ISSN : 2165-4069(Online)
ISSN : 2165-4050(Print)

IJARAI

International Journal of Advanced Research in Artificial Intelligence

Volume 2 Issue 2

www.ijarai.thesai.org

A Publication of
The Science and Information Organization

IJARAI

INTERNATIONAL JOURNAL OF
ADVANCED RESEARCH IN ARTIFICIAL INTELLIGENCE

SAI

THE SCIENCE AND INFORMATION ORGANIZATION

www.thesai.org | info@thesai.org

Editorial Preface

From the Desk of Managing Editor…

"The question of whether computers can think is like the question of whether submarines can swim." ― Edsger W. Dijkstra, the quote explains the power of Artificial Intelligence in computers with the changing landscape. The renaissance stimulated by the field of Artificial Intelligence is generating multiple formats and channels of creativity and innovation.

This journal is a special track on Artificial Intelligence by The Science and Information Organization and aims to be a leading forum for engineers, researchers and practitioners throughout the world.

The journal reports results achieved; proposals for new ways of looking at AI problems and include demonstrations of effectiveness. Papers describing existing technologies or algorithms integrating multiple systems are welcomed. IJARAI also invites papers on real life applications, which should describe the current scenarios, proposed solution, emphasize its novelty, and present an in-depth evaluation of the AI techniques being exploited. IJARAI focusses on quality and relevance in its publications.

In addition, IJARAI recognizes the importance of international influences on Artificial Intelligence and seeks international input in all aspects of the journal, including content, authorship of papers, readership, paper reviewers, and Editorial Board membership.

The success of authors and the journal is interdependent. While the Journal is in its initial phase, it is not only the Editor whose work is crucial to producing the journal. The editorial board members , the peer reviewers, scholars around the world who assess submissions, students, and institutions who generously give their expertise in factors small and large— their constant encouragement has helped a lot in the progress of the journal and shall help in future to earn credibility amongst all the reader members.

I add a personal thanks to the whole team that has catalysed so much, and I wish everyone who has been connected with the Journal the very best for the future.

Thank you for Sharing Wisdom!

Managing Editor
IJARAI
Volume 2 Issue 2 February 2013
ISSN: 2165-4069(Online)
ISSN: 2165-4050(Print)
©2013 The Science and Information (SAI) Organization

Editorial Board

Peter Sapaty - Editor-in-Chief
National Academy of Sciences of Ukraine
Domains of Research: Artificial Intelligence

Alaa F. Sheta
Electronics Research Institute (ERI)
Domain of Research: Evolutionary Computation, System Identification, Automation and Control, Artificial Neural Networks, Fuzzy Logic, Image Processing, Software Relaibility, Software Cost Estimation, Swarm Intelligence, Robotics

Antonio Dourado
University of Coimbra
Domain of Research: Computational Intelligence, Signal Processing, data mining for medical and industrial applications, and intelligent control.

David M W Powers
Flinders University
Domain of Research: Language Learning, Cognitive Science and Evolutionary Robotics, Unsupervised Learning, Evaluation, Human Factors, Natural Language Learning, Computational Psycholinguistics, Cognitive Neuroscience, Brain Computer Interface, Sensor Fusion, Model Fusion, Ensembles and Stacking, Self-organization of Ontologies, Sensory-Motor Perception and Reactivity, Feature Selection, Dimension Reduction, Information Retrieval, Information Visualization, Embodied Conversational Agents

Liming Luke Chen
University of Ulster
Domain of Research: Semantic and knowledge technologies, Artificial Intelligence

T. V. Prasad
Lingaya's University
Domain of Research: Bioinformatics, Natural Language Processing, Image Processing, Robotics, Knowledge Representation

Wichian Sittiprapaporn
Mahasarakham University
Domain of Research: Cognitive Neuroscience; Cognitive Science

Yaxin Bi
University of Ulster
Domains of Research: Ensemble Learing/Machine Learning, Multiple Classification Systesm, Evidence Theory, Text Analytics and Sentiment Analysis

Reviewer Board Members

- **Alaa Sheta**
  Electronics Research Institute (ERI)
- **Albert Alexander**
  Kongu Engineering College
- **Amir HAJJAM EL HASSANI**
  Université de Technologie de Belfort-Monbéliard
- **Amit Verma**
  Department in Rayat & Bahra Engineering College,Mo
- **Antonio Dourado**
  University of Coimbra
- **B R SARATH KUMAR**
  Lenora College of Engineering
- **Babatunde Opeoluwa Akinkunmi**
  University of Ibadan
- **Bestoun S.Ahmed**
  Universiti Sains Malaysia
- **Chien-Peng Ho**
  Information and Communications Research Laboratories, Industrial Technology Research Institute of Taiwan
- **David M W Powers**
  Flinders University
- **Dimitris_Chrysostomou**
  Democritus University
- **Dr.Dhananjay R Kalbande**
  Mumbai University
- **Francesco Perrotta**
  University of Macerata
- **Frank_Ibikunle**
  Covenant University
- **Grigoras Gheorghe**
  "Gheorghe Asachi" Technical University of Iasi, Romania
- **Guandong Xu**
  Victoria University
- **Haibo Yu**
  Shanghai Jiao Tong University
- **Jatinderkumar R. Saini**
  S.P.College of Engineering, Gujarat
- **Krishna Prasad Miyapuram**
  University of Trento
- **Luke_Liming_Chen**
  University of Ulster
- **Marek Reformat**
  University of Alberta
- **Md. Zia Ur Rahman**
  Narasaraopeta Engg. College, Narasaraopeta
- **Mokhtar_Beldjehem**
  University of Ottawa
- **Monji Kherallah**
  University of Sfax
- **Nitin S. Choubey**
  Mukesh Patel School of Technology Management & Eng
- **Rajesh Kumar**
  National University of Singapore
- **Rajesh K Shukla**
  Sagar Institute of Research & Technology-Excellence, Bhopal MP
- **Rongrong Ji**
  Columbia University
- **Said_Ghoniemy**
  Taif University
- **Samarjeet_Borah**
  Dept. of CSE, Sikkim Manipal University
- **Sana'a Wafa Tawfeek Al-Sayegh**
  University College of Applied Sciences
- **Saurabh Pal**
  VBS Purvanchal University, Jaunpur
- **Shahaboddin_Shamshirband**
  University of Malaya
- **Shaidah Jusoh**
  Zarqa University
- **Shrinivas_Deshpande**
- **SUKUMAR SENTHILKUMAR**
  Universiti Sains Malaysia
- **T C.Manjunath**
  HKBK College of Engg
- **T V Narayana Rao**
  Hyderabad Institute of Technology and Management
- **T. V. Prasad**

Lingaya's University

- **Vitus S.W Lam**
  Domains of Research
- **VUDA Sreenivasarao**
  St. Mary’s College of Engineering & Technology
- **Vishal Goyal**
- **Wei Zhong**
  University of south Carolina Upstate
- **Wichian Sittiprapaporn**
  Mahasarakham University
- **Yaxin Bi**
  University of Ulster
- **Yuval Cohen**
  The Open University of Israel
- **Zhao Zhang**
  Deptment of EE, City University of Hong Kong
- **Zne-Jung Lee**
  Dept. of Information management, Huafan University
- **Zhigang Yin**
  Institute of Linguistics, Chinese Academy of Social Sciences

CONTENTS

Emotional Belief-Desire-Intention Agent Model: Previous Work and Proposed Architecture

Mihaela-Alexandra Puică
Computer Science and Engineering Department
Faculty of Automatic Control and Computers
University Politehnica of Bucharest, Romania

Adina-Magda Florea
Computer Science and Engineering Department
Faculty of Automatic Control and Computers
University Politehnica of Bucharest, Romania

***Abstract*— Research in affective computing shows that agents cannot be truly intelligent, nor believable or realistic without emotions. In this paper, we present a model of emotional agents that is based on a BDI architecture. We show how we can integrate emotions, resources and personality features into an artificial intelligent agent so as to obtain a human-like behavior of this agent. We place our work in the general context of existing research in emotional agents, with emphasis on BDI emotional models.**

Keywords— Affective Computing; Agent architecture; Belief-Desire-Intention model

I. INTRODUCTION

When building rational agents, the Belief-Desire-Intention (BDI) model [1] has proven to be one of the best options one can select. It is based on the human reasoning pattern, known as practical reasoning [2]: first decide what one wants to achieve (deliberation), and then decide on how to do it (means-end reasoning). Thus, the agent that follows this model aims at displaying rational behavior by pursuing its *goals*, achieving its *intentions* and using the *beliefs* it has about itself and about the environment.

But research in psychology, neurology and cognitive science shows that people not only use their cognitive functions, but also account for their emotions (even unconsciously) when taking decisions. If these two parts don't interconnect in a proper manner, multiple options are harder to be filtered and bad decisions are easier to take, as stated in work done by Damasio [3]. These results lead to the concept of emotional agents, which aim at being more realistic and providing a more engaging experience in human-computer interaction (HCI), but also at improving the performance of rational agents.

Combining the two concepts, we obtain an agent that reasons based on its beliefs, desires, intentions and emotions. Our aim is to build such an agent, based on previous research done in the field. Consequently, the paper is organized as follows: Section II presents the emotional agent architecture that we developed, and then Section III describes related work concerning existing emotional BDI agents. In Section IV an example of a scenario is proposed and, finally, Section V draws conclusions and outlines future work.

*The work has been funded by the Sectorial Operational Programme Human Resources Development 2007-2013 of the Romanian Ministry of Labor, Family and Social Protection through the Financial Agreement POSDRU/107/1.5/S/76813.

