Parallel Biometrics Computing Using Mobile Agents

J. You, D. Zhang, J. Cao
Department of Computing
The Hong Kong Polytechnic University
Kowloon, Hong Kong
{csyjia, csdzhang, csjcao}@comp.polyu.edu.hk

Minyi Guo
Department of Computer Software
The University of Aizu
Aizu-Wakamatsu City, Japan
minyi@u-aizu.ac.jp

Abstract

This paper presents an efficient and effective approach to personal identification by parallel biometrics computing using mobile agents. To overcome the limitations of the existing password-based authentication services on the Internet, we integrate multiple personal features including fingerprints, palmprints, hand geometry and face into a hierarchical structure for fast and reliable personal identification and verification. To increase the speed and flexibility of the process, we use mobile agents as a navigational tool for parallel implementation in a distributed environment, which includes hierarchical biometric feature extraction, multiple feature integration, dynamic biometric data indexing and guided search. To solve the problems associated with bottlenecks and platform dependence, we apply a four-layered structural model and a three-dimensional operational model to achieve high performance. Instead of applying predefined task scheduling schemes to allocate the computing resources, we introduce a new on-line competitive algorithm to guide the dynamic allocation of mobile agents with greater flexibility. The experimental results demonstrate the feasibility and the potential of the proposed method.

1. Introduction

A wide range of e-commerce applications require high levels of security with reliable identification and verification of the users who access to the on-line services. However, the traditional security measures such as passwords, PIN (Personal Identification Number) and ID cards can barely satisfy the strict security requirements because the use of passwords, PINs and ID cards is very insecure (*i.e.,* they can be lost, stolen, forged or forgotten). More importantly, the password-based methods become ineffective on computer networks, especially the Internet, where attackers can monitor network traffic and intercept passwords or PINs. There is an urgent need to authenticate individuals in the various domains of today's automated, geographically mobile and increasingly electronically wired information society [5]. Biometric technology provides a totally new and yet an effective solution to authentication, which changes the conventional security and access control systems by recognising individuals based on their unique, reliable and stable biological or behavioral characteristics [10].These characteristics include fingerprints, palmprints, iris patterns, facial features, speech patterns and handwriting styles. This new security technique overcomes many of the limitations of the traditional automatic personal identification technologies [11].

The ultimate goal of developing an identity verification system is to achieve the best possible performance in terms of accuracy, efficiency and cost. In general, the design of such an automated biometric system involves biometric data acquisition, data representation, feature extraction, matching, classification and evaluation [1]. Recent studies show that the fusion of multiple sources of evidence can improve performance and increase the robustness of verification [14]. Consequently, it is desirable to utilize and integrate multiple biometric features to improve system accuracy and efficiency. However, a comprehensive, integrated and fully automatic biometric verification and identification system has not yet been fully exploited.

To meet the challenge and immediate need for a high performance Internet authentication service that can secure Internet-based e-commerce applications and overcome the limitations of the existing password-based authentication services on the Internet, we apply biometrics computing technology to achieve fast and reliable personal identification. Considering the reliability and the convenience of biometric data collection from users, four biometric features (*i.e.,* fingerprints, palmprints, hand geometry and face) are used in our proposed system. We adopt a dynamic feature selection scheme for the application-oriented authentication tasks. In other words, a user can determine the level of authentication: either based on a single feature (using an individual's fingerprints, palmprints, hand geometry or face) or multiple features by integrating the individual features into a hierarchical structure for coarse-to-fine matching.

It is noted that it is very important to access and retrieve an individual's biometrics information from large data collections that are distributed over large networks. However, it is difficult to have a uniform search engine that suits various needs.In this paper, we use mobile agents as a navigational tool for a flexible approach to index and search distributed biometrics databases, which can 1) simultaneously extract useful biometrics information from different data collection sources on the network, 2) categorize images by using an index-on-demand scheme that allows users to set up different index structures for fast search, and 3) support a flexible search scheme that allows users to choose effective methods to retrieve image samples.

