Submitted to IEEE Conference on Computer Vision and Pattern Recognition, 2004

Computer Assisted Visual InterActive Recognition: Formal Description and Evaluation

Jie Zou and George Nagy
Department of Electrical, Computer, and System Engineering
Rensselaer Polytechnic Institute
Troy, New York, 12180

Abstract

In the proposed computer assisted visual interactive recognition (CAVIAR) methodology, a parameterized geometrical model serves as the human-computer communication channel. The iterative CAVIAR process is modelled as a finite state machine. A flower recognition system is implemented based on the proposed methodology. Evaluation on 30 subjects shows that 1) the accuracy of the CAVIAR system is 90% compared to 50% for the machine alone; 2) its recognition time is 10.7 seconds compared to 26.4 seconds for the human alone; 3) it can be initialized with as few as one training sample per class and still achieve high accuracy; 4) it demonstrates a self-learning ability.

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 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 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.

As early as 1992, a workshop organized by US National Science Foundation in Redwood, California, stated that “computer vision researchers should identify features required for *interactive image understanding*, rather than their discipline’s current emphasis on automatic techniques” [4]. A more recent panel discussion at the 27th AIPR Workshop also emphasized “... the needs for Computer-Assisted Imagery Recognition Technology” [6]. In the pattern recognition and computer vision community, more and more researchers realize that fully automated model-based vision will not be feasible for a long time [5].

We study the role of interaction in a narrow domain, where higher accuracy is required than is currently achievable by automated systems, but where there is enough time for limited human interaction. In the broad domain of content-based image retrieval, *relevance feedback* has been found effective [8]. Interaction is, however, necessarily limited to selection of acceptable and not-acceptable responses, because there is no effective way to interact with arbitrary images. Interaction with the image was demonstrated in recent work in the narrow domains of face and sign recognition, which are comparable to flower recognition. However, it was confined to preprocessing, i.e., establishing the pupil-to-pupil baseline [9] or text bounding box [3][10]. We have not found any previous work advocating image-based interaction through a domain-specific model, which is the principal contribution of this paper.

In CAVIAR, the user may interact with the image anytime that he or she considers the computer’s response unsatisfactory. The interaction extracts some features directly, and improves the accuracy of other extracted features indirectly, by improving the fit of the computer-proposed model. Fitting the model requires only gestalt perception, rather than familiarity with the distinguishing features of the classes. The computer makes subsequent use of the parameters of the improved model to improve not only its own statistical model-fitting process, but also its internal classifier. Classifier adaptation is based on unsupervised decision-directed approximation [2][7][1], and therefore benefits by human confirmation of the final identification. The automated parts of the system gradually improve, and decrease the need for human intervention. As an important byproduct, the human’s judgment of when interaction is beneficial also improves. Our experiments demonstrate both phenomena.

The key to efficient interaction is the display of the automatically fitted model that allows the human to retain the initiative throughout the classification process. We believe that such interaction must be based on a visible model, because (1) a high-dimensional feature space is incompre-

hensible to the human, and (2) the human is not familiar with the properties of the various classes, and therefore cannot judge the adequacy of the current decision boundaries. Therefore he or she cannot interact efficiently with the feature-based classifier itself. We note, moreover, that human judgment of the adequacy of the machine-proposed prototypes, compared visually to the unknown object, is far superior to any classifier-generated confidence measure. In contrast to classification, deciding whether two pictures are likely to represent the same class does not require familiarity with all the classes.

We propose a formal model for interactive visual recognition, apply it to wild-flower recognition, and evaluate it on "naive" subjects.

2. Formal description

CAVIAR is a methodology for interactively recognizing objects. A collection of objects under consideration of a particular CAVIAR system is called an *object family* O. A generic object is denoted as o, $o \in O$. Each object has a category *label* l. The collection of all possible labels is called *label space* L. An image of an object is called a *picture* p. The collection of all possible pictures is called *picture space* P. The goal of a particular CAVIAR task is to recognize an object by algorithmic and human analysis of a picture of that object.