II. EMOTIONAL AGENT ARCHITECTURE

We propose an emotional agent architecture that focuses on the influence of emotions on the behavior and the way resources are used depending on the emotional state. Therefore, we aim to integrate the following *concepts*:

PERCEPTS

- anything that comes from the environment: stimuli or messages from other agents
- influenced by emotions

BELIEFS

- acquired from percepts
- revised to account for current beliefs and new percepts
- influenced by emotions

DESIRES

- goals received by the agent at design time
- constant over time

OPTIONS

- alternatives to accomplish the desires
- generated based on current beliefs and intentions
- influenced by emotions

INTENTIONS

- options that the agent has committed to
- revised based on current intentions, beliefs, options, emotions and available resources
- open-minded commitment - agent is committed to the intention as long as it is not achieved yet, it is not believed impossible to achieve and it is still a goal for the agent

EMOTIONS

- primary emotions may determine instinctual behavior
- secondary emotions influence cognitive processes and available resources
- fixed set of emotions for each scenario

PERSONALITY

- two axes: extrovert-introvert, psychologically stable-unstable
- four types: sanguine, choleric, melancholic, phlegmatic

RESOURCES

- maintained in a structure that gives access to them selectively, based on emotions
- fixed set of resources for each scenario

The *functions* that define the processes that take place within the agent internal mechanisms are the following:

- $perceive: Env\ x\ E \rightarrow P$ – perceives the stimuli in the environment (Env), influenced by emotions (E); returns the *percepts P*
- $peu: P\ x\ E\ x\ I \rightarrow E$ – primary emotions update; the new percepts, influenced by intentions and current emotions, generate *primary emotions*
- $react: E\ x\ P\ x\ AR \rightarrow \pi -$ defines reactive behavior; reacts to the percepts, influenced by emotions and guided by the *available resources* (AR), returning a *plan* to be executed (generally consisting in only one simple action)
- $brf: P\ x\ B\ x\ E \rightarrow B$ – *belief* revision function; revises current beliefs based on percepts and influenced by emotions
- $seu: B\ x\ E\ x\ I\ x\ AR \rightarrow E$ – secondary emotions update; beliefs, intentions, available resources and current emotions generate *secondary emotions*
- $ru: R\ x\ E \rightarrow AR$ – resources update; emotions influence the *available resources* the agent can use from the set of all *resources* it has (which are not always accessible)
- $analyzer: B\ x\ D\ x\ I\ x\ E \rightarrow O$ – appraises the current situation in the context of its beliefs, desires and intentions, influenced by its emotions, to generate the current *options* that it has to cope with the situation
- $filter: O\ x\ B\ x\ E \rightarrow I$ – filters the available options to find the *intentions* that it will be committed to
- $plan: I\ x\ AR \rightarrow \pi$ – structures intentions into plans, according to the available resources
- $execute: \pi \rightarrow Env$ – executes the plan
- $reconsider: I\ x\ B \rightarrow \{true, false\}$ – decides whether it is necessary to reconsider or not the intentions
- $sound: \pi\ x\ I\ x\ B \rightarrow \{true, false\}$ – decides whether executing the plan will lead to intentions achievement or not

Fig. 1 illustrates the proposed architecture. The processes that take place within the agent control loop are shown in Fig. 2. In what follows, we will explain in detail this process, walking step by step through the internal mechanism of the agent.

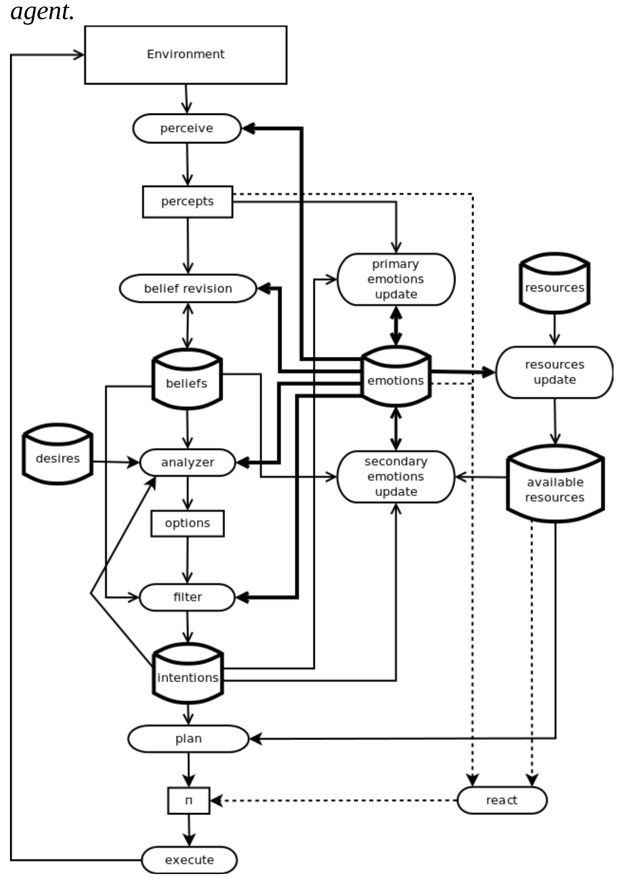


Fig. 1. Proposed Architecture

First, the agent perceives the stimuli in the environment ($perceive$), which can be the occurrence of an action, a change in the state of the world or a message sent by another agent. Percepts are then quickly appraised at an affective level and may determine the experience of primary emotions (peu).

When the intensity of the new emotions passes a certain threshold ε, primary emotions may determine the agent to execute some predefined reactive behavior. For example, strong fear can determine a sudden fall back or a sudden stroke on what caused the fear (depending on the personality). Percepts give the context and available resources determine the action that is to be taken. The $react$ function returns a plan π which generally contains only one action.

Primary emotions generate a reactive behavior which could be unnecessary, but it is very useful in survival situations. Fig. 4 shows a dotted input for the react function to emphasize that the process is not guaranteed to happen at every step, but it is a matter of strong, sudden emotions.

After the initial instinctual behavior, the percepts influence belief revision (br). This process must account for both current beliefs and new percepts, acting like a truth maintenance system which outputs consistent beliefs:

After belief revision, secondary emotions are being computed. These emotions are the result of a cognitive appraisal of the situation which takes into consideration not the raw percepts, but the newly revised beliefs.

```
D = D0;
while true do
    P ← perceive(Env, E);
    E ← peu(P, E, I);
    if intensity(E) > ε then
        π ← react(E, P, AR);
        P ← perceive(Env, E);
    end if
    B ← brf(P, B, E);
    E ← seu(B, E, I, AR);
    AR ← ru(R, E);
    O ← analyzer(B, D, I, E);
    I ← filter(O, B, E);
    π ← plan(I, AR);
    while not (empty(π)
            || succeeded(I, B)
            || impossible(I, B)) do
        α := head(π);
        execute(α);
        π := tail(π);
        P ← perceive(Env, E);
        E ← peu(P, E, I);
        if intensity(E) > ε then
            π ← react(E, P, AR);
            P ← perceive(Env, E);
        end if
        B ← brf(P, B, E);
        E ← seu(B, E, I, AR);
        AR ← ru(R, E);
        if reconsider(I, B) then
            O ← analyzer(B, D, I, E);
            I ← filter(O, B, E);
        end if
        if not sound(π, I, B) then
            π ← plan(I, AR);
        end if
    end while
end while
```

Fig. 2. Agent Control Loop

But emotions in their turn influence the available resources that an agent perceives (ru), a process that models human resource usage: when experiencing fear, one can run faster or be stronger. Thus, at a given moment, depending on the affective state, an agent can have access to different resources.

Next follows deliberation (analyzer and filter) and means-end reasoning (plan). The $analyzer$ function determines the available options that the agent has to accomplish its desires. The $filter$ function chooses from the available options those that the agent will be committed to. The selected options are the agent intentions.

After deciding on the intentions, the agent must $plan$ on how to achieve them, taking into consideration available resources. In the end, the plan is ready to be executed ($execute$).

But during this process, the agent still has to pay attention to the stimuli in the environment. If a plan contains several actions, the environment can change its state during execution, so the agent should not omit to reconsider its plan or its intentions. In the light of new events, the intentions could no longer be necessary, so the agent should consider dropping them. This is done in the $reconsider$ function.

If this functions returns true, then the agent restarts the deliberation process, returning renewed intentions. This approach is known as open-minded commitment strategy relative to intentions. In addition, the agent must test the soundness of the plan in relation to the intentions, and replan if necessary ($sound$).

III. Modeling Emotions

A. Emotions

We define emotions using the circumplex model of affect developed by James Russell in 1980 [13]. This is a dimensional model that represents affect in a 2D emotional space determined by the valence and arousal axes. Fig. 3 shows eight emotions placed in this space. The valence axis is defined by pleasure-misery, while the arousal axis by arousal-sleepiness. The other four emotions simply define the contents of each quadrant.

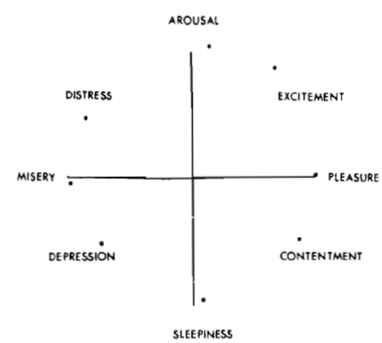


Fig. 3. James Russell's Circumplex Model Of Affect

Consequently, we thus can assume that an emotion is a point in this 2D space given by the pair (v, a), where both v and a are rational numbers within given intervals: v is the value of the valence and a is the value of the arousal. Following the OCC model of emotions [12], an event is appraised in terms of beliefs, desires and intentions, returning a certain score regarding its valence (positive or negative) and arousal (the intensity of emotion felt).

We believe that the circumplex model of affect permits to easily define different sets of emotions for different scenarios. The six basic emotions are not suitable in every context; for example, in a fire scenario it is less likely that one will feel joy. Most probably, a drama scenario may activate emotions in the upper left quadrant, while a romance scenario may activate emotions in the lower right quadrant.

B. Resources

Resources have a central role in decision making because decisions are based on the available resources that the agent knows it has. In a critical situation, people may suddenly appear to have more energy than they usually do. The arousal opens new paths in the person's capabilities, but working at full capacity exhausts the energy, so it is not a desired feature unless situation requires it.

In our architecture, we define three types of resources, each with different access conditions. Type A of resources can always be used by the agent. Type B defines resources that the agent can access only when its emotional intensity goes over a given threshold. Type C of resources is used only in survival situations. Fig. 4 shows the relation between arousal and resource accessibility, using the Yerkes-Dodson law [14] to emphasize the fact that using more resources may improve performance.

The picture depicts the emotion intensity thresholds which determine the type of resources the agent has access to. Thus, if we define by e_i the emotion intensity, and by p the performance, we have the following cases:

$$0 < e_i < e_{i1} \rightarrow \begin{cases} access\ to\ type\ A \\ p \leq p_1 \end{cases}$$

$$e_{i1} < e_i < e_{i2} \rightarrow \begin{cases} access\ to\ type\ A\ \&\ B \\ p_1 < p \leq p_2 \end{cases}$$

$$e_{i2} < e_i < e_{i3} \rightarrow \begin{cases} access\ to\ type\ A\ \&\ B\ \&\ C \\ p_2 < p \leq p_{max} \end{cases}$$

$$e_{i3} < e_i < e_{i4} \rightarrow \begin{cases} access\ to\ type\ A\ \&\ B \\ p_1 < p \leq p_2 \end{cases}$$

$$e_i < e_{i4} \rightarrow \begin{cases} access\ to\ type\ A \\ p \leq p_1 \end{cases}$$

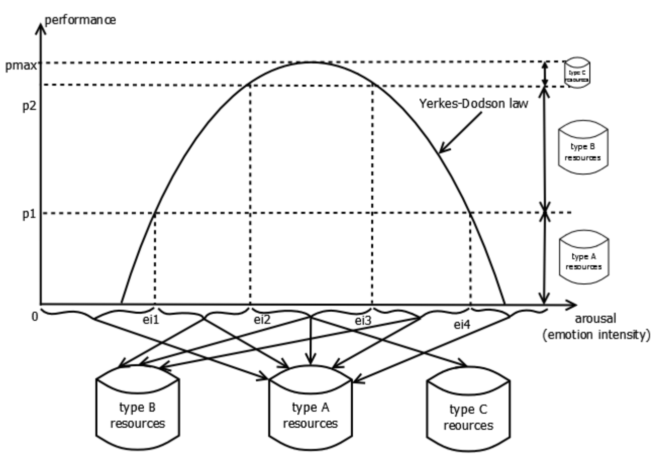


Fig. 4. Agent Accessibility To Its Resources

C. Personality

We model personality through the four temperament types: sanguine, choleric, melancholic and phlegmatic. These are represented by the four areas delimited by two axes in a 2D space: extraversion axis (extrovert/introvert) and neuroticism axis (psychologically stable/unstable). We chose this model because of its simplicity and of a correspondence that can be made between the two axes and the parameters in our agent architecture. The mapping is shown in Fig. 5 and is explained below.