Although biometrics databases are distributed, most of the current research on biometrics computing has been focused on a single-machine-based system. The methods developed for such a system cannot be simply extended to accommodate a distributed system. In order to effectively index and search for images with specific features among distributed image collections, it is essential to have a sort of "agent" that can be launched to create an index based on specific image feature or to search for specific images with a given content. In this paper, we use mobile agents as a tool to achieve network-transparent biometrics indexing and searching. In addition, we introduce a new system structure for dynamic allocation of mobile agents using online task scheduling to address the limitations of the current approaches and to achieve greater flexibility. The proposed multi-agent system structure is enhanced by push-based technology [2]. The mobile agents are created and cloned dynamically, initialized with service units and pushed from remote sites to local sites that are more convenient for local clients to access, thus speeding up the system's response.

The rest of this paper is organized as follows. Section 2 highlights the hierarchical scheme and dynamic algorithms used in biometrics computing for personal identification and verification. The use of platform-independent mobile agents for parallel biometrics computing is briefly described in Section 3, with the focus on a proposed four-layered structural model and a three-dimensional operational model. An on-line task-scheduling algorithm for dynamically allocationg mobile agents is introduced in Section 4. The experimental results are reported in Section 5. Finally, conclusions are given in Section 6.

2. Fundamentals of Biometrics Computing

In general, a biometric system can operate in two modes: verification and identification. The question of how to represent an image by its biometric features is the first key issue in biometric-based identity authentication. The performance of a biometric system is judged by its accuracy and efficiency. For a verification task, the system deals with a one-to-one comparison. Thus, the focus of multimodal biometrics is to improve the accuracy of the system by either the integration of multiple snapshots of a single biometric or the integration of several different biometrics. For an identification task, the system deals with one-to-many comparisons to find a match. An appropriate integration scheme is required to reduce the computational complexity to achieve the comparison with reliable accuracy. It is essential to index biometrics information by using multiple feature integration to facilitate feature matching for biometrics information retrieval. Feature extraction, the index scheme and the search strategy are three primary issues to be solved. This section describes a hierarchical scheme for biometrics computing that is based on wavelet transforms, which includes multiple feature extraction, dynamic feature indexing and guided search.

2.1. Wavelet Based Multiple Feature Extraction

In contrast to the existing approaches which extract each biometric feature individually, we introduce a hierarchical scheme for multiple biometrics feature representation and integration. Initially, we categorize the biometric features into three classes based on their nature (ı.e., texture feature, shape feature and frequency feature). For example, the texture feature class includes fingerprints, palmprints, iris patterns, *etc.*, while the shape feature class contains features such as facial features, retina, hand geometry, and handwriting. Some features such as speech and texture can also be analyzed in the frequency domain, which belongs to frequency feature class. Then, we use a wavelet-based scheme to combine the different feature classes based on their wavelet coefficients. To capture features at different scales and orientations, we use a wavelet filter bank to decompose the sample data into different decorrelated subbands for feature measurements. The details of the extraction of these features are described in [12].

2.2. Dynamic Biometrics Feature Indexing

Biometrics indexing plays a key role in personal identification. Indexing tabular data for exact matching or range searches in traditional databases is a well-understood problem, and structures like B-trees provide efficient access mechanisms. However, in the context of similarity matching for biometrics images, traditional indexing methods may not be appropriate. Consequently, data structures for fast access to the high-dimensional features of the spatial relationships have to be developed. In this paper, we propose a wavelet-based biometrics image hierarchy and a multiple feature integration scheme to facilitate the dynamic biometrics indexing that is associated with data summarization. Our approach is characterized as follows. 1) To apply wavelet transforms to decompose a given biometric image into three layers of 10 sub-images. 2) To use the mean of the wavelet coefficients in three layers as the global feature measurements with respect to texture and shape, and then index them as tabular data in a global feature summary table. 3) To calculate the mean of the wavelet coefficients of the sub-band images (horizontal, vertical and diagonal) in different layers as local biometrics information, and then index them as tabular data in a local biometrics summary table. 4) To detect the interesting points of the objects in the original image and then store them in a table for fine match. To achieve dynamic indexing and flexible similarity measurement, a statistically-based feature-selection scheme is adopted for multiple feature integration. Such a scheme also coordinates data summarization to search for the best match among biometrics similarities.

2.3. Guided Search

The third key issue in biometrics-based verification and identification is feature matching, which is concerned with verifying and identifying the biometrics features that best match a query sample provided by a user. In contrast to the current approaches, which often use fixed matching criteria to select the candidate images, we propose using selective matching criteria that are associated with a user's query for more flexible search. Our system supports two types of queries: a) to pose a query by using a sample image, and b) to use a simple sketch as a query.

