

Smart Shopper: An Agent-Based Web-Mining Approach to Internet Shopping

James Liu, *Member, IEEE*, and Jane You, *Member, IEEE*

Abstract—With the rapid development of e-commerce, there is an immediate need for an autonomous and robust system that can offer decision-making support for customers who are looking for the cheapest, the most familiar, or the best-quality product. This paper presents an agent-based web-mining approach to Internet shopping. In contrast to the traditional multiagent systems, which use rule-based or case-based process flows to coordinate communications for system automation, we propose a fuzzy neural network to tackle the uncertainties in practical shopping activities, such as consumer preferences, product specification, product selection, price negotiation, purchase, delivery, after-sales service and evaluation. The fuzzy neural network provides an automatic and autonomous product classification and selection scheme to support fuzzy decision making by integrating fuzzy logic technology and the back propagation feed forward neural network. In addition, a new visual data model is introduced to overcome the limitations of the current web browsers that lack flexibility for customers to view products from different perspectives. Such a model also extends the conventional data warehouse schema to deal with intensive data volumes and complex transformations with a high degree of flexibility for multiperspective visualization and morphing capability in an interactive environment. Furthermore, an agent development tool named “Aplet” is used as a programming framework for system implementation. The integration of dynamic object visualization, interactive user interface and data mining decision support provides an effective technique to close the gap between the “real world” and the “cyber world” from a business perspective. The experimental results demonstrate the feasibility of the proposed approach for web-based business transactions.

Index Terms—Feature selection, fuzzy neural network, Internet shopping, multiagent, web-mining.

I. INTRODUCTION

THE rapid growth of the Internet has stimulated the emergence of electronic commerce (e-commerce). Global economic forces have created an environment that fosters electronic commerce. Technology advances in the computer industry have led to the rapid development of e-commerce applications. Nowadays, e-commerce is playing an important role in reducing costs, improving product quality, reaching new customers or suppliers, and creating new ways for selling existing products. Such a rapidly growing e-commerce environment involves communications, business transactions, services and online processings [26]. According to [43], e-commerce is characterized as an interdisciplinary area which covers a

wide range of issues including security, trust, law, payment mechanisms, advertisement, online catalogs, multimedia shopping experiences, and intermediaries. However, a significant barrier to effective exploitation of these web-based business activities has been created by a lack of powerful visualization techniques. To simulate and implement the virtual shopping activities in the cyber world, one of the major concerns is to provide the customer with the “touch-and-feel” of the product they are looking at, ranging from small items such as books or casual-wear, to more expensive merchandise such as cars or furniture. The relevant technology involves a multiperspective three-dimensional (3-D) visualization of the objects, morphing, and object distortion techniques based on user interactions. Unfortunately, the existing techniques for multiperspective 3-D visualization and morphing are mainly focused on “pixel-based” image transformations [13], [17] which involve intensive computational effort and lack flexibility for object modeling. Thus, it is vital to introduce flexible models with sophisticated feature patterns [34], [11] for web browsing and object visualization. In addition, it is essential to develop supporting software tools to achieve portability with simple data models in a client-server environment. Although the existing virtual reality modeling language (VRML) [3] provides a feasible solution for 3-D object modeling in the Internet environment, its application is limited.

In this paper, we present a new visualization approach to provide intelligent web browsing support for electronic commerce using data warehousing and data mining techniques. To overcome the limitations of the current web browsers, which lack flexibility for customers to visualize products from different perspectives, a new visual data model is introduced, which extends the conventional data warehouse schema. This model is able to deal with intensive data volumes and complex transformations with a high degree of flexibility in terms of multiperspective visualizations and morphing capacity in an interactive environment. In addition, a fuzzy decision strategy is proposed to guide the retrieval and evaluation of similar products that are used in further business actions. The integration of dynamic object visualization, an interactive user interface and a flexible evaluation scheme provides an effective approach to close the gap between the “real world” and the “cyber world” from a business perspective.

To handle a huge amount of business information effectively and efficiently, there is a need for an autonomous and robust system that can offer decision-making support for customers who are looking for the cheapest, the most familiar, or the best-quality product. Agent technology, which is characterized by its automatic delegation of tasks, and its autonomous and highly mobile components in the web environment, is emerging as an

Manuscript received December 30, 2000; revised October 7, 2002. This work was supported in part by the the Hong Kong Polytechnic University under University Research Grant G-T375 and under Department Research Grant H-ZJ87.

The authors are with the Department of Computing, The Hong Kong Polytechnic University, Kowloon, Hong Kong (e-mail: csnkliu@comp.polyu.edu.hk).

Digital Object Identifier 10.1109/TFUZZ.2003.809900

effective approach for the provision of such a system for e-commerce. In a mobile e-commerce system such as Internet shopping via a WAP phone (namely, m-shopping), we incorporate mobile agent technology based on Aglets, which provides the agent framework to allow both the web-based e-shopping and m-shopping to operate simultaneously. A brief description of Aglets is given in Section VI-C.

Another key issue in online shopping is to understand consumer behavior. The current research suggests that such an issue should be addressed by the integration of technological, business and behavioral perspectives [41]. There have been several models used to describe consumer buying behavior such as the Blackwell [16], the Bettman [7], and the Howard–Sheth [21] models. All of these models can be summarized by the following six fundamental stages of consumer buying behavior: 1) consumer requirements definition; 2) product brokering; 3) merchant brokering; 4) negotiation; 5) purchase and delivery; and 6) after-sales services and evaluation. These six stages can be implemented by using agent technologies to facilitate the web-based consumer shopping activities for e-commerce [42]. Such agent-based systems include *AuctionBot* in a C2C e-auction system [5], *BargainFinder* and *Jango* in a B2C e-shopping system [6], [23], and *MAGNET* in a B2B e-supply chain system [12].

It is noted that the existing agent-based product brokering systems are either case-based or constraint-satisfaction based in their decision making. However, a wide range of uncertainties and probabilities exist during the first three stages of the consumer buying behavior model. It is very difficult to consider all of the possible cases or constraint conditions required to make an accurate decision. Unlike traditional multiagent systems, which use rule- or case-based process flows to coordinate communications for system automation, we propose a fuzzy neural network to tackle the uncertainties in practical shopping activities. The fuzzy neural network provides an automatic and autonomous product classification and selection scheme, and supports fuzzy decision making by integrating fuzzy logic technology and the back-propagation feed forward (BPFF) neural network. An agent development tool named “Aglet” is used as a programming framework for system implementation.

The rest of this paper is organized as follows. Section II describes the use of a data warehouse schema to integrate multiple object features for visual information representation. The concept of an elastic graph dynamic link model (EGDLM) for dynamic object manipulation is introduced in Section III, and the use of fuzzy logic for a fuzzy neural network to support decision making is discussed in Section IV. Section V highlights an agent-based fuzzy shopper system and its implementation strategy. The experimental results are reported in Section VI. Finally, conclusions are given in Section VII.