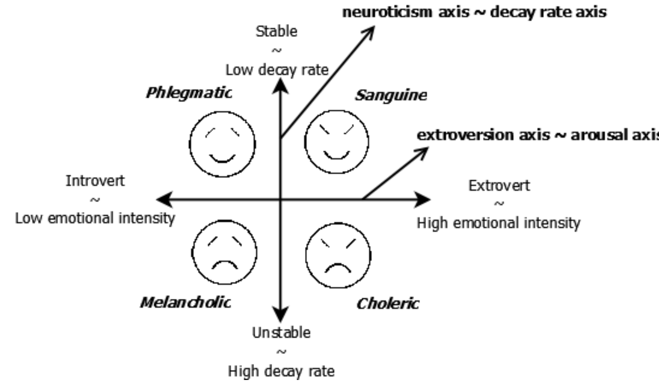


Fig. 5. The four temperaments

The introvert/extrovert axis is associated with the arousal (or emotional intensity). According to Eysenck [15], introverts are characterized by higher cortical arousal than extroverts, therefore the former require less stimuli from the environment (less emotional intentisy), while the latter seek situations which stimulate them more (give higher level of emotional intensity). The psychologically stable/unstable axis is associated with the decay rate of agents' emotions. Thus, a stable agent has a low decay rate, with a coherent affective state; an unstable agent has a high decay rate, quickly moving from one emotional state to another. Resuming, the four temperament types have the characteristics shown in TABLE 1:

TABLE 1. AGENT PERSONALITY

Temperament	Extraversion	Neuroticism	Characteristics
Sanguine	extrovert	stable	-high emotion intensity -low decay rate
Choleric	extrovert	unstable	-high emotion intensity -high decay rate
Melancholic	introvert	unstable	- low emotion intensity - high decay rate
Phlegmatic	introvert	stable	-low emotion intensity -low decay rate

The role of personality in the architecture is to define how emotions are updated. This is done through two variables: emotion intensity and decay rate. First of all, each agent has a predefined value for arousal: the sanguine and choleric have a lower value, the melancholic and phlegmatic have a higher value. When an event takes place or the state of the environment changes, each of this adds a certain emotional intensity to the default value of arousal. Thus, in the 2D emotional space the emotion point is shifted up for the introverts over the extroverts. For example, if the execution of the plan fails, the extrovert might be annoyed, but the introvert might be angry (annoyed and angry represent the same basic emotion, but at different intensities). The second variable is also predefined and represents the value that the emotion is decayed with at each step. If an agent is angry, but then an intention is achieved, the melancholic and the choleric might forget the anger, feeling joyful, but the sanguine and the phlegmatic will not forget the anger so rapidly and will not pass so rapidly to the joyful state.

Personality also influences the performance of the agent. To capture this, we need to take a look at Yerkes-Dodson Law of arousal [14]. This law states that performance increases with the increase of arousal, but up to a point where the performance is maximum. After that, performance drops if arousal keeps increasing. The value of arousal for which performance is optimum is different for different personalities. Thus, for extroverts the graph is shifted to the right comparing to the graph for introverts, as shown in Fig. 6.

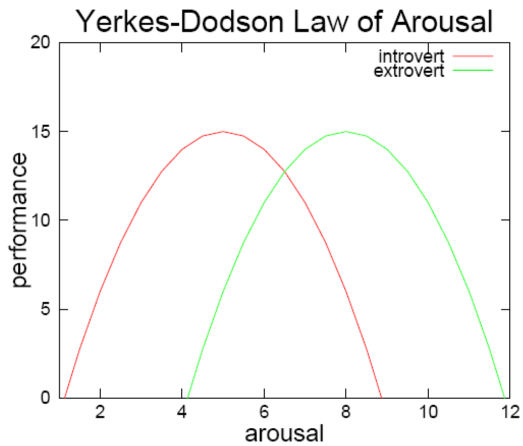


Fig. 6. The Yerkes-Dodson Law Of Arousal For Introverts And For Extroverts

IV. Previous Work

Early work was done by Padgham and Taylor [5]. They built a system in which the reasoning is done by the Distributed Multi Agent System (dMars) [4]. What is specific for their model is that each agent has a given threshold for the emotional gauge, and when this is passed, the *belief* that the agent feels that particular emotion *is asserted*. Therefore, the main problem of this architecture is that emotions do not directly influence the reasoning process, but are considered only as beliefs. Nevertheless, the first step of integrating agent emotions and BDI agent model into the same system was done. Consequent research goes into more detail on the relationship between emotions and the reasoning process.

Pereira et al. [6] describes a conceptual model of an Emotional BDI agent which aims at identifying the disadvantages of the BDI model and to overcome them. The solution was to add new concepts to the model. Thus, their architecture contains resources, capabilities, a module for sensing and perception and a module for managing the emotional state. While resources and capabilities are simply ”internal representations of the means that an agent has to execute upon its environment” (concrete means and abstract plans of action, respectively), the emotional state manager is a module that generates the emotions felt by the agent. Nevertheless, it is not fully specified, the authors mentioning only the main characteristics that this module should include: a well-defined set of Artificial Emotions, various triggering events and a decay rate for each Artificial Emotion. Although the details of the emotional module are left for further consideration, the idea of introducing resources and capabilities in the BDI model is worth to be considered.

The following year, independent of [6], Parunak et al. [7] describes the DETT model for situated agents applicable in the particular situation of a combat. Their aim is to simulate faster than real-time a large number of combatants. The architecture includes two reasoning processes (appraisal and analysis) and four new concepts (Dispositions, Emotions, Triggers and Tendencies). The appraisal process assesses the beliefs in the context of agent *disposition* and returns the *emotion* felt. The beliefs are mapped to digital pheromones which inform the presence of other agents or objects in the environment. These pheromones act as *triggers* for emotions. In their turn, emotions impose a *tendency* on the intentions that are to be selected by the analysis process. Although it is important for an agent to be able to act in real time, we would like to build an architecture that is not specific to a particular situation.

A more sophisticated model was presented by Jiang et al. [8]. They focus on the influence of emotions on the decision making process. The novelty of their EBDI architecture consists in considering primary and secondary emotions, as well as three sources for beliefs: perception, communication and contemplation. Primary emotions are connected to reactive behavior and fast decisions. If time permits, further deliberation is conducted and secondary emotions generated. The authors believe resources and capabilities introduced by Pereira et al. [6] to be unnecessary. Instead, they solve the resource boundary condition by adding priorities for beliefs, desires, intentions and emotions, deleting those with lowest priority when memory is full. On the other hand, they believe the problem of reconsideration to be context specific, so they leave it for the plan execution function to deal with it. The authors also criticize [6] for not emphasizing the differences between the emotional agent and other agents. In contrast, they have implemented their architecture and tested it in Tileworld using one emotional agent and two non-emotional agents. Results show that the first has better results than the latter.

The most recent emotional BDI architecture is that of Jones et al. [9], called PEP-BDI. The authors criticize Jiang et al. [8] for missing personality and physiology aspects and Parunak et al. [7] for modeling only two emotions in relation with two personality aspects. Their main motivation lies in the context of global security, and their goal is to simulate a crisis situation in a virtual reality environment. They add emotions, personality and physiology to the BDI model, stating that these are key concepts in the context of crisis management. The paper illustrates the decision making process on a scenario which involves escaping from a fire. One should note that physiological parameters are not necessary in a non-crisis scenario, but resources are a more suitable component to consider.

In parallel with the above mentioned systems, two more architectures were developed, both in the topic of emotional BDI models, but each one independent of previous works and each one independent of one another.

One work is that of Hernandez et al. [10]. The authors develop a modular architecture, named BDIE, which contains four separate modules: Perceptual System, Motivational System, Behavior System and Emotional System – mapped to Beliefs, Desires, Intentions and Emotions, respectively.

Beliefs, evaluators and emotions are organized on two levels. First level beliefs are acquired through perception and are then affectively appraised by the first level evaluators, giving the primary emotions (fear, surprise). If one of these is active, control passes to the planning algorithm (contained in the Behavior System). Otherwise, second level beliefs are inferred from the first level beliefs and are cognitively appraised by the second level evaluators, giving the secondary emotions (happiness, sadness, anger). Desires comprise goals (need to be achieved) and homeostatic variables (need to be maintained). However, the paper doesn't explain how the agent chooses its desires and intentions; in the scenario considered the only desire is survival and the only behavior is to display facial expressions according to the emotions induced by certain pictures. The modular structure of the system can be appointed as the most interesting aspect of the BDIE architecture.

The second architecture, called BDE, was developed by Florea and Kalisz [11] on the hypothesis that emotions contribute to behavior anticipation, which in turn contributes to making an agent more realistic. The authors show that an event may generate several emotions, which are then integrated into an emotional state using a rule-based approach. Also, the authors account for both emotion decaying and emotional memory; thus, powerful emotions remain in memory even if they are momentarily decayed. The paper specifies the emotion eliciting conditions (EEC) and the influence of emotional state on behavior (IEB) for seven emotions. Although the architecture is not very complex, by not considering all the details of the human mechanisms, the simplicity of the BDE model makes it more appropriate in not so dynamic environments.

TABLE 2 summarizes the characteristics of the presented models. In what follows, we detail the similarities and differences among them.

First of all, one can note that each paper is concerned with different issues:

- Pereira et al. [6] - BDI improvement
- Parunak et al. [7] - faster-than-real-time, large number of combatants
- Jiang et al. [8] - improved decision-making
- Jones et al. [9] - handling crisis situations
- Hernandez et al. [10] - performance, HCI
- Florea and Kalisz [11] - behavior anticipation

Secondly, there are also obvious differences with respect to the architecture components. In [8] beliefs are acquired using three different methods: stimuli in the environment, messages from other agents and contemplation. [6], [7], [10] and [11] don't take into consideration communication, but use one belief revision function that includes both new percepts and current beliefs. The same goes with [9], but percepts can be obtained through stimuli, messages and physiological parameters (which define the agent's health).

Next, the reasoning process happens differently. In [7] desires are constant over time, while in [11] emotions (and only emotions) influence them. The deliberation process updates the intentions, and the means-end reasoning process structures these intentions into plans. For [11] and [7], desires are predefined and need to be filtered only if they are inconsistent.

A set of consistent desires forms the goals of the agent. [6], [8] and [9] consider desires as options that need to be constantly generated based on current desires and beliefs. In [8] and [9], deliberation process is formed by the options function (generating desires) and the filter function (generating intentions); the means-end reasoning is done by the plan function, which structures intentions into plans. In [6] on the other hand, the first process that occurs is generate options, which maps to means-end reasoning; this generates desires and intentions that hierarchically flow from abstract to concrete, until executable actions are obtained. Then, the deliberation process (filter) chooses those intentions that the agent will be committed to.