In the case of query by using a sample image, the search follows the process of multiple feature extraction and image similarity measurement that was described in the previous sections. Based on the nature of the query image, the user can add additional component weights during the process of combining image features for image similarity measurement. In the case of query by using a simple sketch provided a user, we apply a B-spline based curve matching scheme to identify the most suitable candidates from the image database. The goal here is to match and recognize the shape curves that were selected in the previous stage. These candidate curves are then modeled as B-splines and the matching is based on comparing their control points (such as the ordered corner points obtained from boundary tracing at the initial stage). Such a process involves the following steps: 1) projective-invariant curve models: uniform cubic B-splines, 2) iterative B-spline parameter estimation, and 3) invariant matching of the curves. In the case of a query by using a sample image, we use an image component code in terms of texture and shape to guide the search for the most appropriate candidates from a database at a coarse level, and then apply image matching at a fine level for the final output.

3. Parallel Biometrics Computing Using Mobile Agents

Parallel computation has been used successfully in many areas of computer science to speed up the computation required to solve a problem. In the field of image processing and computer vision, this is especially appropriate since it appears that the biological model for vision is a parallel model. In contrast to the conventional parallel implementation where either dedicated hardware or software are required, the parallel implementation of our parallel biometrics computing is carried out by using mobile agents in a server to retrieve service units from a back-end database, and a user interface resides at a local server to accept incoming requests, which have been grouped together according to the number of requests that each client makes.

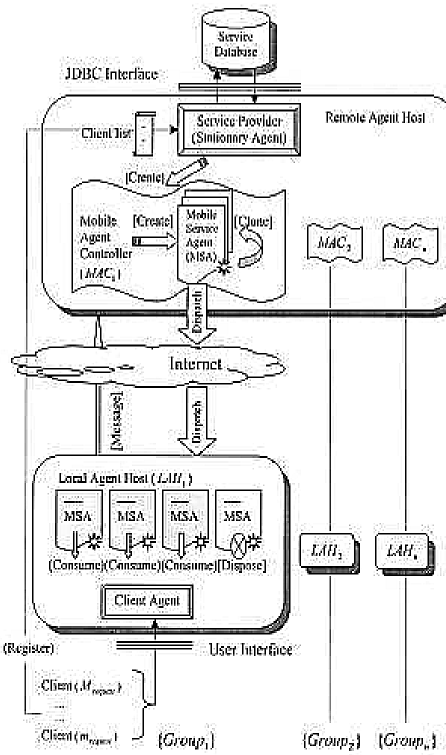


Fig. 1. Multi-agent system structure

The proposed multi-agent system can be viewed in two ways, one way is the system's structural model, and the other is the operational model. The system structural model consists of four layers: 1) A point-to-point end user layer is a P2P (peer-to-peer) communication channel between an end user and an application server. 2) A line-like central controller layer controls all decision-making schemes. 3) A star-like external layer for feature extraction and representation. 4) A oval-like remote application layer is the outer application component for integrating a number of remote legacy systems. From an operational point of view, we propose a three-dimensional model which consists of three blocks: 1) A central-agent host block implementing intelligent detection scheme on a single PC. 2) A neighboring-agent host block implementing a feature extraction scheme in a parallel computing environment. 3) A remote-agent host block implementing a remote decision-making scheme in a distributed environment.

The design of the proposed system is straightforward, but a problem arises when a remote-agent controller prepares for service units: it needs to decide how many mobile service units (agents) to create, clone and dispatch. If insufficient service units are prepared, the agent controller has to clone more and supply them later, which causes a delay in the system's response. If too many service units are prepared, then the unclaimed service agents can be disposed of explicitly, but this increases system overheads. In the proposed system, both lower overheads and faster service performance need to be taken into account. Section 4 presents an on-line algorithm to address the problems described above, which we call the on-line task-scheduling problem (OTSP).