II. VISUAL DATA WAREHOUSE

Data warehousing is viewed as a technology that not only functions as a data superstore, but also processes data to create a data warehouse, an operational data store, or a data mart stored on traditional servers, Intranet servers, or Internet servers. In

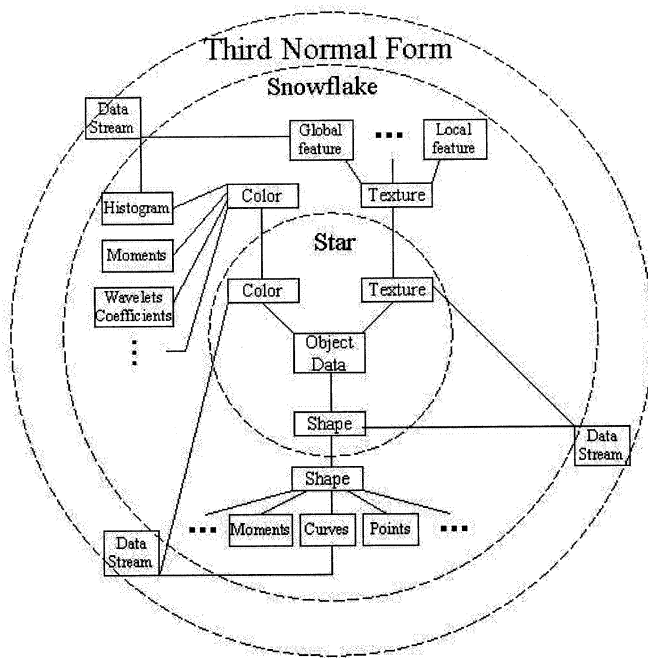


Fig. 1. Starflake schema for visual object representation.

other words, data warehouses are not just large databases; they are large and complex environments that integrate many technologies. A major advantage of using a data warehouse is being able to store data at different levels of granularity along different dimensions such as data type, time, etc. Thus, data warehouse systems require a lot of maintenance and management.

Developing a data warehouse is similar to other software projects. The schema is initially developed for the database design. It is a representation of a database, which gives semantics to the data to model the database. In relational databases, there are two types of schema – conceptual schema and logical schema. A conceptual schema often refers to an entity-relationship diagram, and a logical schema represents the structure and connection between the relational tables. In object-oriented databases, conceptual models are usually used. A schema is also used to construct a data warehouse. Facts and dimensions are two basic concepts related to a data warehouse schema.

Although the concept of a data warehouse originated from traditional business information systems, it can be extended to visual information systems. In this paper, we focus on establishing a visual data warehouse for web-based dynamic object visualization and manipulation for e-commerce applications. The description of an object is regarded as fact data in a visual data warehouse, and multiple object features constitute dimension data. In a general description of a visual object, features such as the color, texture, and shape are considered to be dimension data. Color can be further represented by a color histogram, color moments and color component features. Texture can be further described by its global and local statistical measurements or structural semantics. Shape can be further decomposed into a representation of surface moments, boundary curves and salient points. Fig. 1 illustrates a starflake schema that shows the relationship between fact data and

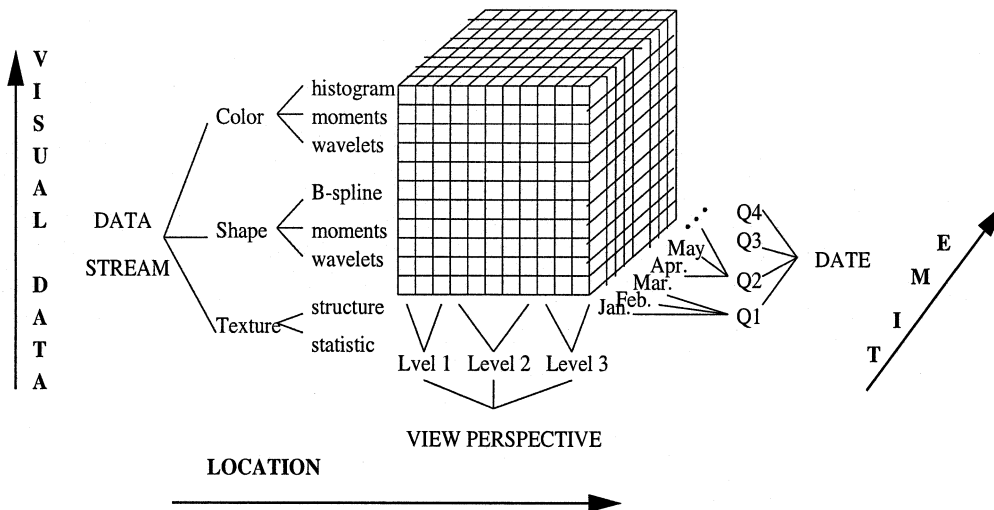


Fig. 2. Multidimensional data cube.

TABLE I
THREE MAJOR IMAGE FEATURE CATEGORIES FOR CONTENT DESCRIPTION

Color	Texture	Shape
histogram	structural description	moments
moments	statistical description	B-spline
wavelet coefficients	wavelets	wavelets
...

dimension data [4], where the center table represents the fact table (object description) and the surrounding (reference) tables are dimension tables (individual object features).

It is noted that, in real life it may be difficult to restructure all of the components within a visual data model into a set of distinct dimensions. In other words, there are situations where there are several components or relationships that span one or more dimensions. To more effectively represent the data, a starflake schema allows a degree of crossover between dimensions. In our proposed system, the fact table defines three major data streams of visual information, namely color, texture and shape. Using the hierarchies of the visual data streams, a multidimensional data cube can be used to visualize the visual information and its OLAP (online analytical processing) data, where the cells hold the quantifying data (often referred to as facts), while the qualifying data describe the axes of the cube and can be used for addressing individual cells or groups of cells. Such a multidimensional data cube is shown in Fig. 2, and Table I illustrates a typical image dimension table for content description.

To facilitate the management and decision support of the data warehouse, the above starflake schema is used, which uses a combination of denormalized star and normalized snowflake schemas. In these cases, a series of combinations of database views is created to allow user access tools to treat the fact table partitions as a single, large table for fast information retrieval. In addition, key reference data is structured into a set of dimensions that are referenced from the fact table. Each dimension is

stored in a series of normalized tables (snowflake), with an additional denormalized star dimension table [4]. The major features of the proposed data structure are summarized as follows:

- support multiple visual data streams using different dimensions;
- represent visual data in a hierarchical structure using normalized snowflake schemas at different levels;
- integrate multiple visual features using overlay between the dimensions;
- speed up visual data query and manipulation processing by data partitioning.

III. WAVELET-BASED VISUAL INFORMATION REPRESENTATION

In our proposed visual data model for the starflake schema in data warehousing, it is very important to represent the content of each image in the dimensions of the model, which are based on the selected feature components and the relevant measurements. In contrast to the traditional approaches that apply different techniques for different features [9], [35], we propose a unique scheme that extracts different features based on a wavelet transform. The following sections highlight three basic types of features: colors, textures and shapes.

A. Color Histogram Features

Color is an important feature of an image. Color histograms provide a global representation of the color in an image and they have been used in many content-based image retrieval (CBIR) systems. As reported in [31], such chromatic information can be extracted from histograms of successive approximation quantizations (SAQs) of the wavelet coefficients. The SAQ scheme progressively decomposes the wavelet coefficients into quantization layers according to their importance, and then it generates a sequence of histograms in accordance with the different resolutions. When a YUV color model is used, we extract the histograms from each of the Y-, U-, and V-subimages by applying a biorthogonal wavelet transform, followed by coefficient quantization and entropy coding.