2.1 Parameterized geometrical model

For each application, a *parameterized* CAVIAR model[1] of the entire object family is created to facilitate the communication between human and computer. The communication (*interaction*) helps the machine to extract discriminating attributes for classification.

A complete CAVIAR model characterizes the *shape* of the components of the objects and the *geometrical relations* among components. It is completely defined by a set of *model parameters*. The collection of all the possible values of the model parameters constitutes the *model space* $\mathbf{\Theta}$. Particular objects are described by *model instances* $\boldsymbol{\theta}$. The model plays a central role in the communication between humans and computers, and should therefore be *understandable* and *adjustable* by the user, and provide enough information to the machine for evaluating discriminating attributes.

A CAVIAR model characterizes only geometrical information about the object, so it can also be considered as a *parametric partition* of the picture. The pixels of the picture are partitioned into mutually exclusive components of the object and of the background. A set of scalar-valued discriminating attributes (called a *feature vector* $\mathbf{x}$, $\mathbf{x} \in \mathbf{X}$) is extracted based on the partition.

2.2 Model building

Visible model instances (typically idealized contours of the objects) are presented to the user. The user can adjust them through a graphic user interface, where the model instance is superimposed on the unknown picture. Each adjustment is called a *model manipulation*, Υ_{MM}.

Model parameters can also be adjusted algorithmically, via *model estimation*, Υ_{ME}. Model estimation utilizes statistics accumulated from a set of labelled pictures (a *training set* s), each of which is associated with both a model instance and a feature vector that describes the object in the picture. The training set should include at least one picture under each label. A particular element of the training set is denoted as $s^i = \{p^i, l^i, \boldsymbol{\theta^i}, \mathbf{x}^i\}$.

Model estimation accepts all the model parameters that have been adjusted by model manipulation, then estimates all the remaining parameters.[2] (In other words, the human has the final say.) The combination of model manipulation and model estimation is called *model building*, Υ_{MB}. Model building is a transformation from picture space and model space to model space:

$$\Upsilon_{MB} : (P, \mathbf{\Theta}) \rightarrow \mathbf{\Theta}.$$

A single step of model building for a particular picture, based on the current model instance $\boldsymbol{\theta}^n$, is written as

$$\Upsilon_{MB} : (P, \boldsymbol{\theta}^n, s) \rightarrow \boldsymbol{\theta}^{n+1}.$$

A model building step consists of model manipulation Υ_{MM} followed by model estimation Υ_{ME}. Model manipulation, where one model parameter θ_i^n is adjusted to θ_i^{n+1}, is written as:[3]

$$\Upsilon_{MM} : (\boldsymbol{\theta}^n) \rightarrow \boldsymbol{\theta}^{n'}$$

where

$$\boldsymbol{\theta}^n = \{\theta_1^n, \theta_2^n, \cdots, \theta_i^n, \cdots, \theta_k^n\},$$

and

$$\boldsymbol{\theta}^{n'} = \{\theta_1^n, \theta_2^n, \cdots, \theta_i^{n+1}, \cdots, \theta_k^n\}.$$

Model estimation can be written as

$$\Upsilon_{ME} : (p, \boldsymbol{\theta}^{n'}, s) \rightarrow \boldsymbol{\theta}^{n+1},$$

[1] It can be useful to define multiple models for an object family. However, these models can always be unified, in principle, into a single model with more parameters. For simplicity, a single model is used in this formal description.

[2] In some special cases, model manipulation or model estimation is NULL. For example, in the initial step, a model instance can be calculated by the model estimation without any user intervention. On the other hand, model estimation is trivial if all the model parameters have already been manipulated.