3.2. Implementation Strategy

To facilitate the implementation of each component of the system, we adopt a hybrid agent computing paradigm. There are two classes of agents: global agents and local agents. The global agents handle inter-image coordination, query processing and reasoning. Each global agent may consist of a few sub-agents. The following list describes the five global agents and their associated sub-agents that are proposed in our system:

- *Coordinator Agent:* coordinating other global agents and image agents

- *Query Agent:* processing users' complex queries using three sub-agents, namely:

  – query understanding: categorizing a query (texturet-based, shape-based, or combination)

  – query reasoning: extracting text-based keywords for image grouping

  – query feature formation: selecting appropriate features for object generation and manipulation

- *Wavelet Agent:* generating wavelet coefficients for multiple feature representation and integration using three sub-agents:

  – wavelet transform : decomposition of an image into a series of sub-band images in an image hierarchy

  – feature representation: of an individual feature vector in terms of the wavelet coefficients

  – feature integration: combining multiple feature vectors with adjustment weights

- *Verification/Identification Agent:* performing hierarchical feature matching for identity verification and identification using two sub-agents:

  – matching criterion selection: selection of similarity measures

  – feature matching: hierarchical image matching

- *User Interface Agent*: managing all user interactions

The local agents are referred to as *Image Agents*, which are responsible for performing relevant biometrics computing tasks on each individual image.

4. On-line Task Scheduling Using A Competitive Algorithm

4.1. Background

On-line problems can be found in many research areas such as data structuring, task scheduling or resource allocation [4]. In general, on-line problems are characterized by the need to satisfy requests or make decisions without foreknowledge of future requests [7]. This is different from traditional system analysis approaches where algorithms are designed with the assumption that the complete sequence of requests is known.

In this paper, we develop a competitive task-scheduling algorithm to allocate mobile agents to client service requests. Our aim is to minimize the total cost of service in an on-line environment. We measure each service in terms of its computing cost. For example, we use *cost(creation)* to denote the average cost of the creation service. Five types of costs are involved in our proposed system: cost(creation), cost(cloning), cost(dispatching), cost(disposal) and cost(messaging).

We examine three scenarios when a client with x requests consumes d ready-made service units, we have the following cases:

1) $x = d$, the client happens to consume all of the ready-made service units. In this scenario, each service agent costs $c = Cost(creation) + Cost(cloning) + Cost(dispatching)$, and the total cost is cd or cx. 2) $x < d$, the number of ready-made service units exceeds the client's requests, so $(d-x)$ units are eliminated. In this scenario, we have $c_1 = Cost(creation) + Cost(cloning) + Cost(dispatching) + Cost(disposal)$, and the total cost is $cx + c_1(d-x)$. 3) $x > d$, the number of ready-made service units are insufficient to meet the client's requests, so another $(x-d)$ units are cloned and dispatched. In this scenario, we have $c_2 = Cost(messaging) + Cost(cloning) + Cost(dispatching)$, and the total cost is $cd + c_2(x-d)$ or $cx + (c_2 - c)(x-d)$.

If the actual request sequence is denoted by $\sigma = (x_1, x_2, ..., x_n)$, where x_i means the actual number of requests in the *i*th period, we can obtain the optimal off-line cost of the problem (for a single client):

$$C_{OPT}(\sigma) = c \cdot \sum_{i=1}^{n} x_i$$

For the same request sequence, if d_i denotes the service units that should be prepared by the on-line decision-maker for the *i*th period, we can obtain the on-line cost of a competitive algorithm A as follows:

$$C_A(\sigma) = c \cdot \sum_{i=1}^{n} x_i + (c_2 - c) \cdot \sum_{i=1, x_i > d_i}^{n} (x_i - d_i) + c_1 \cdot \sum_{i=1, x_i < d_i}^{n} (d_i - x_i)$$

For any on-line algorithm A, the competitive ratio is defined as:

$$\alpha = \inf_{\sigma} \frac{C_A(\sigma)}{C_{OPT}(\sigma)}$$

A small competitive ratio implies that A can do well in comparison with the optimal (OPT) solution. In designing a competitive algorithm for the on-line task scheduling problem, the agent controller (*i.e.*, the on-line decision-maker) does not know beforehand the actual number of requests in the *i*th period. Instead, the controller knows the possible range in the number of requests denoted by $[m, M]$. The on-line competitive algorithm, A, needs to give the best possible choice for the number of service units (d) to prepare for the *i*th period, which would result in the smallest competitive ratio α.

4.2. General Harmonic Algorithm (GHA)

We propose applying a general harmonic algorithm (GHA) to achieve an optimal competitive ratio for the dynamic allocation of mobile agents in an on-line environment. The following description summarizes the major features of this algorithm. The details of the proof and explanations are presented in [9].