B. Texture Features

Historically, structural and statistical approaches have been adopted to extract textural features [20]. The structural approach assumes that a texture is characterized by some graphical primitives that follow a certain rule for their placement in an image. In the statistical approach, texture is regarded as a sample from a probability distribution on the image space and it is defined by a stochastic model or characterized by a set of statistical features. Here, we adopt a wavelet-based approach to represent texture features [22]. When an image is decomposed into decorrelated subimages through a wavelet filter bank, each subimage represents the textural features at a different scale and orientation. Thus, the associated wavelet coefficients can be used to represent textures. If an image is decomposed into three wavelet layers, there will be ten subimages. In each sub-image, the standard deviation of the wavelet coefficients is calculated to represent a component of its texture feature. Consequently, ten standard deviations, corresponding to the ten subimages, are used as the texture representation of the image. The following expression represents this in its vector form:

$$\mathbf{F}_t = [f_{t1}, f_{t2}, \dots, f_{t10}].$$

C. Shape Features

The visual shape of an object is another important datum that is used in the real world. In general, it is difficult to uniquely describe an object's shape in terms of alphanumeric descriptors [36]. To capture the features of the shape, we propose to combine invariant moments, a representation of the object using B-spline curves and wavelet coefficients. There are two key issues involved: representation of an object from the original image using B-splines, and using moments as a description of the object at different scales and resolutions in the subimages that are created by the wavelet transform. B-splines are regarded as of the most efficient representations of a curve, and they possess very attractive properties such as spatial uniqueness, boundedness and continuity, controllability of the local shape, and invariance to affine transformations. Given a set of n ordered corner points $((x_1, y_1), (x_2, y_2), \dots, (x_n, y_n))$, our aim is to fit them with a real-valued B-spline curve. Previously, we proposed the idea of finding an approximate B-spline so that the measurement of the "error" is small between the observed data and the corresponding values of the B-spline curve. If we deal with a closed cubic B-spline with $m+1$ parameters, C_0, C_1, \dots, C_m , (as the control points), then the curve can be modeled as a linear combination of four cubic polynomials in the parameter t , where t is normalized between 0 and 1 ($0 \leq t \leq 1$) for each segment of the curve

$$r_i(t) = C_{i-1}Q_0(t) + C_iQ_1(t) + C_{i+1}Q_2(t) + C_{i+2}Q_3 \quad (1)$$

where

$$Q_k(t) = a_{k0}t^3 + a_{k1}t^2 + a_{k2}t + a_{k3}, \quad k = 0, 1, 2, 3.$$

To determine the error between the data for the curve and their corresponding B-spline points, we use the minimum mean square error estimation (MMSE). We determine the final B-spline representation of the given curve data by iteratively estimating the parameters of the B-spline. This representation of boundary of the object using a B-spline is used in the final retrieval of the output using curve matching when a query image is presented as either a sketch or as an image that contains clear boundaries for the objects that it contains.

It is known that the central moments, normalized according to the area, which are computed relative to the principal axis, are invariant under magnification, translation, and rotation of the object. Therefore, they are powerful measurements in the analysis of two-dimensional shapes. Important information about a shape, such as its size, center and orientation, are all attributes that are based on moments. A number of other useful features of the shape can also be derived from moments, such as maximum and minimum moments of inertia, moment invariants, diffusion of the shape and elongation of the shape. For a given image of size $M \times N$ pixels, $I, I(i, j)$ represents the gray scale of each individual image pixel at position (i, j) , its (p, q) th central moments are expressed as

$$\mu_{pq} = \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} (i - \bar{x})^p (j - \bar{y})^q I(i, j) \quad (2)$$

where

$$\bar{x} = \frac{\sum_{i=0}^{M-1} \sum_{j=0}^{N-1} i \times I(i, j)}{\sum_{i=0}^{M-1} \sum_{j=0}^{N-1} I(i, j)}$$

$$\bar{y} = \frac{\sum_{i=0}^{M-1} \sum_{j=0}^{N-1} j \times I(i, j)}{\sum_{i=0}^{M-1} \sum_{j=0}^{N-1} I(i, j)}.$$

The normalized central moments, denoted by η_{pq} , $p, q = 0, 1, \dots$, are defined as

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^\gamma} \quad (3)$$

where $\gamma = (p + q)/2 + 1, p + q = 2, 3, \dots$

In this paper, we apply a wavelet transform to decompose the original image into a collection of sub-band images that range from low to high resolution. The related first, second and third (normalized) central moments for each subimage are computed, and the average values of the same category of moment for all subband images are used as individual components of features of the shape. Thus, the vector of the image's shape features is defined as:

$$\mathbf{F}_s = [\eta_{01}, \eta_{10}, \eta_{11}, \eta_{02}, \eta_{20}, \eta_{03}, \eta_{12}, \eta_{21}, \eta_{30}].$$

IV. DYNAMIC OBJECT VISUALIZATION

A. Elastic Graph Dynamic Link Model (EGDLM)

The EGDLM is an integration of the active contour model (ACM) for “contour” pattern extraction and the dynamic link architecture (DLA) for the neural dynamics. The ACM [8], [28] provides an effective approach to link image boundary feature points using a continuous curve, the so-called “snake.” This snake will position itself along the feature points in a robust manner by stretching and bending, which is controlled by the parameters in its energy expression

$$E_{\text{int}}(u(s)) = \alpha(s) |u_s(s)|^2 + \beta(s) |u_{ss}(s)|^2 \quad (4)$$

where $u(s) = (x(s), y(s))$ is the snake curve, x and y represent the coordinate of each point on the curve, and s is the arc-length of the curve. The parameters of elasticity α and β control the smoothness of the snake curves.

The success of the extension of ACM to EGDLM is reported in [30]. One of the most important reasons for the use of EGDLM is the high degree of flexibility and elasticity of the object model under various transformations, distortions and morphing operations. In addition, the structural behavior is achieved based on the correlation properties of the feature vectors in the attribute graph topology. The merging of three-dimensional elastic attribute structural graphs (EASGs) for 3-D object display involves the integration of the two-dimensional (2-D)-EASGs from different perspectives into a 3-D model (3-D-EASG). This is based on the “elastic graph matching” used in the EGDLM model on overlapping regions of different 2-D-EASGs. More specifically, we make use of a 3-D-EASG to first extract the structural skeleton of a visual object from multiperspective views, and then it undergoes a series of transformations to model the object. Based on the elastic attribute graph modeling (EGM) approach, such a model can provide a high degree of flexibility in terms of multiperspective visualization and morphing capability in accordance with user interactions.

B. Optimal Object Selection via Fuzzy Compactness

The concept of a fuzzy set was first introduced by Zadeh [48] in 1965. In contrast to classical crisp sets which have sharp boundaries, a fuzzy set does not have precise boundaries to determine its members being definitely in or definitely out. In other words, a member of a fuzzy set may belong to the set to a greater or lesser degree. Therefore, each fuzzy set is uniquely determined by a particular membership grade function, which assigns to each object of interest its grade of membership in the set. Recent advances in fuzzy set technology and its applications are presented in a survey paper [19]. In the area of visual information processing, the identification of interesting object components for display and manipulation is based on the selection of a threshold which can be determined by applying fuzzy set theory. When a feature image is regarded as a fuzzy set \mathbf{X} of size $M \times N$ and L levels of feature measurements, we can extend Pal and Rosenfeld’s fuzzy compactness approach [40] to determine the optimal threshold value by minimizing the measurement of fuzziness. They introduce an index of fuzziness to represent the average amount of ambiguity (fuzziness) present

in an image I by measuring the distance between its fuzzy property $\mu_{\mathbf{X}}$ and the nearest two-level property $\mu_{\overline{\mathbf{X}}}$. A linear index of fuzziness is defined as

$$v_l(\mathbf{X}) = \frac{2}{MN} \sum_{i=1}^M \sum_{j=1}^N |\mu_{\mathbf{X}}(x_{ij}) - \mu_{\overline{\mathbf{X}}}(x_{ij})| \quad (5)$$

$$= \frac{2}{MN} \sum_{i=1}^M \sum_{j=1}^N \mu_{\mathbf{X} \cap \overline{\mathbf{X}}}(x_{ij}) \quad (6)$$

$$= \frac{2}{MN} \sum_{i=1}^M \sum_{j=1}^N \min\{\mu_{\mathbf{X}}(x_{ij}), (1 - \mu_{\mathbf{X}}(x_{ij}))\} \quad (7)$$

where $\mu_{\overline{\mathbf{X}}}(x_{ij})$ denotes the nearest two-level version of \mathbf{X} such that

$$\mu_{\overline{\mathbf{X}}}(x_{ij}) = 0, \quad \text{if } \mu_{\mathbf{X}}(x_{ij}) \leq 0.5 \\ = 1, \quad \text{otherwise.}$$

In contrast to the conventional measurement of fuzziness in a gray-scale image as detailed previously, we extend this approach by applying the distance measurement to a feature image rather than the measurement of the fuzziness of the image in gray scale. Accordingly, \mathbf{X} refers to the feature set and x_{ij} represents a certain feature measurement. Instead of processing the gray scale image, we consider the minimization of the fuzziness in an image feature to determine an appropriate threshold value for object component selection. Such an algorithm is summarized as follows.