What all the papers have in common is the focus on the influence of emotions over deliberation and means-end reasoning. How the affective state affects the planning algorithm is not a priority for any of the authors. The commitment strategy is yet less discussed, although the trade-off between reconsideration and the degree of commitment is an important aspect of practical reasoning. From the agent control loops presented it can be inferred that [11] uses an open-minded strategy (the agent is committed to an intention as long as it is still believed possible), while [6], [8] and [9] use a single-minded strategy (the agent is committed to an intention until it is either realized or not possible).

Last, but not least, [6] only describes a conceptual architecture, [9] shows a fire scenario and [11] exemplifies the emotion generation and influence over behavior for seven basic emotions. On the other hand, [7], [8] and [10] implement their agents and show experimental results. Thus, [8] runs both emotional and non-emotional agents in Tileworld, showing that the EBDI agent has a better performance; [10] implement their architecture on a robotic head which changes its facial expression based on color and luminance of an image; and [7] creates an architecture which is specific to a combat situation, mapping to concrete beliefs, desires and emotions.

We built our architecture based on the work presented above, focusing on agent performance, on simulating human mechanisms of internal resource usage and on agent personality. We were mostly attracted by the EBDI model [8] for emphasizing the influence of emotions on decision making and for taking into account both primary and secondary emotions. Additionally, we are interested in the model of Pereira et al. [6] for adding resources and in the PEP→BDI model [9] for considering the personality influences on agent behavior. Ultimately, we valued the usage of the OCC theory on beliefs, desires and intentions in the BDE model [11].

TABLE 2. COMPARISON OF EXISTING EMOTIONAL BDI SYSTEMS

Components / Model	Belief	Desire	Intention	Emotion	Other
Pereira et al., 2005 [6]	- acquired through perception &inference - one revision function	- revised through distinct algorithms, depending on resources, capabilities and emotional state	- revised through distinct algorithms, depending on resources, capabilities and emotional state	- Emotional State Manager - not detailed	- resources - capabilities - Sensing and Perception Module
Parunak et al., 2006 [7] (DETT)	- mapped to digital pheromones	- wants - constant over time	- result of analysis process - tendency imposed by emotions	- result of appraisal process (OCC) - triggered by beliefs, depending on disposition	- Disposition, Emotion, Trigger, Tendency
Jiang et al., 2007 [8] (EBDI)	- acquired through perception, communication & contemplation - three revision functions	- not influenced by emotions	- influenced by emotions, desires & intentions - state of affairs that the agent has committed to achieve	- primary - secondary	-
Jones et al., 2009 [9] (PEP-BDI)	- acquired through perception, communication &physiology - one revision function, three perception functions	- options - influenced by beliefs, intentions, personality and physiology, but not by emotions	- filtered options - influenced by beliefs, desires, intentions, emotions and physiology, but not by personality	- primary - secondary	- Personality, Emotion, Physiology
Hernandez et al., 2004 [10] (BDIE)	- first level (acquired through perception) - second level (acquired through first level belief revision)	- goals (to be achieved) - homeostatic variables (to be maintained)	- contains the planning algorithm - connected with emotions through goals	- primary & secondary - emotional space divided into emotional sectors	-
Florea and Kalisz, 2004 [11] (BDE)	- acquired through perception & inference - one revision function	- influenced only by the emotional state	- course of action to be taken in order to achieve desires	- OCC - emotional state - emotional memory - decay	-

V. SCENARIO

The story is that of a man that wants to gather the garbage in a city, with garbage being well-defined items that can be collected by that man. Because he is walking, he risks to be bitten by dogs that have no master and wander freely on the streets. He thus has to be careful to avoid the dogs. The man doesn't know where the garbage items are, new items being possible to appear anytime and anywhere in the city, but other people may tell him about certain item locations.

We can model this story with an agent situated in a dynamic environment, a 2D map which contains artifacts, but also traps. The agent has the goal of gathering as many artifacts as possible, taking care to avoid the traps. The dynamism of the environment is given by the random appearance of the artifacts and by the random appearance, disappearance and moving of the traps. The agent can see only a fixed number of cells in front of him, but it can remember the position of the artifacts and the traps.

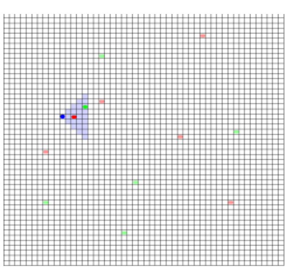


Fig. 7. Example Of A Map For The Proposed Scenario

In Fig. 9, the agent is represented by the blue circle. The light blue triangle besides the agent shows what the agent can see from its current position. The green ovals are the artifacts and the red ovals are the traps. The artifacts appear random on the map. If their number decreases below a specified value (because of being collected by the agent), other artifacts randomly appear on the map. Traps may appear and disappear randomly, and they also have a random move (they are more like non-rational agents which wander on the map with no purpose whatsoever). An agent is said to *fall into a trap* if it steps into a cell simultaneously with a trap. If this is the case, its energy level decreases. Traps cannot appear on the cell that the agent is currently standing, neither on a cell where there is already an artifact. TABLE 3 shows how the components integrated into the architecture are mapped to the current scenario.

In this scenario, resources refer to the possible actions that an agent may take. The agent has a specified energy level that is decreased when using resources (more or less and each type of resource used consumes more or less of this energy. The same energy is consumed when falling into a trap. The agent has the possibility to restore its energy level by taking the action *sit* (analogous to the situation in which one has to relax to recover from a great effort).

Depending on the personality, the agent may pass from one emotion to another faster or slower. For example, if it fell into a trap and is angry, but shortly after it collects an artifact, a choleric agent will be joyful, forgetting the anger, while a phlegmatic one will feel only slight joy, still alert to traps (low decay rate for phlegmatic, high decay rate for choleric). The belief that a trap is close will generate fear in a melancholic agent, but a sanguine agent will feel fear only if the trap is much closer (the melancholic has a high default arousal, so small increase in emotion intensity will make it alert, while the sanguine needs higher arousal level to actually feel the new emotion).

TABLE 3. INSTANTIATION OF THE AGENT COMPONENTS

Percepts	What the agent has in sight (clear cell, artifact, trap) Messages from other agents containing artifacts or traps coordinates (not shown in the figure)
Beliefs	Agent coordinates Artifact coordinates Traps coordinates
Desires	Collect artifacts Avoid traps
Options	Move to the artifacts with known position
Intentions	Move to the selected artifact(s)
Emotions	Joy – artifact collected (goal accomplished) Fear – trap near (negative belief) Anger – fall into trap (goal not accomplished) Relief – trap disappears from sight (positive belief)
Personality	One of Choleric / Melancholic / Sanguine / Phlegmatic
Resources	Possible moves: Type A: turn 90°, move forward 1 cell, sit Type B: jump left / right / behind 1 cell Type C: jump forward over trap (2 cells)

VI. CONCLUSION

In this paper we presented an emotional BDI architecture which accounts for agent resources and personality. We presented the agent control loop, the emotional mechanisms, the resource usage and the personality influence on agent behavior. The main contribution is that the agent built using this architecture has both cognitive and reactive behavior. The BDI architecture focuses on the cognitive functions of the brain, but in a very dynamic environment the reactive behavior gives the agent the possibility to quickly deal with unexpected events. For this reason, we believe our architecture should improve agent performance. Moreover, by considering the influence of both primary and secondary emotions, we can faithfully replicate the human reasoning process, which is a step forward into giving agents human intelligence.

We based our work on previous research, which we have reviewed briefly, showing similarities and differences and noting the features that inspired us. Thus, we were specifically attracted by the EBDI model [8], because of its proved improved performance; also, the model of Pereira et al. [6] is interesting for including resources in the architecture and the PEP→BDI model [9] for including personality; last, but not least, the cognitive appraisal theory applied on beliefs, desires and intentions, as described in the BDE model [11].

Further on we need to implement this architecture and to actually test it on the described scenario. Other scenarios will also have to be developed and implemented, so that the agent can be tested in different situations.

REFERENCES

[1] Rao, A. and Georgeff, M. (1991). Modeling rational agents within a BDI-architecture. In Proceedings of Knowledge Representation and Reasoning(KR&R-91), pages 473–484.

[2] Bratman, M., Israel, F., and Pollack, M. (1988). Plans and resource-bounded practical reasoning. Computational Intelligence.

[3] Damasio, A. (1994). Descartes' Error: Emotion, Reason and the Human Brain. Gosset/Putnam Press.

[4] D'Inverno, M., Luck, M., Georgeff, M., Kinny, D. and Wooldridge, M. (2004) The dMARS Architecture: A Specification of the Distributed Multi-Agent Reasoning System. Autonomous Agents and Multi-Agent Systems, 9(1-2):5-53.

[5] Padgham, L. and Taylor, G. (1997). A system for modelling agents having emotion and personality. In Proceedings of PRICAI Workshop on Intelligent Agent Systems: Theoretical and Practical Issues, Lecture Notes in AI 1209, pages 59–71. Springer-Verlang.

[6] Pereira, D., Oliveira, E., Moreira, N., and Sarmento, L. (2005). Towards an architecture for emotional BDI agents. In IEEE Proceedings of the 12th Portuguese Conference on Artificial Intelligence (EPIA'05), pages 40–47. Springer.

[7] Parunak, H. V. D., Bisson, R., Brueckner, S., Matthews, R., and Sauter, J. (2006). A model of emotions for situated agents. In Proceedings of 5th International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'06), pages 993–995.

[8] Jiang, H., Vidal, J., and Huhns, M. (2007). EBDI: An architecture for emotional agents. In Proceedings of the 6th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'07), pages 38–40.

[9] Jones, H., Saunier, J., and Lourdeaux, D. (2009). Personality, emotions and physiology in a bdi agent architecture: the PEP→BDI model. In Proceedings of WI-IAT - IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology, volume 2, pages 263–266.

[10] Hernandez, D., Deniz, O., Lorenzo, J., and Hernandez, M. (2004). Bdie: a bdi like architecture with emotional capabilities. In Papers from AAAI Spring Symposium: Architectures for Modeling Emotion: Cross-Disciplinary Foundations.

[11] Florea, A. M. and Kalisz, E. (2004). Behavior anticipation based on beliefs, desires and emotions. International Journal of Computing Anticipatory Systems, 14:37–47.

[12] Ortony, A., Clore, G., and Collins, A. (1988). The cognitive structure of emotions. Cambridge University Press.

[13] Russel, J. (1980). A circumplex model of affect. Journal of Personality and Social Psychology, 39(6):1161–1178.

[14] Yerkes, R. and Dodson, J. (1908). The relationship of strength of stimulus to rapidity of habit formation. Journal of Comparative Neurology and Psychology, (18):459–482.

[15] Eysenck H. (1967). The biological basis of personality. Springfield, IL: Thomas Publishing.