- *Algorithm Description*

  Theorem 1 For the on-line OTSP problem, the best choice that an on-line decision-maker can make is: $d = \frac{Mm \cdot (p+q-1)}{Mp + m \cdot (q-1)}$, where $p = c_1/c, q = c_2/c$ and $[m, M]$ is the possible range in the number of requests for a client from a given group.

- *Competitive Ratio*

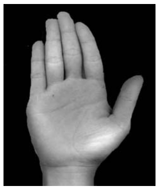
  Theorem 2 For the competitive GHA algorithm given in Theorem 1, the competitive ratio is: $\alpha = [1 + p(q-1)\frac{M-m}{Mp+m(q-1)}]$

- *Lower Bound for the Competitive Ratio*

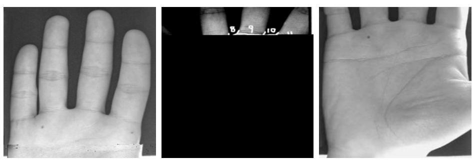
  Theorem 3 For the competitive ratio of on-line GHA algorithm, the lower bound is: $1 + p(q-1)\frac{M-m}{Mp+m(q-1)}$.

5. Experimental Results

The biometrics image samples that were used for testing are of size 232 × 232 with a resolution of 125 dpi and 256 grayscales. Four types of biometrics features, namely hand geometries, fingers, palmprints and faces are considered. A total of 2,500 images from 500 individuals are stored in our database. These biometrics samples were collected from both female and male adults within the age range from 18 to 50. A special electronic sensor was used to obtain digitized samples on-line. Fig. 2 illustrates the samples of digitized hand boundary, fingers and palmprints of one hand. A series of experiments were carried out to verify the high performance of the proposed algorithms.



hand



fingers   palm boundary   palmprint

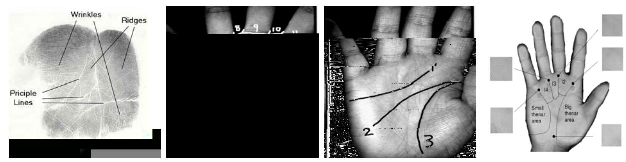
Fig. 2. An example of multiple feature integration for hand representation

5.1. Multiple Biometrics Feature Extraction Test

The shape of a hand boundary can be used as a global biometrics feature for coarse matching. This data can be presented by a data cube, where each dimension corresponds to a particular feature entity. The following list describes the relevant features as dimensions for hand shape classification. (1) A list of boundary feature points, sorted in an anti-clockwise order along the hand boundary. (2) Parameters that control the active contour along the hand boundary. (3) Measures of invariant moments, sorted in descending order. (4) A list of the coefficients of a B-Spline curve. (5) The time when an image was taken. (6) A list of individual's identity details.

The above multi-dimensional structure for the data cube offers flexibility to manipulate the data and view it from different perspectives. Such a structure also allows quick data summarization at different levels. For a collection of 200 hand image samples, 80% of the candidates are excluded after the coarse level selection.

The dynamic selection of image features is further demonstrated by multi-level palmprint feature extraction for personal identification and verification (refer to [5] for a survey, also see our preliminary work on palmprint verification [13]). Our experiment is carried out in two stages. In stage one, the global palmprint features are extracted at a coarse level and candidate samples are selected for further processing. In stage 2, the regional palmprint features are detected and a hierarchical image matching is performed for the final retrieval output. Fig. 3 illustrates the multi-level extraction of the palmprint features. Fig. 3(a) shows a sample of palmprint, Fig. 3(b) shows delineation of the boundary of a palm as a global boundary feature. Fig. 3(c) shows the detection of global principal lines and Fig. 3(d) shows the details of the regional palmprint texture features for local feature representation at a fine level. The average accuracy rate for classification is 97%.



(a) palm   (b) palm-boundary   (c) palm-line   (d) palm-region

Fig. 3. Hierarchical feature extraction

5.2. Hierarchical Feature Matching Test

The performance of the proposed coarse-to-fine curve-matching approach is further demonstrated in the second test, which is face recognition for personal identification. At a coarse level, a fractional discrimination function is used to identify the region of interest in an individual's face. At a fine curve-matching level, the active contour tracing algorithm is applied to detect the boundaries of interest in the facial regions for the final matching. Fig. 4 illustrates the tracing of facial curves for face recognition. Fig. 4(a) is an original image, Fig. 4(b) shows the boundaries of a region of on the face and Fig. 4(c) presents the curve segments for hierarchical face recognition by curve matching.

original image   boundary detection   face curves

Fig. 4. Face curve extraction

To verify the effectiveness of our approach, a series of tests were carried out using a database of 200 facial images collected from different individuals under various conditions, such as uneven lighting, moderate facial tilting and partial occlusion. Table 1 lists the correct recognition rate of the coarse-level detection.