- Step 1: Convert the original visual data (fact table in the visual data warehouse) to a feature image (dimension tables associated with the fact table) using its relevant measurement (color, texture or shape). The maximum and minimum values are l_{\max} and l_{\min} , respectively.
- Step 2: Construct the feature membership $\mu_{\mathbf{X}}$, where

$$\mu_{\mathbf{X}}(l) = S(l; a, l_i, c), \quad l_{\min} \leq l, l_i \leq l_{\max} \quad (8)$$

and

$$S(l; a, b, c) = 0 \quad l \leq a \\ = 2 \left[\frac{(l-a)}{(c-a)} \right]^2, \quad a < l \leq b \\ = 1 - 2 \left[\frac{(l-c)}{(c-a)} \right]^2, \quad b < l \leq c \\ = 1 \quad l > c$$

with cross-over point $b = l_i = (a + c)/2$ and bandwidth $\Delta b = b - a = c - b$. The details of how to determine a, b, c are given in [40].

- Step 3: Obtain the linear index of fuzziness of the image feature \mathbf{X} by computing

$$v_l(\mathbf{X}) = \frac{2}{MN} \sum_l T_i(l) h(l) \quad (9)$$

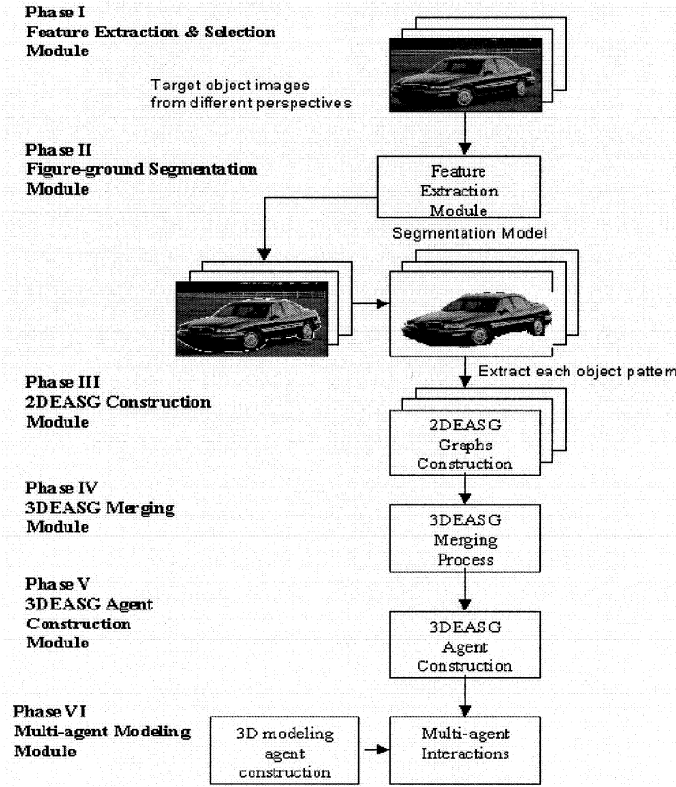


Fig. 3. Basic components of 3-D visualization.

where

$$T_i(l) = \min\{S(l; a, l_i, c), 1 - S(l; a, l_i, c)\} \quad (10)$$

and $h(l)$ denotes the number of occurrences of the level l .

- Step 4: Vary l_i from l_{\min} to l_{\max} and choose $l_i = l_c$ to correspond to the minimum of $v(X)$.

C. Flexible 3-D Visualization: Basic Components

To overcome the lack of flexibility of existing systems, we propose an agent-based system to facilitate the 3-D modeling and morphing functionality using a 3-D-EASG modeling technique. Our approach is to use an extension of our previous work on the 2-D-EGDLM for invariant vision object recognition, which has been extensively applied to different areas such as human face recognition, scene analysis, Chinese character recognition and the identification of tropical cyclones from satellite images [30]. Fig. 3 shows the basic components for 3-D dynamic visualization.

The following discussion summarizes the six major phases in the proposed system.

- **Phase I: Feature Extraction**

A feature extraction module is designed to extract vital features from 2-D images for the target object under different perspectives. Unlike most of the existing techniques, which rely on the detection of feature points in the resolution pyramid for object description, we apply a

wavelet-based hierarchical scheme [47] to decompose an image into a series of sub-band images, and then extract image interesting points for the generation of an image feature hierarchy.

- **Phase II: Figure-Ground Segmentation**

To segment the target objects from the scene image, we extend the traditional neural oscillator models [37] to construct a composite neural oscillator image segmentation model. Unlike the traditional models, in our model an oscillator with common inhibitory neurons is employed as a safeguard against phase-locking of the composite neural oscillators that stimulate the visual cortex of the color image perceptions. Each color image is broken into a 2-D mesh of composite neural oscillator sites. Each site consists of a number of oscillators which denote the neural oscillation from each local feature response (obtained from Phase I). To further improve the self-organization rate, a near-neighbor scheme is adopted for neuron oscillations. The segmentation criterion, which is a measurement of the binding strength (phase relationship) between the composite neural oscillators and their nearest neighbors [37], is governed by correlation factors.

- **Phase III: The Construction of the 2-D Elastic Structural Graph**

The construction of the 2-D-ESG is based on the framework of EGDLM for the attribute graph construction technique. One of the most important reasons for the use of EGDLM is the high degree of flexibility and elasticity of the object model under various transformations, distortions and morphing operations. In addition, the structural behavior is achieved based on the correlation properties of the feature vectors in the attribute graph topology.

- **Phase IV: The Merging of the 3-D Elastic Attribute Structural Graph**

The merge involves the integration of the 2-D EASGs from different perspectives to create a 3-D model (3-D-EASG), which is based on the elastic graph matching used in the EGDLM model applied to overlapping regions of different 2-D-EASGs. The innovative application of an EGDLM technique for 3-D modeling is further demonstrated by facilitating mobility and distributed computing functionality in the Internet environment.

- **Phase V: 3-D-EASG Agent Construction**

This phase focuses on the construction of the 3-D-EASG as a mobile agent, which can be embedded as a fully independent entity and transported to the client machine dynamically and on demand. The Internet offers the possibility of access to information and applications anywhere, any time and from any device [38]. We need a platform for mobile coding and programming so as to provide critical infrastructure for developing, deploying, and maintaining mobile computing solutions. In our system, we use the Ara platform for the portable and secure execution of mobile agents in heterogeneous networks.

- **Phase VI: Multiagent Interactions for 3-D Visualization**

This phase deals with multiagent interactions between 3-D visualization and modeling agents and 3-D-EASG

agents for the target object with user interactions. The main focus of this phase is to construct the 3-D visualization and modeling agent for the manipulation of the 3-D-EASG agents in relation to user interactions, which include various transformations such as translation, rotation, dilation, contraction, distortion and morphing process from one 3-D-EASG agent to another. This multiagent approach should outperform other concurrent technologies.

V. ONLINE DECISION SUPPORT: FUZZY NEURAL NETWORK

A. Background

Although fuzzy set theory was originally intended to solve problems associated with complex manufacturing systems using simple models, it has played an essential role in the development of intelligent systems. The combination of fuzzy set theory with neural network computation, genetic algorithms, evolutionary computation, and other advanced methodological tools has laid the foundation for a diversity of successful applications. These tools have been applied in areas such as expert systems, database and information retrieval systems, pattern recognition and clustering, signal and image processing, speech recognition, risk analysis, robotics, medicine, psychology, chemistry, ecology, and economics. A comprehensive survey of the applications of fuzzy set theory is presented in [29].