AUTHORS PROFILE

Mihaela-Alexandra Puică received her BSc in Computer Science in 2009 and MSc in Artificial Intelligence in 2011, both from University Politehnica of Bucharest, Faculty of Automatic Control and Computers. She is now a PhD student at the same university, doing her research in the field of Affective Computing, both emotion simulation and emotion recognition. She is also a member of the Artificial Intelligence and Multi-Agent Systems (AI-MAS) Laboratory.

Adina-Magda Florea is a Professor at the Computer Science and Engineering Department from the University Politehnica of Bucharest, Faculty of Automatic Control and Computers. She is teaching Artificial Intelligence, Machine Learning, Knowledge Representation and Reasoning and Multi-Agent Systems. She is head of the Artificial Intelligence and Multi-Agent Systems (AI-MAS) Laboratory. Her research interests include Multi-Agent Systems, Ambient Intelligence, Affective Computing and E-learning.

A Kabbalah System Theory of Ontological and Knowledge Engineering for Knowledge Based Systems

Gabriel Burstein, Constantin Virgil Negoita
Department of Computer Science
Hunter College, City University of New York (CUNY)
New York, USA

***Abstract*—Using the Kabbalah system theory (KST) developed in [1], [2], we propose an ontological engineering for knowledge representation of domains in terms of concept systems in knowledge based systems in artificial intelligence. KST is also used for the knowledge engineering of the knowledge model building based on ontology. KST provides thus an integrative, unifying, domain independent framework for both the knowledge representation via ontologies and knowledge model building via knowledge engineering in knowledge based systems.**

Keywords—knowledge based system; knowledge representation; ontological engineering; knowledge modeling; knowledge engineering; artificial intelligence; Kabbalah; system theory; category theory

I. Introduction

One of the difficulties in the knowledge engineering (KE) of knowledge based systems (KBS) in artificial intelligence (AI) is the fact that knowledge is domain dependent. This means that knowledge based systems are in general domain dependent, knowledge bases are domain specific and cannot be re-used in general in another domain, a drawback that applies for early AI constructs such as semantic networks and expert systems. Ontological engineering [3],[4] is trying to solve that problem by building general ontologies defined as systems of generic concepts, their attributes and properties and relations between concepts. Individual concepts and objects are seen as instances of the generic concept. Such ontologies can serve as a common knowledge base on which to develop specific knowledge model building. The idea of ontological engineering is inspired from ontology, the philosophy of existence and its categories and relations between them. Aristotle 10 category based ontology (upper level ontology to be more precise) is one of the earliest well known ontologies [3].

The other main difficulty in KBS and AI in general is in capturing together in knowledge representation 1) the cognitive, 2) the behavioral / emotional and 3) the action / implementation and articulation levels of human knowledge in knowledge representation. This has been an obstacle to AI progress given that most of the focus has been on formal rules, learning models and knowledge modeling at levels 1 and / or 3 above only, the cognitive and action, implementation and articulation levels, missing completely the essential level 2, the behavioral and emotional together with its interconnections with levels 1 and 3. Human knowledge is cognitive and action based as it is emotions and behavior based.

In [1],[2] we introduced a Kabbalah system theory based on exploring and formalizing principles of the philosophy of Kabbalah [5],[6],[7] using system theory [8] in a mathematical category theoretic formulation [9], [10], [11], [12].

Kabbalah is an ancient philosophy of the creation and existence developed over centuries by Arizal (Ari), Rashash (Shalom Sharabi), Ramhal (M.H. ben Luzzatto), Yehudah Ashlag, Shimon bar Yochai [5], [6], [7]. Its central concept, the Tree of Life [7], integrates together the cognitive, emotional / behavioral and action / implementation levels of existence. The Tree of Life actually has three interconnected levels and they are exactly the cognitive, the emotional / behavioral and the action level.

We show here how to create a Kabbalistic Tree of Life general model or representation for any concept in any knowledge domain but also for a system of concepts and the relation between them (ontology). The Tree of Life of Kabbalah becomes thus a standardized model for both concepts and ontologies. This leads to building a Kabbalah based ontological engineering for knowledge representation in KBS using our Kabbalah system theory (KST) [1], [2]. KST, through its system dynamics and category theoretic instruments like pullback and pushout, is also the framework for knowledge engineering and knowledge model building describing concept formation and dynamics, knowledge ontology dynamics and transformation etc.

The advantage of our Kabbalah ontology is using a standardized architecture, the Tree of Life, whereas in ontological engineering every domain has a different type of ontology architecture. Every concept will be represented by a Tree of Life and ontologies as systems of concepts will also be represented as a tree of Life etc.

II. A Kabbalah Based Ontology Using the Tree of Life Architecture

According to the Kabbalah philosophy, existence can be described in terms of 10 fundamental attributes / qualities or basic structural components called “sefirot” in Hebrew / Aramaic which means counts (each one is called a sefira), grouped in three categories [7]:

- Cognitive level (including objective, spiritual knowledge, meta-knowledge) : Crown (will, faith and desire, meta-knowledge level called Keter in original Hebrew or Aramaic), Wisdom (idea called Chochmah in Hebrew / Aramaic), Understanding (Binah) and Knowledge (Da'at) which in fact prepares the transition and implementation of understanding at the emotional level. We are not going here into the detailed structure of this sefira, we did so in [1].

- Emotional / behavioral level: Lovingkindness (Chesed), Judgment, Strength, Rigor or Severity (Gevurah) and Harmony or Beauty (Tiferet) which is connected to the next level below

- Action, implementation, articulation level: Perseverance or Endurance (Netzach), Victory or Majesty (Hod), Foundation (Yesod) and Kingship (Malchut)

Despite their metaphorical anthropomorphic names, they do represent a very general metaphoric coordinate system of 10 general basic attributes (11 sefirot including Knowledge which normally is not represented in the same time with Crown, we will often refer to this as "10+1 sefirot"), properties, attributes that can be used to describe complex systems in general and existence. In fact, ontology attempts to find a sort of philosophical, generalized, abstract Cartesian-like system of coordinates for existence. From this point of view Kabbalah can be seen as one of the first attempts to create ontology to describe existence, creation and creatures. In the Tree of Life, the ten sefirot fundamental units or components are interconnected by 22 arcs based on the interactions between them and between each of the three fundamental levels described above, in which these sefirot are integrated.

The internal sub-structure of each sefira is again of the type of a Tree of Life made of 10 sub-sefirot of the same type as the original 10 sefirot. This way, each sefira contains an internal model of the Tree of Life and of each of the sefirot it is in interaction with. In principle, we can go on and speak of the sub-sub-structure of sub-sefirot which will also be in the shape of Tree of Life etc. This means that the Tree of Life has a fractal structure or an inter-inclusive structure. However, for purposes of our Kabbalah system theory we will restrict ourselves to the first order sub-structure of the Tree of Life described by sub-sefirot of sefirot.

We can use the Tree of Life structure to create here a model, representation or architecture for a concept:

- Cognitive level of a concept: Crown (Keter) "K"- Concept meta-knowledge, Wisdom (Chochmah) "C" - Idea of the concept, Understanding (Binah) "B" – the meaning, essence and understanding of a concept, Knowledge (Da'at) "D" – Practical relevance of a concept. This level is denoted by CBD or ChaBaD.

- Emotional, behavioral level of a concept: Lovingkindness (Chesed) "C'" - synonymous to the concept, the is-like the concept, Strength, Judgment (Gevurah) "G" – Strength, rules, the "is-not-like" a concept, antonymous to the concept, Harmony (Tiferet) "T" – Beauty, balance, harmony of a concept. This level is denoted by C'GT or ChaGaT.

- Action, implementation, articulation level of a concept : Endurance (Netzach) "N" – Endurance of a concept, continuity of a concept, concept as an object, permanent entity, Victory or Majesty (Hod) "H" – Concept occurrence and perdurance [3], discontinuity of a concept, concept as a process on and off, Foundation (Yesod) "Y" – Concept re-actualization, updating, renewal, channeling towards implementation and Kingship (Malchut) "M" – Practical manifestation and implementation of a concept. This level is denoted by NHY.

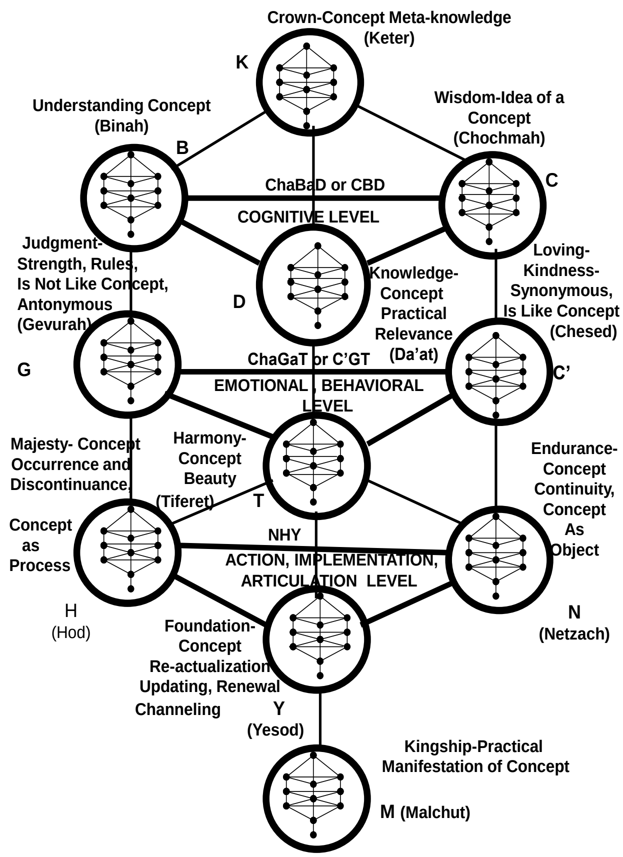


Fig. 1. A Kaballah ontological model for a concept based on a Tree of Life representation: the Tree of Life has three levels (cognitive, behavioral / emotional and action / implementation) and 10+1 basic components called "sefirot". A concept is represented in terms of its 10+1 "sefirot" main sub-concepts, coordinates, components, properties, attributes. Each main coordinate or component of a concept has an internal Tree of Life sub-structure representation..

We can now represent our new Kabbalah ontological model of a concept based on the Tree of Life given in Fig. 1. A concept is represented as a Tree of Life of it main attributes, properties, sub-concepts grouped in the 10+1 Sefirot of a Tree of Life acting as a generalized coordinate system. Each sub-concept or attribute, each property is also represented by its internal sub-Tree of Life structure

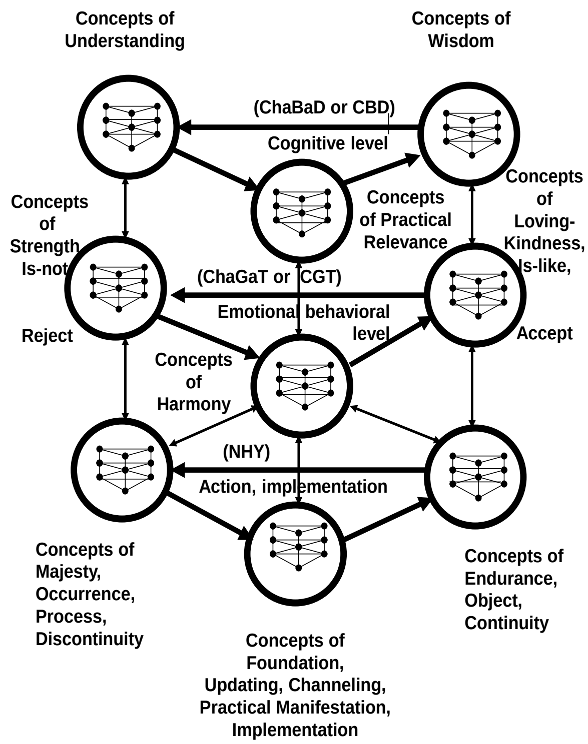


Fig. 2. A Kabbalah system theory based ontological engineering model for an ontology seen as a system of concepts, their properties, attributes and relations between concepts. An ontology is represented by the simplified form of the Tree of Life. Here each sefirot represents a concept or class of concepts which has an internal simplified sub-Tree of Life structure too. According to our Kabbalah system theory, this simplified form of the Tree of Life is modeled by a hierarchical 3-level feedback control system.