Proceedings of the 2003 International Conference on Parallel Processing (ICPP'03)
0190-3918/03 $ 17.00 © 2003 IEEE

Table 1: Performance of face detection at coarse-level

Face Condition	Correct Detection Rate
unevenness of lighting	98%
multiple faces	95%
moderate tilt of faces	97%
partial sheltering	85%

To show the robustness of the proposed algorithm for face detection that is invariant to the perspective view, partial distortion and occlusion, the fine-level curve matching is applied to facial images with different orientations and expressions. Fig. 5(a) illustrates sample images of the same person using person under different conditions such as facial expression, partial occlussion and distortion. Table 2 and Table 3 summarizes the test results for 100 cases.

The face samples at different orientations

The face samples of different conditions

Fig. 5. The face samples

Table 2: Performance of face recognition at different orientions

Viewing Perspective	Correct Classification Rate
-20^0 (vertical)	84%
-10^0 (vertical)	86%
$+10^0$ (vertical)	86%
$+20^0$ (vertical)	83%
-20^0 (horizontal)	85%
-10^0 (horizontal)	87%
$+10^0$ (horizontal)	87%
$+20^0$ (horizontal)	84%

5.3. Evaluation of System Efficiency

To test the increased efficiency of the proposed agent-based approach, a group of external assistant agents were employed. The central agent controller is responsible for dissecting the task and assembling the final result. According to the number of available worker agents, different strategies will be adopted to partition and distribute the subtasks.

Table 3: Performance of face classification with different conditions

Face Condition	Correct Classification Rate
partial occlusion	77%
various expressions	81%
wearing glasses	82%

- Five-worker pattern

  For facial feature extraction, if all facial features (such as the chin, left eye, right eye, mouth and nose) form a set $X = \{c, l, r, m, n\}$, this strategy simply dissects the whole task into five parts, and then distributes them to five worker agents. The total processing time is $\max\{t_1(c), t_2(l), t_3(r), t_4(m), t_5(n)\} + latencies$, where *latencies* include network latency and data preparation latency for packaging and unpacking.

- four-worker pattern

  From experiments, it could be noted that with the same CPU speed: $t(l+r) < t(m)$. So it will not take longer if $\{l, r\}$ could be processed on one host. The total processing time is $\max\{t_1(c), t_2(l+r), t_3(m), t_4(n)\} + latencies$.

- Three-worker pattern

  The three worker-pattern is proposed due to the fact that $t(l+r+n) \approx t(m)$. So the total execution time is $\max\{t_1(c), t_2(l+r+n), t_3(m)\} + latencies$ if $\{l, r.n\}$ would be processed on one host.

- Two-worker pattern

  The two-worker pattern is the minimum requirement for the proposed system structure. A simple algorithm is introduced to decide how to schedule tasks between two hosts. The task subset, $A(A \subset X, A \neq \emptyset)$, that needs to be handled by Host 1 must satisfy the following formula:

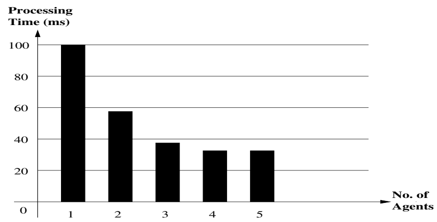
  $$\min_{A\subset X, A\neq\emptyset}\{|\sum_{x\in A} t_{Host1}(x) - \sum_{x\in A} t_{Host2}(x)|\}$$

  and the total processing time is

  $$\max_{A\subset X, A\neq\emptyset}\{\sum_{x\in A} t_{Host1}(x), \sum_{x\in A} t_{Host2}(x)\} + latencies$$

The increased speed ratio of different patterns is illustrated in Fig. 6.