In this paper, we develop a fuzzy decision support system for Internet shopping. In contrast to traditional decision support systems, fuzzy systems deal with imprecise terms and provide decision support based on the fuzziness measurement of set members [14]. For example, if users would like to buy a pair of shoes, they need to specify their requirements for the product. If both size and fitness are used as the measurement for the shoes, we often use imprecise terms for this judgment. To describe size, instead of using the exact measurement, we may use the term “very large,” “large,” “quite large,” or even “medium” and “small.” Similarly, we may also use the term “fit,” “loose fit,” or “tight fit” to describe the fitness. This type of inaccurate description and uncertainty can be measured in terms of fuzziness, which is represented by fuzzy membership functions. Fig. 4 illustrates the definition of a fuzzy membership function to measure the degree of fitness. As shown in Fig. 4, for example, if “Degree of Fitness” is divided into a 10 point scale, a 7.3 degree of fitness can be interpreted as 0.64 of “Fit” and 0.32 of “Tight.” Based on the same argument, a customer requirement of “7.3 degree of fitness and 6.0 degree of size” can also be fuzzified as a combination of fuzzy variables including “fitness” and “size” in a numerical format. Consequently, the membership functions for fitness take values on [0,1] as defined later, where x represents the degree of fitness

$$\begin{aligned} \underline{\text{fitness}} &= \left\{ \text{Loose, fit, Tight} \mid \forall \mu_{\underline{\text{Fitness}}}(\text{Loose}), \right. \\ &\quad \left. \mu_{\underline{\text{fitness}}}(\text{Fit}), \mu_{\underline{\text{fitness}}}(\text{Tight}) \in [0, 1] \right\} \\ \mu_{\underline{\text{fitness}}}(\text{Loose}) &= \begin{cases} 1, & \text{if } 0 \leq x \leq 3 \\ 4 - x, & \text{if } 3 < x \leq 4 \\ 0, & \text{otherwise} \end{cases} \end{aligned}$$

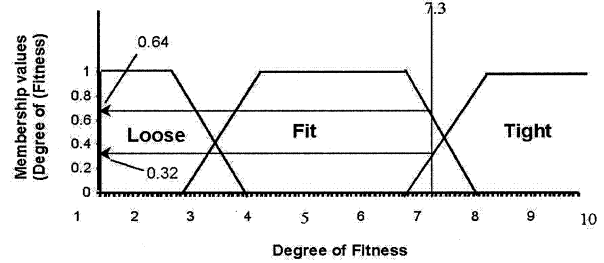


Fig. 4. Fuzzy membership functions for the degree of fitness.

$$\begin{aligned} \mu_{\underline{\text{fitness}}}(\text{Fit}) &= \begin{cases} x - 3, & \text{if } 3 \leq x \leq 4 \\ 1, & \text{if } 4 < x \leq 7 \\ 8 - x, & \text{if } 7 < x \leq 8 \\ 0, & \text{otherwise} \end{cases} \\ \mu_{\underline{\text{fitness}}}(\text{Tight}) &= \begin{cases} 1, & \text{if } x > 8 \\ x - 7, & \text{if } 7 \leq x \leq 8 \\ 0, & \text{otherwise} \end{cases} \end{aligned}$$

B. Structure

Fuzzy systems require the specification of fuzzy sets and the relevant fuzzy membership functions. However, these two key issues are traditionally predefined by the system designers (experts). Consequently, such a fuzzy system would lack flexibility to handle unspecified cases. To equip a fuzzy system with learning capabilities, an alternative approach is to integrate fuzzy sets with other learning models. Many researchers have proposed the use of neural networks in fuzzy systems: Wong and Wang have proposed a FuzzyNet model [46], Enbutsu *et al.* develop a hybridization scheme to extract fuzzy rules from a multilayered neural network [15], and Janikow suggests the use of a genetic algorithm for a learning fuzzy controller [24]. Previously, we have applied fuzzy neural networks to automatic machine maintenance and rainfall forecast [32], [33]. To tackle the uncertainties in practical shopping activities such as consumer requirements, product specification, product selection, price negotiation, purchase and delivery, after-sales service and evaluation, we integrate fuzzy set theory with the back propagation feed forward (BPFF) neural network. This achieves flexibility and robustness for automatic and autonomous decision support. Fig. 5 shows the general structure of the proposed FFBP network for product selection. The network consists of two parts: a fuzzy module and a feed forward back propagation (FFBP) neural network module. The fuzzy module provides the network with a collection of fuzzy variables as input nodes. The FFBP neural network is a multilayer neural network that is responsible for product selection based on a pattern classification scheme.

VI. SMART SHOPPER—AN AGENT-BASED INTERNET SHOPPING SYSTEM

A. Agent Technology and E-Commerce Applications

The Internet provides an ideal platform to support web-based e-commerce. The current Internet-based systems are mainly based on a client and server architecture, where all transactions are made by many request/response interactions over the

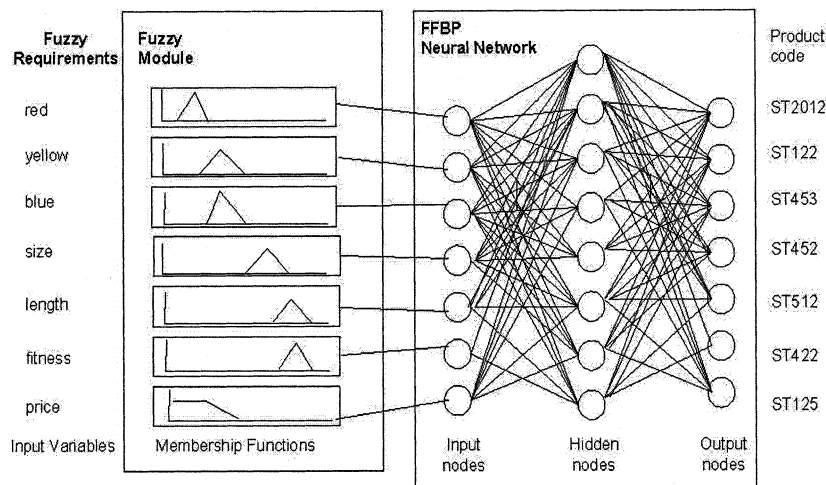


Fig. 5. Fuzzy forward back propagation network for product selection.

network. However, a user may experience communication difficulties with a long response time. To reduce the communication overhead, a mobile agent-based approach is more appropriate for e-commerce applications. A mobile-agent-based system uses different techniques to send a mobile software agent to a remote system to conduct multiple interactions with the software embedded on the remote system. The outputs are sent back to the user upon completion of the interaction. An agent can also interact with other agents on the Internet before returning to its original system. Therefore, mobile-agent-based systems are more flexible for web-based e-commerce applications than the existing client/server systems. The current agent technologies include IBM Aglets [2], ObjectSpace Voyager Agents [44], FTP Software Agent [18], General Magic Odyssey Agent System [39], Java Agent Template Lite (JATLite) [25], and Agent Builder Environment from IBM [1].

There are several advantages of using mobile agent technologies for e-commerce applications. The following text summarizes the major issues.

- Network load:

Distributed systems rely heavily on communication protocols for multiple interactions for a given task. By contrast, mobile agents reduce the load by packing a conversation and dispatching it to a destination host where the interaction can take place locally.

- Network latency:

E-commerce applications such as online shopping, price negotiation, and e-auction require real-time interactions. The normal network does not meet such real-time requirements because of its significant network latencies. However, mobile agents can overcome this limitation because they can be dispatched from a central controller to function locally, and directly execute the controller's operations.