The Tree of Life of a concept can be represented in a simplified form showing only its three triadic levels that we explained above which were denoted in Fig. 1 by CBD (ChaBaD), C'GT (ChaGaT) and NHY, notation based on the initials of the three sefirot at each level. Such simplified representation is often used in Kabbalah to focus on the three interconnected hierarchic levels and is called in Hebrew "Drush HaDaat".

We can build next our new Kabbalah based ontological engineering model for an ontology seen as a system of concepts, their properties, attributes and relations between concepts. An ontology will be represented by the simplified form of the Tree of Life as in Fig. 2. Here each sefirot represents a class, category of concepts of the ontology which is also modeled by an internal simplified sub-Tree of Life structure.

We developed in [1], [2] a Kabbalah system theory of the simplified form of the Tree of Life given in Fig. 2. System theory emerged as a program to address, in a holistic rather than reductionist way driving and controlling or control engineering different type of systems and the interdependence between these in a unified formalism [8]: input or cause, output or effect, states, feedback from states or output back to input called control or feedback control meant to drive the system to a desired state or objective etc. System theory has been so far very little applied to AI, knowledge engineering, ontological engineering yet all these are called "engineering". We will introduce here a system theoretic interpretation of knowledge engineering seen as achieving desired outputs and objectives based on driving, controlling a knowledge based model, an ontology etc which is the general objective of control system engineering in general. Our Kabbalah system theory is based on modeling the simplified Tree of Life in Fig. 2 as a 3-level feedback control system: the three triadic levels discussed before CBD, C'GT. NHY operate each as a feedback loop with the mid-line sefirot D, T, Y acting as feedback "controllers" moderating, modulating balancing the interaction between respectively C and B, C' and G, N and H. These three feedback control systems are vertically interconnected both ways, up and down, to achieve hierarchic feedback control [1].

One of the main merits of the Kabbalah system theory approach developed in [1], [2] is that it is both holistic and reductionist. System theory approach is holistic which leads to de-emphasizing reductionist aspects and this was a drawback. Postmodern approaches aim to restore the emphasis on the reductionist, de-constructivist aspects [1]. Our Kabbalah system theory is both holistic and reductionist given that each sefirot is "deconstructed" in detail through its sub – Tree of life model. From this point of view, Kabbalah system theory is a postmodern system theory. The structure of the reductionist model is the same as the holistic model due to the inter-inclusive structure of the Tree of Life.

III. A Kabbalah System Theory Approach to Ontological Engineering, Knowledge Engineering And Knowledge Based Modeling

The main feature of the system theoretic approach is the emphasis on cause-effect relations and the dynamics of systems. Ontologies are in general static, represented by hierarchically branching trees yet one wants a dynamic ontological engineering to act as a base for knowledge based modeling and knowledge engineering [3]. All these are dynamic in nature and so is learning, concept formation and evolution etc.

This is why we concentrate in this section on developing Kabbalah system theory dynamic models for the ontological engineering framework in Fig. 2. These models will be in fact Kabbalah system theory models for knowledge engineering and knowledge based modeling rooted in the ontological engineering model of Fig. 2. We will provide thus a common, unifying, integrative framework for ontological engineering and knowledge based modeling / knowledge engineering.

The first step is to model the dynamics of the ontology backcloth or foundation, concept formation dynamics within the ontology Tree of Life model from Fig. 2. For this purpose we introduced in [1] the mathematical category theoretic formalism as the mathematics behind our Kabbalah system theory. This is based on the operations of pullback and pushout which we need to re-define here for ease of understanding. A category is made of objects, morphisms (maps) between objects and morphism composition [9], [10].

Definition 1 The pushout of objects A and B over object C in a category containing objects A,B,C connected by the morphisms g: C $\rightarrow$ B , f: C $\rightarrow$ A, is an object PO of that category together with morphisms n: B → PO and m: A → PO in the category morphism set such that i) the diagram in Fig. 4 commutes m o f = n o g (where "o" denotes morphism composition) and ii) PO has the universality property meaning that for any other object P in the category and morphisms m': A → P and n' : B → P that satisfy the commutativity of the diagram in Fig 3 m'o f = n' o g, there exists a unique morphism p: PO → P such that p o m = m' and p o n = n' (see Fig. 3).

We introduce next pullback, dual to pushout, by y-define here for ease of understanding.abbalah system theory. This is based on the operations of pullback and pushout which wreversing the morphism arrows in Definition 1.

Definition 2 The pullback of objects A and B over object C in a category containing objects A,B,C connected by the morphisms g: B → C , f: A → C, is an object PB of that category together with morphisms n: PB → B and m: PB → A such that i) the diagram in Fig. 4 "commutes" that is f o m = g o n where "o" denotes morphism composition and ii) PB has the universality property meaning that for any other object, P, in the category and morphisms m': P → A and n' : P → B that satisfy the "commutativity" of the diagram in Fig 4 that is f o m' = g o n', there exists a unique morphism p: P → PB such that m o p = m' and n o p = n' (see Fig. 4).

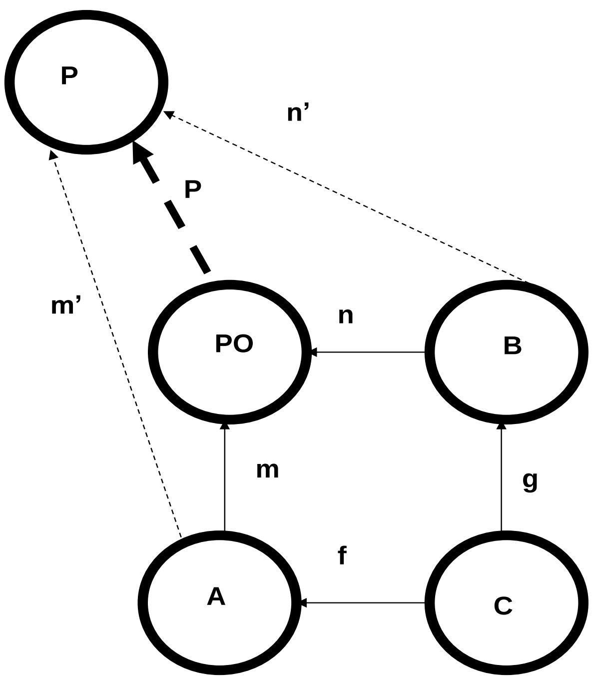


Fig. 3. The category theoretic commuting diagram definition of the pushout PO of objects A and B over C including the universality property (stability, robustness) of PO with respect to any other P. Objects can be concepts, sefirot, classes or categories of concepts etc.

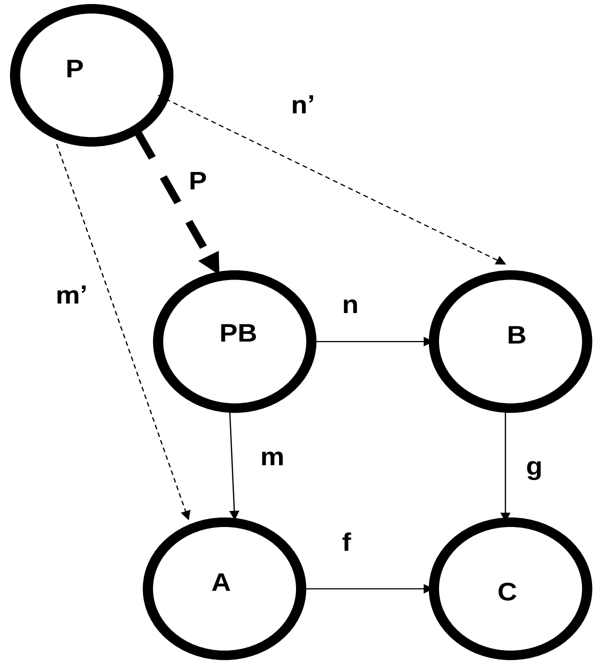


Fig. 4. The category theoretic commuting diagram definition of the pullback PB of A and B over C including the universality property (stability, robustness) of PB with respect to any other P. Objects A, B, C can be concepts, sub-concepts.

The Tree of Life in Fig. 2 can be seen as a diagram in the category of graphs and morphisms between graphs mapping vertices to vertices and edges to edges. Each sefirot corresponding to a class or category of concepts is represented by its own Tree of Life graph and the connections between sefirot as classes or categories of concepts can be modeled by morphisms between graphs. By considering the levels CBD, C'GT, NHY as 2-dimensional faces we can model the sub-Tree of Life of each concept or concept class as a simplicial complex which has vertices, edges or arcs and also two dimensional (triangle) faces CBD, C'GT, NHY. The Tree of Life in Fig. 2 can then be seen as a commuting diagram in the category of simplicial complexes and simplicial maps between them [10]. We obtain the algebraic category theoretic model for the Kabbalah based ontology of Fig. 2 given by Fig. 5.

In the algebraic category theoretic framework of Fig. 5 for the Tree of Life ontology model in Fig. 2, we can define, impose or calculate pullbacks and pushouts across the Tree of Life based on Definitions 1 and 2. It was shown in [13] how pullbacks and pushouts can be used to model concept formation based on bottom up learning and generalization, induction (pushout) and top down validation or verification, deduction. The universality property in Definitions 1 and 2, applied to concepts as pullbacks/pushouts, gives the structural stability of concepts.