The system efficiency is further judged using the round trip time (RTT) test. Instead of calculating the difference between the arrival and departure time at 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 that are spent on each of the various operations, starting with the collection of user requirements, continuing with biometrics feature extraction, similarity measurement, and searching for the best matching. Two servers are used in this test. Server 1 is located on the same local area network (LAN) as



2

Fig. 6. The increased efficiency ratio of different patterns

the client machine, whereas Server 2 is situated at a remote site within the campus. Table 4 shows the average execution time at different stage for 100 trials. It is noted that most of the execution time is spent on the fine-level matching. In practice, the network traffic should be considered for real applications.

Table 4: The evaluation of system efficiency evaluation – average RTT for 100 trials

Server Location	Execution Time (ms) Server 1	Execution Time (ms) Server 2
A. Client Machine		
user requirement		
query processing	210	285
B. Server Machine		
feature extraction	1020	1315
feature integration	360	430
C. Client Machine		
coarse-level matching	1320	1830
fine-level matching	2510	3280
TOTAL RTT	5420	7140

6. Conclusion

This paper explores the integration of distributed computing methodology, agent technology, pattern recognition techniques and on-line competitive algorithms to provide an effective and efficient approach to identity authentication using personal features (biometrics). To overcome the limitations of the current security systems, which use fixed preselected features and have bottlenecks of slow performance and platform dependence, we develop a parallel biometrics based personal identification and verification system using mobile agents. To tackle the key issues such as biometrics feature extraction, indexing and search, we propose a hierarchical approach to fast content-based biometric image retrieval by dynamic indexing and guided search. In addition, we introduce an innovative four-layer structural model and a three-dimension operational model to achieve high performance. Furthermore, the proposed competitive algorithm has the optimal competitive ratio to guide dynamic task scheduling.The experimental results confirm that our approach is feasible for on-line identity authentication and verification and will be useful for many other security applications.

7. Acknowledgement

The authors would like to thank for the partial support of a Hong Kong Government CERG Research Grant (B-Q518: PolyU5076/01E) and a Hong Kong Polytechnic University Research Grant (G-T613).

References

[1] S. Antani, R. Kasturi and R. Jain, “A survey on the use of pattern recognition methods for abstraction, indexing and retrieval of images and video,” *Pattern Recognition*, vol. 35, pp. 945-965, 2002.

[2] G. Cybenko, “The foundations of information push and pull,” *Mathematics of Information*, D. O’Leary ed, 1999.

[3] P. Dasgupta, N. Narasimhan, L.E. Moser and P.M. Smith, “MAGNET: Mobile agents for networked electronic trading,” *IEEE Trans. on Knowledge and Data Engineering*, vol. 11, no. 4, pp. 509-525, 1999.

[4] A. Fiat and G.J. Woeginger, “Online algorithms – the state of the art,” *Lecture Notes in Computer Science*, Springer, 1998.

[5] A. Jain, R. Bolle and S. Pankanti, *Biometrics: Personal Identification in Networked Society*, Kluwer Academic Publishers, 1999.

[6] J. Kiniry, D. Zimmerman, “Special feature: A hands-on look at Java mobile agents,” *IEEE Internet Computing*, vol. 1, no, 4, 1997.

[7] E. Koutsoupias and C. papadimitriou, “Beyond competitive analysis,” *SIAM Journal on Computing*, vol. 30, no. 1, pp. 300-317, 2000.

[8] D.B. Lange and M. Oshima, *Programming and Deploying Java Mobile Agents with Aglets*, Addison-Wesley, 1998.

[9] Weimin Ma, *Competitive Algorithms for On-line Problems and Their Applications to Mobile Agents*, M.Phil. Thesis, The Hong Kong Polytechnic University, 2002.

[10] B. Miller, “Vital signs of identity,” *IEEE Spectrum*, vol. 32, pp. 22-30, 1994.

[11] W. Shen, M. Surette and R. Khanna, “Evaluation of automated biometrics-based identification and verification systems,” *Proceedings of the IEEE*, vol. 85, no. 9, pp. 1464-1478, 1997.

[12] 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, no. 9, pp. 1547-1559, 2000.

[13] J. You, W.X. Li and D. Zhang, “Hierarchical palmprint identification via multiple feature extraction,” *Pattern Recognition*, vol. 35, pp. 847-859, 2002.

[14] D. Zhang (edited), *Biometrics Resolutions for Authentication in An e-World*, Kluwer Academic Publishers, 2002.