nurseries and florists have only *cultivars*, or hybrids, which don't fit neatly into any taxonomy.)

We performed the initial training ourselves: it took about thirty minutes. We also labelled the test set to allow automated determination of the classification accuracy of each subject.

Ten Subjects, chosen only by availability, participated in the experiments. They were given a one-page description of the tools, given five minutes of training on

using them (on a file of pictures excluded from the experiment), and asked to classify the test set.



Figure 2. Examples of easily confused species.

### 4. Results

The subjects took between 10 and 26 minutes (average: 16 minutes) to classify the 80 flowers. The average time to classify a flower was 12 seconds. The classification accuracy ranged from 66% to 88% (average: 80%). Table I shows the average time spent by the subjects using each tool, deciding which tool to use next, and finally choosing the class.

Table I – Experimental Results for 10 Subjects

Tool	Tool Time (sec)		Times used
	Average	Total	
COUNT PETALS	2.2	545	246
POINT TO PETAL	1.4	170	126
BOUNDINGBOX, CENTER	5.3	543	102
TRACE FLOWER	35.0	140	4
TRACE PETAL	25.2	277	11
Average tool time	2.1 seconds per sample		
	Decision Time(Sec)(before)		
	Average	Total	
COUNT PETALS	4.6	1129	
POINT TO PETAL	4.9	620	
BOUNDINGBOX, CENTER	8.0	817	
TRACE FLOWER	7.1	28	
TRACE PETAL	8.3	91	
FINAL DECISION	6.4	5109	
Average decision time	9.7 seconds per sample		

We see from Table I that the most popular tools were COUNT PETALS and POINT TO PETAL. These were also the fastest tools and produced the most discriminating features. The two tracing tools (“electronic

scissors”) offered a live-wire feature based on pre-computed colour-gradients [9], but were still relatively time-consuming. All subjects took significant time for the final classification, scrolling the display horizontally for “other flowers of the same class” and vertically for “flowers of other classes”.

The integrity of the reported time budgets was checked by comparing their sum to the total session time. Computation time accounted for less than 0.1% of the total session time, but may become more significant with larger training sets.

## 5. Discussion

We demonstrated a small but complete system for interactive classification of visual objects. A variety of subjects were able to interact with pictures simply and effectively. However, it was only a demonstration, not a real experiment. The number of classes was far too small to show the value of the automated computations, because with at most three clicks the subjects could inspect an example of every class. This is reflected in the low usage of the automated tools. Furthermore, the artifacts in the data and the minimal number of training samples inevitably bias the results. We have, however, collected many valuable suggestions from both the subjects and others. In the next several months we hope to collect a larger database and to add some interactive tools and automated features.

We have not yet compared CAVIAR to either an automated flower recognizer, or to a human without computer assistance (for instance, using a flower guide only). If we do find an automated classifier, we will incorporate it into CAVIAR and let the operator begin with the final ranking and confidence values computed by the automated system.

The questions that we expect to shed light on, through further research, are the following.

- Can an interactive system provide more accurate classification, for certain families of objects, than a conventional automated classifier?
  - If so, how much human time is required? How quickly does it decrease with experience? How much of the decrease is due to human learning, and how much to machine improvement through exposure to more samples? Are the two correlated?
  - How well can the accuracy and time be predicted from statistical analysis of the training set characteristics? From human performance on small samples?
  - What features are best suited for interactive extraction in contrast to automated methods?
- What characteristics of a classification task (i.e., of a family of objects) determine its suitability for interactive classification? How widely is interactive classification applicable?

Possible modes of deployment include web cameras with server-mediated classification, camera-back interaction, PDA-camera combinations, and self-contained stationary systems for industrial or luggage inspection. We are interested in applications where one option available to the operator is to take another photograph from a different angle or distance. We also hope to find partners to apply CAVIAR to industrial classification and *training*, and to K1-12 and university-level education. However, our principal research objective is to establish a sound basis for partitioning the necessary tasks between the operator and the machine, which can serve as the foundation for a theory of interactive visual recognition.

## 6. Acknowledgements

We are grateful for technical advice and encouragement to Rebecca Seth, naturalist at the City of Lincoln Nature Center, and to Prof. Sharad Seth, of the UNL Department of Computer Science and Engineering.

## References

- [1] R. Jain, *US NSF Workshop on Visual Information Management Systems*, 1992.
- [2] C. Meilhac and C. Nastar, “Relevance Feedback and Category Search in Image Database,” *Proc. Int’l Conf. Multimedia Computing and Systems*, 512-517, 1999.
- [3] A. Vailaya, M. Figueredo, A. Jain, and H. Zhang, “Content-based Hierarchical Classification of Vacation Images,” *Proc. Int’l Conf. Multimedia Computing and Systems*, 1999.
- [4] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker, “Query by Image and Video Content: The QBIC Systems,” *IEEE Computer*, vol. 28, 23-32, 1995.
- [5] T. Gevers, A. W. M. Smeulders, “PicToSeek: Combining Color and Shape Invariant Features for Image Retrieval,” *IEEE Trans. Image Processing*, vol. 9, 102-119, 2000.
- [6] I. J. Cox, M. L. Miller, T. P. Minka, T. V. Pappathomas, and P. N. Yianilos, “The Bayesian Image Retrieval System, PicHunter: Theory, Implementation, and Psychophysical Experiments,” *IEEE Trans. Image Processing*, vol. 9, 20-37, 2000.
- [7] K. S. Fu, Y. T. Chien, and G.P. Cardillo, “A Dynamic Programming Approach to Sequential Pattern Recognition,” *IEEE-EC*, 16, 313-326, 1967.
- [8] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification and Scene Analysis*, Wiley, 2000.
- [9] E. N. Mortensen and W. A. Barrett, “Interactive Segmentation with Intelligent Scissors,” *Graphi99cal Models and Image Processing*, vol. 60, 349-384, 1998.