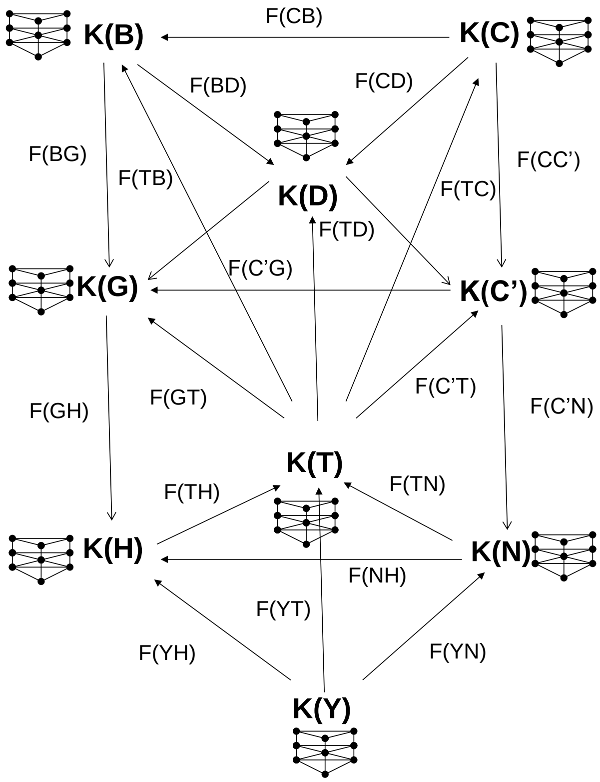


Fig. 5. Algebraic category theoretic model for the Kabbalah system theory based ontological engineering framework in Fig. 2: each concept or class of concepts of the ontology is represented by a simplicial complex of sub-concepts or attributes/properties and the connections between them are simplicial maps between simplicial complexes, denoted F(CB), F(BD),…based on the notation for the domain and co-domain sefirot. The overall diagram is a commuting diagram in the category of simplicial complexes .

In the Tree of Life ontology model in Fig. 2, pullback and pushout are a powerful model for concept formation dynamics across the ontology. One of the examples, is the construction, concept formation, of Harmony (Tiferet) concept, represented by K(T) in Fig. 5, as pullback / pushout of Wisdom (Chochmah) concept, K(C) in Fig. 5, and Understanding (Binah) concept, K(B) in Fig. 5, over Knowledge (Da'at), K(D) in Fig. 5. This was formulated in detail in [1], in general, not for concepts in particular. According to [11],[12], pullback describes system dynamic behavior while pushout is a model of system interconnection. The "blending" of concepts to yield new concepts can be modeled by pushout [14]

The dynamics of the Tree of Life ontology model can be modeled as we explained above, as a diagram in the category of simplicial complexes of each sefirot concept or class of concepts C, B, D….denoted K(C), K(B), K(D),…together with possible pullback and pushout square sub-diagrams within the Tree of Life commuting diagram in Fig. 5. The edges denote simplicial maps between simplicial complexes. This is ontology base dynamics.

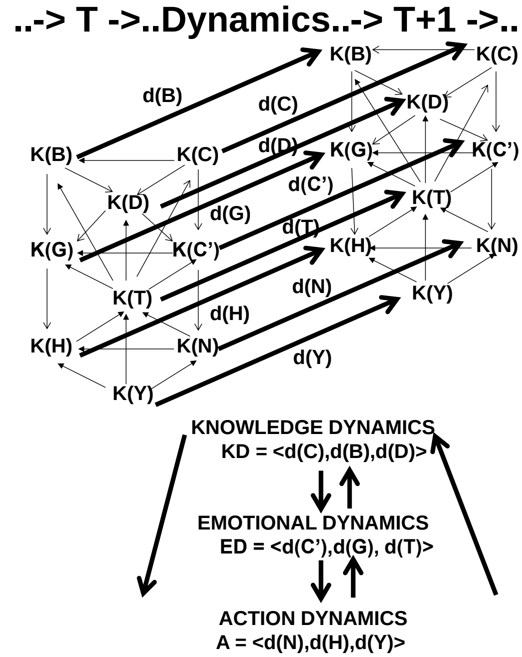


Fig. 6. Knowledge engineering and model building dynamics modeled as a transformation in time of the Kabbalah ontological model in Fig. 2 through a commuting diagram.

The knowledge engineering and knowledge modeling dynamics based on this dynamical ontology is given by the transition from the ontological Tree of Life at one stage to the next stage. This is represented by morphisms (maps) from the three levels of state of knowledge about the concepts (sefirot) at one stage to the three levels of state of knowledge about the concepts at the next stage, a sort of overall morphism between ontological Trees of Life yielding the overall commuting diagram in the category of simplicial complexes given in Fig. 6. We have in fact three sub-dynamics: cognitive, emotional and action, implementation dynamics describing dynamic transformations at each of the three levels of the Kabbalah ontological base model.

If we consider different ontologies modeled by two different Trees of Life in Fig. 6, then we have ontology mapping or transformations from one ontology to another mapping cognitive level to cognitive level, emotional / behavioral level to emotional / behavioral level and action level to action level.

IV. Conclusion

The Kabbalah system theory based ontological and knowledge engineering introduced here, based on the Kabbalah system theory [1],[2] unifies in a common framework both the concept formation dynamics in the ontological base and the dynamics associated with knowledge engineering and knowledge model building.

We have a longitudinal dynamics of the knowledge engineering and knowledge modeling, from one Tree of Life model to another whereas the pullbacks / pushouts of concept formation reflect transversal ontological base dynamics, across the Tree of Life base, due to learning, evolution at the level of the system of concepts of the ontology base.

ACKNOWLEDGMENT

The first author would like to express his gratitude for great teachings and guidance to R. Dangur, A. Babayove, M. Kranz, M. Handler, M. Millstein, N. Citron, S. Anteby, N. Blavin, A. Sutton, A. Greenbaum, R. Afilalo, D. Pinson, Y. Ginsburgh, E. Jungreis, S. Eisenberg, P. Greenberg, E. Buchwald and B.T. Krasnianski.

REFERENCES

[1] G. Burstein, and C.V. Negoita, “Foundations of a postmodern cybernetics based on Kabbalah”, Kybernetes, Vol. 40 No. 9-10, pp. 1331-1353, 2011.

[2] G. Burstein. and C.V. Negoita, “A Kabbalah system theory modeling framework for knowledge based behavioral economics and finance”, in Modeling and Simulation of Complex Social Systems, Intelligent Systems Reference Library series, V. Dabbaghian and V. Mago, Eds. New York: Springer Verlag, in press

[3] R. Mizoguchi, “Tutorial on ontological engineering, Part 1: Introduction to ontological engineering”, New Generation Computing, Vol. 21, No. 4, pp. 365-384, 2003

[4] A. Gomez-Perez, M. Fernandez-Lopez and O. Corcio, Ontological Engineering, in Advanced Information and Knowledge Processing Series, X. Wu and L. Jain, Eds. New York: Springer Verlag, 2010.

[5] R. Afilalo, The Kabbalah of the Arizal According to the Ramhal, Montreal: Kabbalah Editions, 2004.

[6] M.C. Luzzato, 138 Openings of Wisdom, translated by A.Y. Greenbaum. Jerusalem: The Azamra Institute, 2005.

[7] D.W. Menzi, and Z. Padeh, The Tree of Life: Chayyim Vital’s Introduction to the Kabbalah of Isaac Luria. New York: Arizal Publications, 2008.

[8] L. von Bertalanffy, General System Theory. New York: George Braziller, 1969.

[9] S. Mac Lane, Categories for the Working Mathematician. New York: Springer, 1998.

[10] S. Mac Lane, S. and I. Moerdijk, Sheaves in Geometry and Logic: A First Introduction to Topos Theory. New York: Springer-Verlag, 1992.

[11] J. Goguen, “Mathematical representation of hierarchically organized systems”, in Global Systems Dynamics, E. Attinger, (Ed.). New York: Wiley Interscience, 1970, pp. 112-128.

[12] J. Goguen, “Sheaf semantics for concurrent interacting objects”, Mathematical Structures in Computer Science, vol. 11, pp. 159-191. 1992.

[13] Y. Neuman. and O. Nave, “A mathematical theory of sign-mediated concept formation”, Applied Mathematics and Computation., Vol. 201, pp. 72-81, 2008.

[14] J. Goguen, “An introduction to algebraic semiotics with application to user interface design”, in Computation for Metaphor, Analogy and Agents, C. Nehaniv (Ed.). Berlin: Springer-Verlag, 1999, pp. 242-291.

Genetic Programming for Document Segmentation and Region Classification Using Discipulus

Priyadharshini N
Research Scholar
PSGR Krishnammal College for Women
Coimbatore, India

Vijaya MS
Associate Professor
GR Govindarajulu School Of Applied Computer Technology
Coimbatore, India

***Abstract*— Document segmentation is a method of rending the document into distinct regions. A document is an assortment of information and a standard mode of conveying information to others. Pursuance of data from documents involves ton of human effort, time intense and might severely prohibit the usage of data systems. So, automatic information pursuance from the document has become a big issue. It is been shown that document segmentation will facilitate to beat such problems. This paper proposes a new approach to segment and classify the document regions as text, image, drawings and table. Document image is divided into blocks using Run length smearing rule and features are extracted from every blocks. Discipulus tool has been used to construct the Genetic programming based classifier model and located 97.5% classification accuracy.**

Keywords—Document analysis; Information retrieval; Classification; Feature extraction; Document segmentation.

I. INTRODUCTION

Document segmentation is defined as a method of subdividing the document regions into text and non-text regions. A non-text region includes images, drawings, rules etc. Document segmentation plays a significant role in document analysis, because every day, millions of documents including technical reports, government files, Newspapers, books, magazines, letters, bank cheque, etc,. have to be processed. A lot of time, effort and money will be saved if it can be executed automatically. Automation of document analysis involves region extraction, identification of type of region and finally processing of each region separately. Document segmentation does the work of identifying the type of region. The documents contain both text and non-text regions. In order to process each region, the document should be segmented and then fed into the respective system for processing. For example, text regions are processed using OCR system.

It converts text region into machine-readable form and non-text regions are conserved for processing such as compression, enhancement, recognition, and storage etc. Some document segmentation applications are field extraction and recognition, word searching, logo detection, retrieving imaged documents in digital libraries, retrieval of documents containing tables or drawings. Also document segmentation is being adopted in postal industry where the address fields have to be identified before being sent to OCR readers and stamps have to be recognized.

Generally document regions are segmented in two ways namely, geometric and logical based segmentation. In geometric based segmentation, the document is segmented upon its geometric structure such as text and non-text regions. Whereas in logical segmentation the document is segmented upon its logical labels assigned to each region of the document such as title, logo, footnote, caption, etc., [2].

Till now lots of methods have been proposed for document segmentation in the literature. Document segmentation techniques are broadly classified into three categories: top-down, bottom-up and hybrid approach [2]. A top-down approach repetitively segments the document image into smaller regions until further it cannot be segmented. Run-length smearing algorithm, projection profile methods, Fourier transforms etc., [4, 5, 7] are the methods which make use of top down approach. A bottom-up approach begins by merging pixels into characters. Then the characters are merged into words until whole document regions are merged. The methods which follow this approach are connected component analysis [3], run-length smoothing [6], region-growing methods [4], and neural networks [8]. A hybrid approach is the combination of both top down and bottom up approach. Few hybrid based methods are texture based and Gabor filters. The advantage of using top-down approach is, its high speed processing and the drawback is, it cannot process table, improper layout documents and forms.

This research work associates the existing features specified in [4] [6] [8] and proposes few features which subsidizes more in document segmentation. Features such as perimeter/height ratio, energy, entropy are employed. Perimeter/height ratio is defined as a fraction perimeter to the height of the block. A block in document image is a connected component and it is defined as a collection of black runs that are 8-connected. Both perimeter and height of the block diverges in their values. Text blocks have slighter value for perimeter/height ratio when compared to non-text blocks. Thus perimeter/height ratio is essential in classifying the blocks.