[3] Depending on the GUI, the user can adjust one or several parameters simultaneously. Here, only one parameter θ_i^n is adjusted to θ_i^{n+1}.

where

$$\boldsymbol{\theta}^{n+1} = \{\theta_1^{n+1}, \theta_2^{n+1}, \cdots, \theta_i^{n+1}, \cdots, \theta_k^{n+1}\}.$$

A new set of parameters (i.e., a new CAVIAR model instance) is estimated with the parameter θ_i^{n+1} left unchanged. Several model building steps may be necessary to build a model instance that describes the discriminating aspects of the object.

2.3 Feature extraction

Feature extraction in CAVIAR, denoted as Υ_{FE}, is the algorithmic process of extracting a feature vector based on the model instance obtained by model building. It is a transformation from picture space and model space to feature space,

$$\Upsilon_{FE} : (P, \boldsymbol{\Theta}) \rightarrow \mathbf{X}.$$

Feature extraction from a specific picture p, based on a model instance $\boldsymbol{\theta}$, is written as

$$\Upsilon_{FE} : (p, \boldsymbol{\theta}) \rightarrow \mathbf{x}.$$

2.4 Indexing

Indexing Υ_{CI} is the algorithmic process of computing the similarity r_i of the unknown feature vector to the training sample associated with label l_i. It is therefore a transformation from feature space $\mathbf{X}$ to index space $\mathbf{R}$ of index vectors $\mathbf{r}$ with elements r_i:

$$\Upsilon_{CI} : \mathbf{X} \rightarrow \mathbf{R}.$$

In order to indicate that indexing utilizes the information accumulated from a particular training set, it is written as:

$$\Upsilon_{CI} : (\mathbf{x}, s) \rightarrow \mathbf{r}.$$

The results of indexing are presented to the user *visually* on the CAVIAR GUI. This display is called *visible index*.

A commonly used visible index, which we call *visible rank order*, is a sequence of pictures ordered according to their similarities to the unknown picture. The user can compare the unknown picture to the visible rank order by *browsing* it.

2.5 The CAVIAR finite state machine

A 3-tuple $q_{o,p} = \{\boldsymbol{\theta}, \mathbf{x}, \mathbf{r}\}$ is called an *interactive visual recognition state*. *Interactive visual recognition* is a sequence of states $\{q_{o,p}^0, q_{o,p}^1, \cdots q_{o,p}^n\}$. To start a recognition task, an initial state is created automatically:

$$q_{o,p}^0 = \{\boldsymbol{\theta}^0, \mathbf{x}^0, \mathbf{r}^0\}$$

Figure 1: Several examples in our flower database.

where

$$\Upsilon_{ME} : (p, s) \rightarrow \boldsymbol{\theta}^0,$$
$$\Upsilon_{FE} : (p, \boldsymbol{\theta}^0) \rightarrow \mathbf{x}^0,$$
$$\Upsilon_{CI} : (\mathbf{x}^0, s) \rightarrow \mathbf{r}^0.$$

Model manipulation leads to a state transition:

$$\{\boldsymbol{\theta}^n, \mathbf{x}^n, \mathbf{r}^n\} \longrightarrow \{\boldsymbol{\theta}^{n+1}, \mathbf{x}^{n+1}, \mathbf{r}^{n+1}\}$$

where

$$\Upsilon_{MB} : (p, \boldsymbol{\theta}^n, s) \rightarrow \boldsymbol{\theta}^{n+1},$$
$$\Upsilon_{FE} : (p, \boldsymbol{\theta}^{n+1}) \rightarrow \mathbf{x}^{n+1},$$
$$\Upsilon_{CI} : (\mathbf{x}^{n+1}, s) \rightarrow \mathbf{r}^{n+1}.$$

2.6 Identification

Identification assigns a label to the object based on the index vector. It is a transformation from the index space to label space,

$$\Upsilon_I : \mathbf{R} \rightarrow L, \text{ and } \Upsilon_I : \mathbf{r} \rightarrow l.$$

Identification can be performed algorithmically, but in CAVIAR concluding a pattern recognition task requires that the user identify the object by selecting one candidate from the visible index.