- Asynchronous and autonomous issues:

Traditional mobile applications rely on expensive and fragile network connections. Therefore, it is not feasible to keep the connection open between mobile device and the selected network site for a certain task. Nevertheless,

mobile agent technology facilitates asynchronous and autonomous operations by embedding tasks into mobile agents and dispatching them back into the network when they complete a remote activity.

- Flexibility:

One of the major advantages of agent technology is the dynamic adjustment of an agent's operations with respect to its execution environment, and its ability to react to the changes autonomously. Therefore, multiple mobile agents can distribute themselves among the hosts of a network so as to maintain an optimal configuration for any Internet-based operations.

- Heterogeneous issues:

In general, network computing is fundamentally heterogeneous from both hardware and software perspectives. Since mobile agents are computer and transport-layer independent, they are dependent on their execution environment only. Consequently, mobile agent-based systems are positioned for seamless system integration.

- Robustness and fault-tolerance:

Robustness and fault-tolerance are two crucial issues for any Internet-based e-commerce applications. The nature of the flexibility and dynamic adjustment to the execution environment makes it feasible for mobile agents to provide a solid base for the development of robust and distributed e-commerce systems.

B. System Framework

The proposed Smart Shopper system consists of the following six modules:

- 1) customer requirements description (CRD);
- 2) requirement fuzzification scheme (RFS);
- 3) fuzzy agents negotiation scheme (FANS);
- 4) fuzzy product selection scheme (FPSS);
- 5) product defuzzification scheme (PDS);
- 6) product evaluation scheme (PES).

The relationship between these six modules is shown in Fig. 6. The definition and implementation details of each module are given in Section VII.

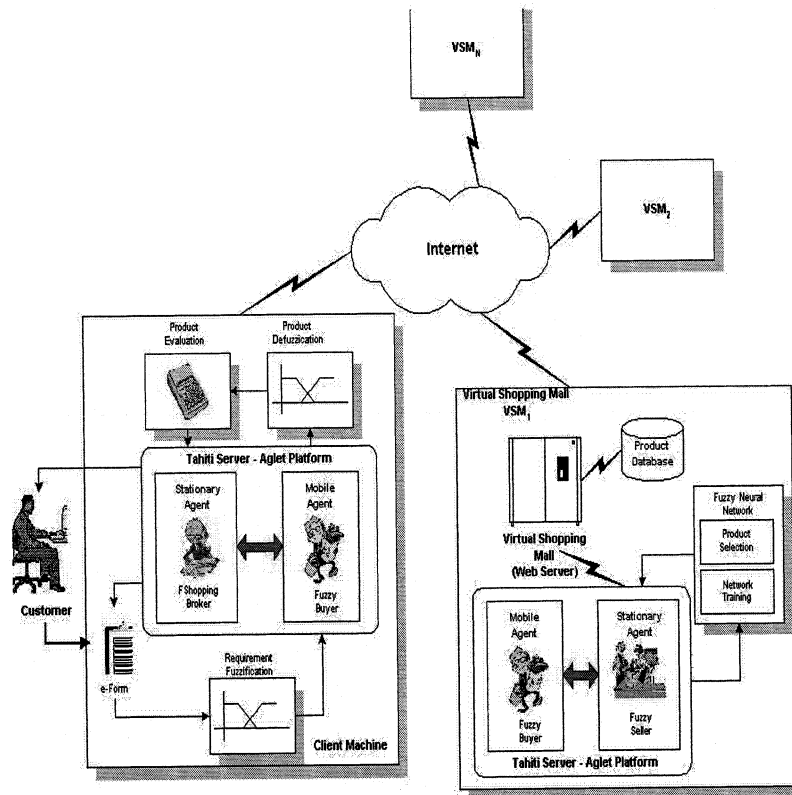


Fig. 6. System framework of Smart Shopper.

C. System Platform

IBM Aglets is used as the system platform for the implementation of our Smart Shopper system. IBM Aglets is a Java-based framework for implementing mobile agents [2] with the following features: 1) it provides an object-oriented programming interface; 2) it offers mechanisms for moving codes, data and state information from one machine to another; 3) it creates a platform-independent development and runtime environment; and 4) it provides security mechanisms.

The basic functionalities and runtime properties of Aglets are defined by the Java Aglet, AgletProxy, and AgletContext classes. The abstract class Aglet defines the fundamental methods that control the mobility and life cycle of an aglet. It also provides access to the inherent attributes of an aglet, such as creation time, owner, code-base, and trust level, as well as dynamic attributes, such as the arrival time at a site and the address of the current context.

The main function of the AgletProxy class is to handle access to the aglet. All of the communication among aglets are performed through their proxies. In addition, AgletProxy also provides location transparency by forwarding requests to remote hosts and returning results to the local host.

The AgletContext class provides the runtime execution environment for an aglet within the Tahiti server. When an aglet is dispatched to a remote site, it is detached from the current AgletContext object, serialized into a message bytestream, sent across the network, and reconstructed in a new AgletContext. As a result, the same execution environment will be created at the remote site.

Security is another critical component of the Aglet environment. Aglets provide a security model in the form of an AgletSecurityManager, which is a subclass of the "standard" Java SecurityManager. This software component protects the user from the threat of hostile code. This is because the class files for an applet are automatically downloaded when a user goes to the relevant web page in a browser. It is likely that a user will encounter applets from untrusted sources. Without any security, this would be a convenient way to spread a virus. Thus, the Aglet environment's security mechanisms help make it suitable for networks because they establish trust in the safety of network-mobile code.

VII. IMPLEMENTATION STRATEGIES

The Smart Shopper system is implemented by the use of IBM Aglets in both Linux and Unix platforms. This section summarizes the implementation details of the six modules of the system.

- CRD

There are two types of agents in this module: the FShopping Broker and Fuzzy Buyer.

- FShopping Broker

A stationary agent that acts as a buyer broker on behalf of the customers. This is an autonomous software agent that contains all of the necessary information (such as e-forms and analytical tools) for fuzzification/defuzzification and product evaluation.

- Fuzzy Buyer

A mobile agent that acts as a virtual buyer in the virtual marketplace. It coordinates all agent communication, interaction and negotiation operations.

In CRD, customers use an e-form provided by the FShopping Broker to describe their requirements via the browser. Fshopping Broker also provides a set of “cues” to facilitate product description.

- RFS

The FShopping Broker converts the customers’ inaccurate descriptions (e.g., color, size, style, fitness) into fuzzy variables based on the “embedded” knowledge in terms of its membership functions. The FShopping Broker also performs input data checks and validation. The following examples of fuzzy membership functions are used for shoe selection in terms of color measurement, where x refers to a color value within the range 0 to 192:

$$\begin{aligned} \text{Color}(\text{color}) &= \{\text{red, yellow, blue}\} \\ \underline{\text{color}} &= \left\{ \text{Light, Normal, Deep} \mid \forall \mu_{\underline{\text{color}}}(\text{Light}), \right. \\ &\quad \left. \mu_{\underline{\text{color}}}(\text{Normal}), \mu_{\underline{\text{color}}}(\text{Deep}) \in [0, 1] \right\} \\ \mu_{\underline{\text{color}}}(\text{Light}) &= \begin{cases} 1, & \text{if } 0 \leq x \leq 64 \\ \frac{96-x}{32}, & \text{if } 64 < x \leq 96 \\ 0, & \text{otherwise} \end{cases} \\ \mu_{\underline{\text{color}}}(\text{Normal}) &= \begin{cases} \frac{x-64}{32}, & \text{if } 64 \leq x \leq 96 \\ 1, & \text{if } 96 < x \leq 160 \\ \frac{192-x}{32}, & \text{if } 160 < x \leq 192 \\ 0, & \text{otherwise} \end{cases} \\ \mu_{\underline{\text{color}}}(\text{Deep}) &= \begin{cases} 1, & \text{if } x > 192 \\ \frac{x-160}{32}, & \text{if } 160 \leq x \leq 192 \\ 0, & \text{otherwise} \end{cases} \end{aligned}$$

- FANS

The Fuzzy Buyer starts its buying activities as soon as all of the relevant information has been collected from the customer and represented in fuzzy functions. To speed up the buying process by parallelism, the Fuzzy Buyer can act as a virtual salesperson in the shopping activities.