3. CAVIAR flower recognition system

Following the methodology proposed in Section 2, we developed an experimental flower recognition system to:
1) demonstrate the methodology with a concrete example; and 2) verify the hypothesis that it can, in some situations, outperform human-alone and computer-alone.

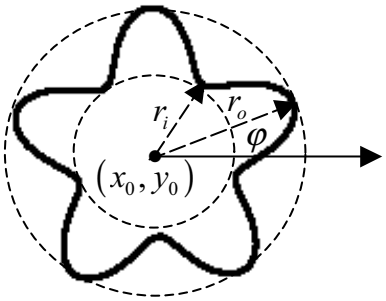

Figure 2: An example of the rose curve.

We collected a database of 987 flowers with a digital camera. The flower recognition system discussed here was developed on a subset of 216 flowers from 29 classes and evaluated on a subset of 102 classes with 6 samples per class.

All pictures are 320 by 240 pixels. The pictures were taken under highly variable illumination. The majority of the flowers are yellow, white, red, or blue. The background is the real scene, which can be very complicated. We do not assume that the flower is isolated from other flowers of the same or other species, because they often overlap in photos of flowerbeds (Figure 1).

3.1 Rose curve model

The *rhodonea (rose curve)* was defined by the Italian mathematician Guido Grandi between 1723 and 1728. We use a slightly modified rose curve to model the flowers:

$$\rho = \frac{r_o + r_i}{2} + \frac{r_o - r_i}{2} cos(n\theta + n\varphi) = a + bcos(n\theta + n\varphi) \quad (1)$$

A particular rose curve model instance (Figure 2) is completely determined by 6 parameters: the center (x_0, y_0), the outer radius r_o, the inner radius r_i, the number of petals n, and the initial phase φ.

$$\boldsymbol{\theta} = \{x_0, y_0, r_o, r_i, n, \varphi\}.$$

This model assumes that the flowers have circular symmetry and are composed of petals. The petals taper towards their tips. We restrict the possible number of petals n to the range $[3, 8]$, and use a circle ($n = 0$) for the rest.

3.2 Model building

Figure 3 shows the Graphic User Interface. The blue curve superimposed on the unknown picture is the visible rose curve model instance. The dots on the curve are the inner and outer radius control points. The rays from the center of the rose curve to the outer radius control points indicate the number of petals and the outer radius.

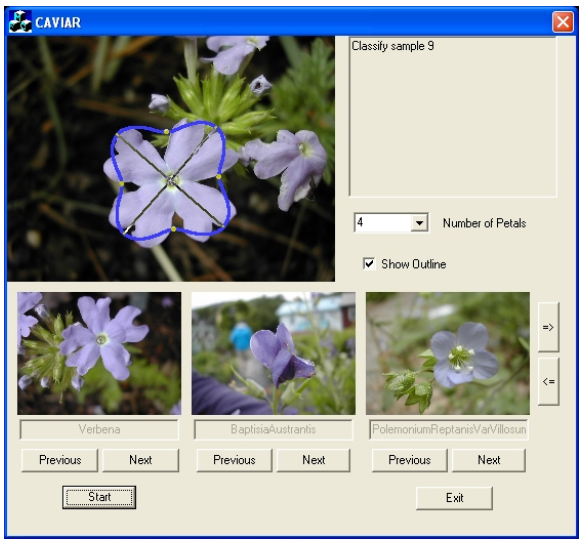

Figure 3: The GUI of the CAVIAR system for flowers. The rose curve is superimposed on the unknown picture. The visible rank order is computed and the top three candidates are displayed.

The user can compare the real flower boundary and the visible rose curve and, if necessary, adjust the rose curve by dragging the center point and the inner and outer radius control points. The number of petals can be changed with the ComboBox. When the user adjusts a parameter, the computer always accepts the adjustment. The remaining parameters are re-estimated as follows.

From the prior RGB color histogram generated from training samples, we compute the likelihood of each pixel being a foreground (flower) pixel and transform the color picture into a likelihood map, $P(x, y)$.