- FPSS

The Fuzzy Seller collects all of the customer’s requirements and uses a fuzzy neural network to perform product selection. The general structure of such a fuzzy neural network is briefly described in Section VI-A.

- PDS and PES

The PDS and PES are the last two modules of the Smart Shopper system which provide recommendations on products for customers based on the results from FPSS. To aid the user in understanding the system output, PDS maps the fuzzy decision results from FPSS, which is in the form of fuzziness membership measurements, to its fuzzy description in terms of the product attributes. PES is an objective evaluation scheme that provides the final ranking of all of the recommended products. The results are displayed on the client machine via the Fuzzy Buyer and the customer makes the final decision using the client browser.

VIII. EXPERIMENTAL RESULTS

The prototype Smart Shopper system was developed using IBM Aglets on both Linux and Unix platforms. An e-catalog

TABLE II
SYSTEM EFFECTIVENESS EVALUATION BY DM

Product Category	Degree of Match DM (%)
T-shirt	81
Shirt	78
Shoes	89
Trousers	88
Skirt	65
Sweater	81
Tablecloth	85
Napkin	86
Average Score	81.6

was generated based on a product database, which consists of more than 200 items under eight categories (T-shirt, shirt, shoes, trousers, skirt, sweater, tablecloth, and napkins). It is noted that, most of the current e-shopping agent systems deal with items such as books and music which can be described precisely with specific terms. In contrast, we choose items which are difficult to describe precisely so as to test the power of the fuzzy measurements. The neural network was well trained using the training set which covers all of the product types in the e-catalog. For the current e-catalog, eight different fuzzy neural networks were constructed, corresponding to the eight categories in the catalog.

To evaluate system effectiveness and efficiency, two types of experiment were conducted: a product selection (PS) test and a round-trip time (RTT) test. The PS test is to judge whether the system recommendations suit customer’s personal preferences. A group of 40 candidates were invited to participate in this experiment. During the test, each candidate bought one product from each category based on their own preference requirements. The candidates browsed the e-catalog to choose a list of their five “best choices” (L) that best matched their requirements. For evaluation, this list is compared with the recommendation list generated by the Smart Shopper. The degree of match between the system output and candidate’s individual preferences is measured by DM , which can be calculated as

$$DM = \frac{\sum_{i=1}^5 i \times m}{15} \quad \text{where } m = \begin{cases} 1, & \text{if } i \in L \\ 0, & \text{otherwise} \end{cases}$$

To evaluate the performance of the system, scores in the range of 1–5 are assigned to the “correct match” of the candidate’s five “best choices” in ranked order. For example, if the candidate’s choice of product with rank 1, 2, 3, and 5 is found in the recommendation list from Smart Shopper, the degree of match (DM) would be 73%, which is the sum of 1, 2, 3, and 5 divided by 15. Table II lists the test results for different product categories. The results indicate that the performance of the Smart Shopper is highly dependent on the “variability” (or “fuzziness”) associated with the merchandise. We found that the higher the inaccuracy (fuzziness) in expressing preference, the lower the degree

TABLE III
SYSTEM EFFICIENCY EVALUATION—AVERAGE RTT FOR 100 TRIALS

Server Location	Execution Time (ms)	
	Server 1	Server 2
A. Client Machine		
CRD		
RFS	310	305
B. Server Machine		
FANS	320	2015
FPSS	4260	4133
C. Client Machine		
FDS	320	330
PES	251	223
TOTAL RTT	5461	7006

of match found by the system. In reality, the final decision made by the real customer also varies case by case.

The system efficiency is judged using the RTT test. Instead of calculating the difference between the arrival and departure time to/from the server, the RTT test uses the total round trip time for the agents involved. RTT is determined from all of the fragments of time spent on each component for the various operations, starting from the collection of user requirements, and continuing with fuzzification, product selection, and evaluation. Two Tahiti servers are used in this test. Server 1 is located on the same local area network (LAN) as the client machine, while Server 2 is situated at a remote site within the campus. Table III shows the average execution time at different stage for 100 trials. It is noted that most of the execution time is spent on the FPSS). In practice, the network traffic should be considered for real applications.

IX. CONCLUSION

This paper explores the integration of visualization techniques, agent technology and a fuzzy neural network to provide intelligent and “virtual reality” support for today’s fast growing e-commerce sector. To overcome the limitations of the current web browsers, which lack the flexibility needed for customers to visualize products from different perspectives, we propose an intelligent browsing system to simulate “actual” shopping activities in the cyber world based on user interactions. Unlike the existing technology for object visualization and morphing in relation to image processing techniques, the proposed system uses fuzzy set theory and adopts a 3-D elastic attribute structure graph to deal with intensive data volumes and complex transformations with a high degree of flexibility in an interactive environment. The development of a fuzzy neural network, which combines fuzzy logic technology with a back propagation feed forward neural network, provides an effective method for tackling the uncertainties in practical shopping activities. With respect to the need for an autonomous, mobile and robust system that can offer decision-making support for customers performing Internet shopping, the preliminary results from the Smart Shopper experiments are promising. The proposed techniques should be feasible for other e-commerce applications.

ACKNOWLEDGMENT

The authors would like to thank the two anonymous reviewers for their constructive advice on the revision of the manuscript, which helped to improve this paper.

REFERENCES

- [1] ABE [Online]. Available: <http://www.networking.ibm.com/iag>
- [2] Aglets [Online]. Available: <http://www.trl.ibm.co.jp/aglets>
- [3] A. L. Ames, D. R. Nadeau, and J. L. Moreland, *The VRML 2.0 Sourcebook*, 2nd ed. Reading, MA: Addison-Wesley, 1997.
- [4] S. Anahory and D. Murray, *Data Warehousing in the Real World – A Practical Guide for Building Decision Support Systems*. Reading, MA: Addison-Wesley, 1997.
- [5] AuctionBot [Online]. Available: <http://auction.eecs.umich.edu>
- [6] BargainFinder [Online]. Available: <http://bf.cstar.ac.com/bf>
- [7] J. Bettman, *An Information Processing Theory to Consumer Choice*. Reading, MA: Addison-Wesley, 1979.
- [8] A. Blake and M. Isard, *Active Contours*. New York: Springer-Verlag, 1998.
- [9] T. M. Caelli and D. Reye, “On the classification of image regions by color, texture and shape,” *Pattern Recognit.*, vol. 26, no. 4, pp. 461–470, 1993.
- [10] F. S. Cohen, Z. Huang, and Z. Yang, “Invariant matching and identification of curves using B-splines curve representation,” *IEEE Trans. Image Processing*, vol. 4, pp. 1–10, Jan. 1995.
- [11] T. Cootes, C. Taylor, A. Lanitis, D. Cooper, and J. Graham, “Building and using flexible models incorporating grey-level information,” in *Proc. 4th Int. Conf. Computer Vision*, 1993, pp. 242–246.
- [12] P. Dasgupta, N. Narasimhan, L. E. Moser, and P. M. Smith, “MAGNET: Mobile agents for networked electronic trading,” *IEEE Trans. Knowledge Data Eng.*, vol. 11, pp. 509–525, Apr. 1999.
- [13] E. R. Davies, *Machine Vision: Theory, Algorithms, Practicalities*. New York: Academic, 1990.
- [14] D. Dubois and H. Prade, *Fuzzy Sets and Systems: Theory and Applications*. New York: Academic, 1990.
- [15] L. Enbutsu, K. Baba, and H. Hara, “Fuzzy rule extraction from a multilayered neural network,” in *Proc. Int. Joint Conf. Neural Networks*, 1991, pp. 794–799.
- [16] J. Engel and R. Blackwell, *Consumer Behavior*. Hinsdale, IL: CBS College Publishing, 1982.
- [17] O. Faugeras, *3-D Computer Vision*. Cambridge, MA: MIT Press, 1993.
- [18] FTP Software Agents [Online]. Available: <http://www.ftp.com>
- [19] S. Ghosh, Q. Razouqi, H. J. Schumacher, and A. Celmins, “A survey of recent advances in fuzzy logic in telecommunications networks and new challenges,” *IEEE Trans. Fuzzy Syst.*, vol. 6, pp. 443–447, Aug. 1998.
- [20] R. M. Haralick, “Statistical and structural approaches to texture,” *Proc. IEEE*, vol. 67, pp. 786–804, 1979.
- [21] J. Howard and J. Sheth, *The Theory of Buyer Behavior*. New York: Wiley, 1969.
- [22] T. S. Huang, S. Mehrotra, and K. Ramchandran, “Multimedia analysis and retrieval system (MARS) project,” presented at the Proc. 33rd Annu. Clinic Library Application Data Processing—Digital Image Access Retrieval, 1996.
- [23] . [Online]. Available: <http://www.jango.com>
- [24] C. Z. Janikow, “Learning fuzzy controllers by genetic algorithms,” in *Proc. ACM Symp. Applied Computing*, New York, 1994, pp. 1105–1112.
- [25] JATLite [Online]. Available: <http://java.stanford.edu/java-agent/html>
- [26] R. Kalakota and A. B. Whinston, *Electronic Commerce*. Reading, MA: Addison-Wesley, 1997.
- [27] N. N. Karnik, J. M. Mendel, and Q. Liang, “Type-2 fuzzy logic systems,” *IEEE Trans. Fuzzy Syst.*, vol. 7, pp. 643–658, Dec. 1999.
- [28] M. Kass, A. Witkin, and D. Terzopoulos, “Snakes: Active contour models,” in *Proc. 1st Int. Conf. Computer Vision*, 1987, pp. 259–268.
- [29] G. J. Klir, U. H. S. Clair, and B. Yuan, *Fuzzy Set Theory: Foundations and Applications*. Upper Saddle River, NJ: Prentice-Hall, 1997.
- [30] R. S. T. Lee and J. N. K. Liu, “An automatic satellite interpretation of tropical cyclone patterns using elastic graph dynamic link model,” *Int. J. Pattern Recognit. Artificial Intell.*, vol. 13, no. 8, pp. 1251–1270, 1999.
- [31] K. C. Liang and C. C. J. Kuo, “Progressive image indexing and retrieval based on embedded wavelet coding,” *Proc. IEEE Conf. Image Processing*, vol. 1, pp. 572–575, 1997.

- [32] J. N. K. Liu and K. Y. Sin, "Fuzzy neural networks for machine maintenance in mass transit railway system," *IEEE Trans. Neural Networks*, vol. 8, pp. 932–941, Aug. 1997.
- [33] J. N. K. Liu and R. S. T. Lee, "Rainfall forecasting from multiple point sources using neural networks," *Proc. IEEE SMC'1999*, vol. 3, pp. 932–941, 1999.
- [34] D. Lowe, "Fitting parameterized 3-D models to images," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 13, pp. 441–450, May 1991.
- [35] D. Marr and E. Hildreth, "Theory of edge detection," in *Proc. R. Soc. Lond. B*, vol. 207, 1980, pp. 187–217.
- [36] R. Mehrotra and J. E. Gary, "Similar-shape retrieval in shape data management," *IEEE Trans. Comput.*, vol. 28, pp. 57–62, Sept. 1995.
- [37] C. Malsburg and J. Buhmann, "Sensory segmentation with coupled neural oscillators," *Biol. Cybern.*, vol. 67, pp. 233–242, 1992.
- [38] G. Muller, B. Moura, F. Bellard, and C. Consel, "Harissa: A flexible and efficient Java environment mixing bytecode and compiled code," in *Proc. 3rd Conf. Object-Oriented Technologies Systems*, 1997, pp. 1–20.
- [39] Odyssey [Online]. Available: <http://www.genmagic.com>
- [40] S. K. Pal and A. Rosenfeld, "Image enhancement and thresholding by optimization of fuzzy compactness," *Pattern Recognit. Lett.*, vol. 7, pp. 77–86, 1988.
- [41] M. J. Shaw, D. M. Gardner, and H. Thomas, "Research opportunities in electronic commerce," *Dec. Support Syst.*, vol. 21, pp. 149–156, 1997.
- [42] I. Terpsidis, A. Mouskas, B. Pergiodakis, G. Doukidis, and P. Maes, "The potential of E-commerce in re-engineering consumer-retail relationships through intelligent agents," in *Advances in Information Technologies: The Business Challenge*. Amsterdam, The Netherlands: IOS Press, 1997.
- [43] E. Turban, J. Lee, D. King, and H. M. Chung, *Electronic Commerce: A Managerial Perspective*. Upper Saddle River, NJ: Prentice-Hall, 2000.
- [44] Voyager [Online]. Available: <http://www.objectspace.com/voyager>
- [45] L. X. Wang and J. M. Mendel, "Fuzzy adaptive filters, with application to nonlinear channel equalization," *IEEE Trans. Fuzzy Syst.*, vol. 1, pp. 161–170, Apr. 1993.
- [46] F. Wong and P. Z. Wang, "Fuzzy neural systems for decision making," in *Proc. Int. Joint Conf. Neural Networks (IJCNN'91)*, 1991, pp. 497–505.
- [47] J. You and P. Bhattacharya, "A wavelet-based coarse-to-fine image matching scheme in a parallel virtual machine environment," *IEEE Trans. Image Processing*, vol. 9, pp. 1547–1559, Sept. 2000.
- [48] L. A. Zadeh, "Fuzzy sets," *Inform. Control*, vol. 8, no. 3, pp. 338–353, 1965.



James Liu (M'93) received the B.Sc. (Hons) and M.Phil. degrees in mathematics and computational modeling from Murdoch University, Australia, in 1982 and 1987, respectively. He received the Ph.D. degree in artificial intelligence from La Trobe University, Australia, in 1992.

While working on the degree, he worked as a Computer Scientist at Defence Signal Directorate in Australia from 1988 till 1990. He joined the Aeronautical Research Laboratory (ARL) of Defence Science and Technology Organization, Australia, as a Research Scientist in 1990. At ARL, he helped perform AI research in areas of human factors and mission enhancement. He joined the Department of Computing, the Hong Kong Polytechnic University, Hong Kong, in 1994, where he is now an Associate Professor. He has published technical papers on subjects in expert system verification, forecasting systems, pattern recognition, biometrics technology, and e-Commerce applications. His current interests include intelligent business computing, multilingual system development, weather simulation and forecasting, data mining and Web-based information systems, and agent modeling.

Dr. Liu is a Member of AAAI.



Jane You received the B.Eng. degree in electronic engineering from Xi'an Jiaotong University, P.R. China, and the Ph.D. degree in computer science from La Trobe University, Australia, in 1986 and 1992, respectively.

From 1993 to 1995, she was a Lecturer in the School of Computing and Information Science, the University of South Australia, Australia. From 1996 to 2001, she was with School of Computing and Information Technology, Griffith University, Australia, where she was a Lecturer (1996–1998) and later a Senior Lecturer (1999–2001). Currently, she is an Academic Staff Member with the Department of Computing, the Hong Kong Polytechnic University, Hong Kong. Her research interests include visual information retrieval, image processing, pattern recognition, multimedia systems, biometrics computing, and data mining. So far, she has over 120 research papers published as journal articles, book chapters, and conference publications in these areas.

Dr. You was awarded a French Foreign Ministry International Fellowship in 1993.