The center (x_0, y_0) is the centroid (mean) of the likelihood map:

$$x_0 = \frac{\int\int xP(x,y)dxdy}{\int\int P(x,y)dxdy}, y_0 = \frac{\int\int yP(x,y)dxdy}{\int\int P(x,y)dxdy} \quad (2)$$

To estimate n, we introduce a Fourier-like transform from the 2D binary image space to the discrete 1D K-space:

$$\phi(k) = \int\int P(x,y)e^{jk\theta}dxdy, k \in Z^+,$$

then,

$$n = \underset{k}{\text{argmax}} \left(|\phi(k)|\right) = \underset{k}{\text{argmax}} \left(|\int\int P(x,y)e^{jk\theta}dxdy|\right) \quad (3)$$

To estimate r_o, r_i, and φ, we define on $P(x,y)$ the following integrals, which are similar to geometric moments.

$$E_0(P(x,y)) = \int\int P(x,y)dxdy$$

$$E_{COS}(P(x,y)) = \int\int P(x,y)\cos n\theta dxdy$$

$$E_{SIN}(P(x,y)) = \int\int P(x,y)\sin n\theta dxdy$$

For an ideal rose curve silhouette $B_0(x,y)$,

$$E_0(B_0(x,y)) = a^2\pi + \frac{b^2}{2}\pi$$

$$E_{COS}(B_0(x,y)) = ab\pi\cos n\varphi$$

$$E_{SIN}(B_0(x,y)) = ab\pi\sin n\varphi$$

$E_0(P(x,y))$, $E_{COS}(P(x,y))$, and $E_{SIN}(P(x,y))$ can be computed for any $P(x,y)$. We force them to be equal to the corresponding integrals of the ideal rose curve silhouette. Thus,

$$E_0(P(x,y)) = a^2\pi + \frac{b^2}{2}\pi$$

$$E_{COS}(P(x,y)) = ab\pi\cos n\varphi$$

$$E_{SIN}(P(x,y)) = ab\pi\sin n\varphi$$

Solving for a, b, and φ:

$$a = \sqrt{\frac{E_0}{2\pi} + \frac{\sqrt{E_0^2 - 2(E_{COS}^2 + E_{SIN}^2)}}{2\pi}} \quad (4)$$

$$b = \sqrt{\frac{E_0}{\pi} - \frac{\sqrt{E_0^2 - 2(E_{COS}^2 + E_{SIN}^2)}}{\pi}} \quad (5)$$

$$\varphi = \frac{1}{n}\arctan(\frac{E_{SIN}}{E_{COS}}) \quad (6)$$

Figure 4 shows four model building steps on a difficult example (the picture is out of focus).

3.3 Feature extraction

From the rose curve model, eight features are derived for classification. The two global shape features are the petal number n and the ratio $\eta = r_o/r_i$. The color values of the pixels within the rose curve are converted from RGB to HSI. Then the histograms of hue and saturation are generated. The six color features $h_1, h_2, h_3, s_1, s_2, s_3$ are the first three moments of the hue and saturation histograms. This process is automated: our earlier experiments suggest that, once the model instance is refined, human intervention is of very limited value in feature extraction.

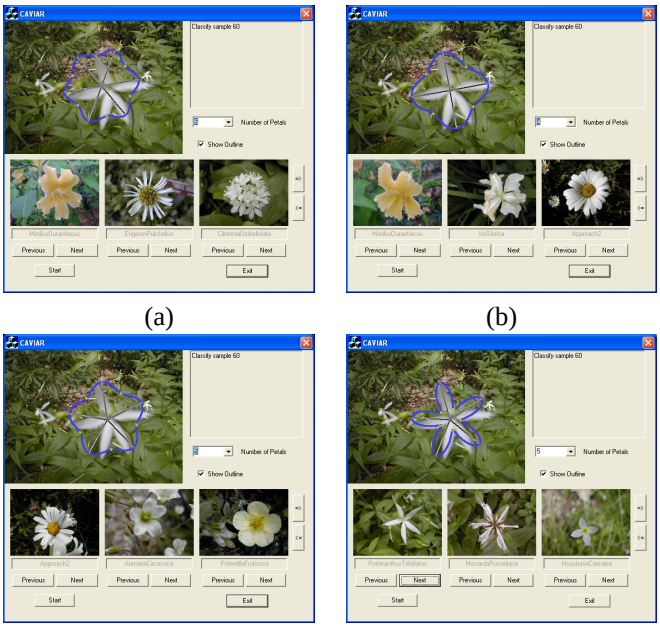

Figure 4: The rose curve is superimposed on the unknown picture. The top 3 candidates are displayed. After three interactive model building steps, the correct candidate appears in the first place. (a) initial automatic result, (b) after adjusting the rose curve center, (c) after adjusting petal number, (d) after adjusting inner circle radius.

3.4 Indexing

The standard deviation of the probability distribution of each feature extracted from the training samples is computed offline for normalization. The normalized distance of the unknown to each training sample is computed. Each element of the index vector is the distance from the unknown sample to the closest sample of the corresponding class.

Our CAVIAR flower recognition system displays the visible rank order based on the index vector. The user can browse all species by clicking on the "⇒" or "⇐" buttons, or browse all examples of the same class by clicking on the "Previous" or "Next" button (Figure 3).

3.5 Interactive flower recognition

The initial interactive visual recognition state is created automatically. The core algorithm [11], takes the following three steps: 1) the coarse flower region is found with a circle model; 2) strong segmentation by a *seeded watershed* algorithm based on training set statistics[4]; 3) the initial rose

[4]We subsequently discovered, however, that weak segmentation by fitting a rose curve directly yields almost as good initial rank ordering as our elaborate and accurate strong segmentation algorithm.

curve is fitted to the resulting boundary. The top candidates are then predicted and displayed to the user.

The computer's initial prediction (or after interactive model building) does not always agree with the human's. The visible model instance and the visible rank order provide the information necessary for the user to decide whether to move to a new state by model manipulation, to browse, or to identify the flower by clicking on one of the visible candidates.

Figure 4 shows a difficult example. The automatically estimated rose curve is poor due to the blurring of photo. After each of three model manipulations (center, petal number, and inner circle radius), the computer re-estimates the remaining un-adjusted parameters, and predicts new top candidates. Finally, the computer displays the correct candidate.

4. Evaluation

A subset of the flower database, 102 classes with 6 samples of each, was used for evaluation. The detailed experimental protocol, data collection, and analysis are described in [12]. Here we present only the results and a brief discussion.

Thirty subjects participated in 5 experiments, 6 subjects for each. In Experiment I, there is no computer assistant, the order of the candidate flowers is fixed, and the subject can only browse the candidate pictures to identify the unknown sample. Experiments II to V are all interactive experiments, but based on different training samples. All training samples of Experiment II are correctly labelled. Experiment III uses only one training sample for each class. The training set of Experiment IV includes all of the training samples and the interactively recognized unknown samples of Experiment III. Some of the interactively recognized samples are not correctly-labelled, so we call them *pseudo-training samples*. The training set of Experiment V includes the training samples of Experiment III and the pseudo-training samples generated by Experiments III and IV. Every subject labels 102 pictures (one per class) excluded from the training set for that experiment.

4.1 CAVIAR compared to human-alone and machine-alone

Experiments I and II use the same set of training samples and the same set of test samples. Experiment I is considered the human-alone experiment. The initial automatic recognition phase of experiment II reflects the performance of the machine alone. So comparing CAVIAR to human-alone and machine-alone is to compare the results of experiment II to the results of experiment I, and to the results of the initial automatic recognition of experiment II, respectively.

There are two critical aspects of the system performance,

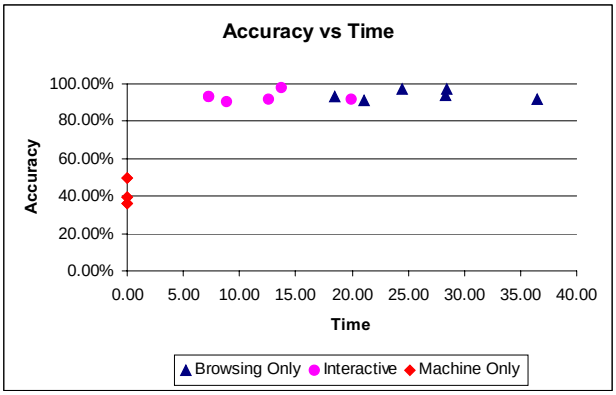

Figure 5: CAVIAR reduces the recognition time significantly compared to the human-alone (browsing only), and increases the accuracy significantly compared to machine-alone.

accuracy and time. The machine time depends on the hardware configuration of the machine and on the degree of software optimization. Since it is always much shorter than the human time, we ignore the machine time, and compare only the human time.

The first two rows of Table 1 and Figure 5 show the results. We observe that there are no obvious differences between CAVIAR and human-alone in accuracy. Every subject achieves above 90% accuracy. With the machine's help, the median time spent on each test sample is only half of the human-alone. On the other hand, the accuracy of the machine-alone is less than 50%. With a little human help (10 seconds average per flower), the accuracy increases to more than 90%. In summary, well-designed interactive recognition can halve the time compared to human-alone, and almost double the accuracy compared to machine-alone.

4.2 CAVIAR learning

Experiments III, IV, and V were designed to evaluate the machine's learning ability. These three experiments simulate the scenario where CAVIAR accumulates statistics as it is used. The same set of samples are used in Experiment V (1 ground truth and 4 pseudo per class) and Experiment II (5 ground truth per class).

From Table 1, we observe that 1) the accuracy of Experiment III, which has only one training sample, is still high (median 90%); 2) there is not much difference in accuracy among these four experiments. The median accuracies are all above 90%. However, the rank orders[5] of automatic

[5]The rank order is between 1 and the number of classes (102). If the correct candidate appears in the first place, the rank order is 1.

Table 1: System adaptation

Exp't	Accuracy (%)			Auto Rank Order			Time (s)		
I	93	91	92				18.4	21.1	36.5
	94	97	97				28.3	24.5	28.4
	Max: 97 Min: 91 Median: 93.6			—			**Max: 36.5 Min: 18.4 Median: 26.4**		
II	92	90	92	6.6	6.6	8.1	19.9	8.8	12.5
	93	93	98	8.1	6.3	6.3	7.3	7.2	13.8
	Max: 98 Min: 90 Median: 92.6			**Max: 8.1 Min: 6.3 Median: 6.6**			**Max: 19.9 Min: 7.2 Median: 10.7**		
III	85	98	90	14.6	14.1	12.1	27.1	20.3	15.1
	83	92	90	12.6	12.6	12.8	16.5	16.3	15.0
	Max: 98 Min: 83 Median: 90.2			**Max: 14.6 Min: 12.1 Median: 12.7**			**Max: 27.1 Min: 15.0 Median: 16.4**		
IV	91	95	92	10.5	11.0	8.4	10.3	13.6	8.4
	99	94	99	10.5	12.9	10.6	12.3	14.6	13.1
	Max: 99 Min: 91 Median: 94.6			**Max: 12.9 Min: 8.4 Median: 10.6**			**Max: 14.6 Min: 8.4 Median: 12.7**		
V	91	94	88	9.0	9.0	8.3	9.3	12.2	7.3
	92	90	92	8.3	8.6	8.6	13.7	10.4	10.9
	Max: 94 Min: 88 Median: 91.7			**Max: 9.0 Min: 8.3 Median: 8.6**			**Max: 13.7 Min: 7.3 Median: 10.7**		

recognition decrease from 12.7 to 8.6. This means that the performance of the machine improves by adding pseudo-training samples, although some pseudo-training samples are not correctly labelled; 3) in consequence of the improved automatic prediction, the time for interactive recognition decreases from 16.4 seconds to 10.7 seconds. This means that the improved performance of the automated components of CAVIAR does help the users to identify the flowers faster. 4) Both automatic rank order and time for complete interactive recognition are near the corresponding values of Experiment II, which, as expected, has the best performance. This suggests that instead of initializing the CAVIAR system with many training samples, we can trust the system's self-learning ability. Of course, the first users would need more time.

5. Conclusions

We proposed a parameterized geometrical model to mediate the communication between human and computer for interactive visual object recognition. We modelled the recognition procedure as a finite state machine. We demonstrated the methodology with a flower recognition system. Based on the evaluation of the system, we claim that: 1) A parameterized visible model leads to effective human-computer interaction; 2) Human intervention, especially in the early segmentation stage and at final identification, is valuable; 3) Calculating features and culling unlikely candidates are appropriate tasks for the machine; 4) CAVIAR can, in some situations, outperform human-alone and machine-alone; 5) the system can be initialized with a minimum number of training samples, but still achieve high accuracy; 6) the system shows self-learning ability.

Model-mediated visual interactive recognition poses many exciting research challenges and may also have practical applications. We have begun to study the user's cognitive state transitions through examination of our detailed log of the timing of the interactions and through eye tracking with an ASL GazeTracker. With a mobile (Sharp Zaurus) implementation developed at Pace University [13], we will investigate the benefits of obtaining additional photographs (e.g., of pistils and leaves). Candidate applications include faces, fruit, and photographs of skin diseases.

References

[1] H.S. Baird and G. Nagy, "A Self-Correcting 100-Font Classifier," *Proc. of SPIE, Document Recognition*, vol. 2181, pp. 106-115, February 1994.

[2] R.O. Duda, P.E. Hart, and D.G. Stork, *Pattern Classification*, Wiley, 2000.

[3] I. Haritaoglu, "Scene Text Extraction and Translation for Handheld Devices," *Proc. of IEEE conf. on Computer Vision and Pattern Recognition*, vol. 2, pp. 408-413, December, 2001.

[4] R. Jain, *US NSF Workshop on Visual Information Management Systems*, 1992.

[5] A.C. Kak and G.N. Desouza, "Robotic Vision: What Happened to the Visions of Yesterday," *Proc. of Int. Conf. Pattern Recognition 2002*, vol. 2, pp. 839-847, 2002.

[6] R. J. Mericsko, "Introduction of 27th AIPR Workshop - Advances in Computer-Assisted Recognition," *Proc. of SPIE*, vol. 3584, October 1998.

[7] G. Nagy and G.L. Shelton Jr., "Self-Corrective Character Recognition System," *IEEE Trans. Information Theory*, vol. IT-12, No. 2, pp. 215-222, 1966.

[8] Y. Rui, T.S. Huang, M. Ortega, and S. Mehrotra, "Relevance Feedback: A Power Tool for Interactive Content-Based Image Retrieval," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 8, No. 5, pp. 644-655, 1998.

[9] J. Yang, X. Chen, and W. Kunz, "A PDA-based Face Recognition System," *Proc. of the 6th IEEE Workshop on Applications of Computer Vision*, pp. 19-23, December, 2002.

[10] J. Zhang, X. Chen, J. Yang, and A. Waibel, "A PDA-based Sign Translator," *Proc. of the 4th IEEE Int. Conf. on Multimodal Interfaces*, pp. 217-222, 2002.

[11] J. Zou, "A Procedure for Model-Based Interactive Object Segmentation," *submitted to ICPR04*, 2004.

[12] J. Zou, *Computer Assisted Visual InterActive Recognition*, Ph.D. thesis, ECSE department, Rensselaer Polytechnic Institute, May, 2004.

[13] http://utopia.csis.pace.edu/cs615/2002-2003/team2/, last accessed on Nov. 14, 2003.