Additionally energy and entropy features are used to classify the blocks. Energy and entropy are renowned properties of an image. Energy identifies the uniformity of the image. Whereas, entropy identifies the randomness (texture) of the image. Each block of the document varies in its energy and entropy specifically in case of table, drawings and image blocks. Thus these new features offer a notable influence in document segmentation.

This paper presents a genetic programming based document region classification. As genetic programming progress the classifiers in the manner of a program and measure the final classifier (program) for the classification conclusion this encourages in straightforward and speedy analysis of results. In this paper, genetic programming is implemented using commercial software called Discipulus. Discipulus writes computer programs automatically in most high level languages. Discipulus uses a multi-run evolutionary algorithm to evolve computer programs from the data. These evolved programs are high-precision models built from the data and map the inputs to the output. Unlike statistical techniques like neural networks, decision trees , discipulus builds models without any user tuning. It is self tuning and self parameterizing.

II. Proposed Model for Document Segmentation and Region Classification

The proposed approach reduces the computational complexities of supporting procedure after document segmentation such as OCR system for processing text regions and for processing non text region. It reformulates document segmentation problem as classification task and solved using genetic programming. Documents are collected from various journals and features are extracted from document images. The proposed method consists of four phases: preprocessing, segmentation using Run length smearing algorithm (RLSA), labeling connected components and feature extraction. Each phase is described in the following sections and system architecture is shown in Fig. 1.

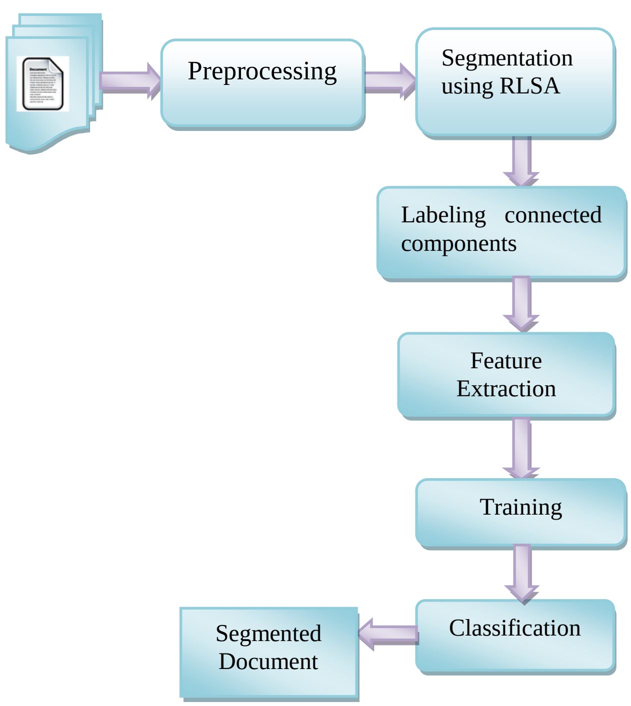


Fig. 1. System Overview

A. *Preprocessing*

Preprocessing is a sequence of tasks performed on the document image. It enhances the quality of the image for segmentation. The various tasks performed on the image in preprocessing stage are scanning, binarization, and noise removal.

1) *Scanning:*

The documents are collected from various journals and scanned at 200 dots per inch (dpi).

2) *Binarization:*

It is a process which converts the grayscale image into a binary image using the global threshold method. A binary image has only two values 0 or 1for each pixel. 0 represents white pixel and 1 represents black.

3) *Noise removal:*

From the binarized image, the noise is removed using ostu method. The obtained noiseless image is subjected to RLSA algorithm for further processing.

B. *Segmentation using RLSA algorithm*

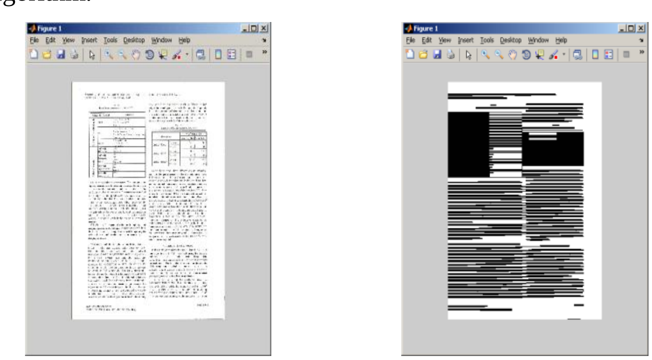
The RLSA algorithm is used for segmenting the document. The document is subdivided into blocks, where each block contains only one type of data such as text, graphic, halftone image, etc.,.

The Run length algorithm is employed to a binary image where 0 represents white pixels and 1 represents black pixels. The algorithm transforms a binary image **x** into an output image **y** as follows:

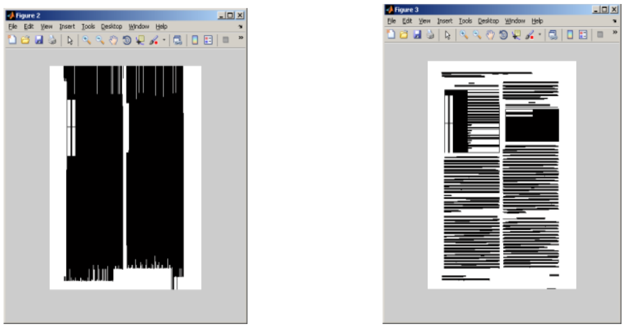
1. White pixels in x are replaced with black pixels in y if white runs are less than or equal to a predefined threshold.

2. Black pixels in x are untouched in y.

With a selection of optimal threshold value, the connected areas will form blocks of the same region. The Run Length Algorithm is employed horizontally with horizontal threshold hTh = 300 as well as vertically with vertical threshold vTh = 280 to a document image, producing two individual images. The two images are then combined using logical AND operation. Furthermore, horizontal smearing is applied with horizontal threshold Th = 30 to generate segmented blocks. Fig. 2 shows the segmentation of document image using RLSA algorithm.



(a) (b)



(c) (d)

Fig. 2. (A) Original Document Image. (B) Result Of Horizontal Smearing. (C) Result Of Vertical Smearing. (D) Final Result Of Segmentation.

C. Labeling connected components

Labeling is the process of identifying the connected components in an image and assigning each component a unique label an integer number which must be same as connected black runs. Labels are used to differentiate each block. Fig. 3 shows the labeled connected components of document image.

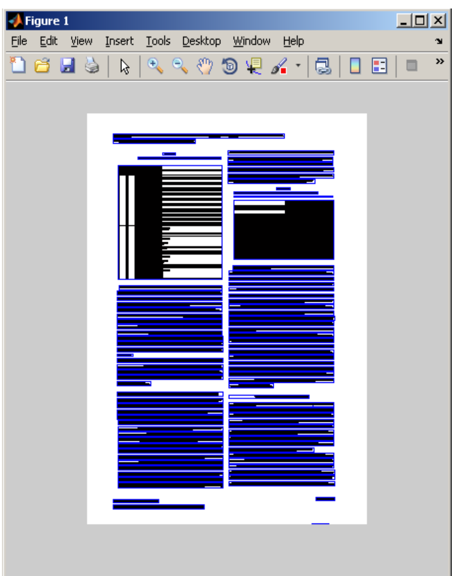


Fig. 3. Labeling Connected Components

D. Feature Extraction

Feature extraction is the process of transforming the document image block into set of features. The features describing the properties of each block are extracted based on its edges. The coordinates of the edges, specify the size of the block. Each extreme edges of block are defined by considering the origin of binary image.

Features such as height, area, aspect ratio, perimeter, perimeter/height ratio, average horizontal length are extracted to differentiate the non-text block from text block. Image, drawings and table blocks are identified by means of following features: density, density of segmented block, horizontal transition along x-axis and y-axis, vertical transition along x-axis and y-axis. The table and drawings blocks can be recognized more accurately using the features such as mean standard deviation, active pixels, energy and entropy.

Blocks are represented by (Xmin, Ymin) and (Xmax, Ymax).Where,

Ymin - Left-most pixel value of column.

Ymax - Right-most pixel value of column.

Xmin - Left-most pixel value of row.

Xmax - Right-most pixel value of row.

The following block features are computed for classification of document region.

Height (H) - Height of the block is computed by subtracting the leftmost pixel from rightmost pixel of column.

$$H=Dx = (Ymax-Ymin) +1$$

Width (W) - Width of the block is computed by subtracting the leftmost pixel from rightmost pixel of row.

$$W=Dy = (Xmax-Xmin) +1$$

Aspect Ratio (E) - It is defined as the ratio of a block's width-to-height.

$$E=W/H$$

Area (A) - Area of the block is obtained by multiplying the height and width.

$$A=H*W$$

Density (D) - Density is defined as the ratio of total number of black pixels within each block of document image to the area.

$$D=N/Area$$

Horizontal transition along x axis (HTx) - It is defined as ratio of horizontal transitions per unit height and computed as

$$HTx=HT/Dx$$

Where, HT is the horizontal transitions of black to white or white to black pixels in a block of document image.

Vertical transition along x axis (VTx) - It is defined as ratio of vertical transition per unit height and is computed using the following formula.

$$VTx=VT/Dx$$

Where, VT is the vertical transitions of black to white or white to black pixels in a block of document images.

Horizontal transition along y axis (HTy) - It is defined as horizontal transition of white to black pixels per unit width and is given by

$$HTy=HT/Dy$$

Vertical transition along y axis (VTy) - It is defined as vertical transition of white to black pixels per unit width and is given by

$$VTy=VT/Dy$$

Density of segmented block (D1) - It is the ratio of total number of black pixels of segmented block after applying Run length smearing algorithm to the area.

D1=C/Area

Average Horizontal Length(R) - It is defined as mean horizontal length of black runs of document image within each block.

R =N/HT

The Product of block height and ratio R is given by RH=R*H

The Product of ratio R and ratio E is given by

RE=R*E

The Product of ratio D1 and ratio R is given by

RD=R*D1

Mean (E) - Mean value of pixel intensities in the block is defined as arithmetic average of distribution of pixels in the block. It is computed using the function mean () in mat lab.

$$E_i^{n-1} = \sum x_i / N$$

Standard Deviation (σ) - Standard deviation of pixel intensities in the block is defined as calculated using the function std () in mat lab.

$$\sigma_i^{n-1} = \sum (x - \mu)$$

Active pixels (A) - Active pixels of the block is defined as the count of pixels with intensity < mean-k*std and it is computed as follows.

A = Sum (I< (mean-k*std))

Perimeter (P) - Perimeter is defined as the distance around the block and computed using the formula

P = 2*(H+W)

Ratio S - It is defined as ratio of perimeter to height of the block and is given by

S=Perimeter/height

Energy (E) - It is defined as sum of squared elements in the image and is also known as uniformity or the angular second moment.

$$E_{i=1}^{n} = \sum x_i^2$$

Entropy- It is defined as statistical measure of randomness that can be used to characterize the texture of the input image. It is calculated using the function entropy () in mat lab.

Entropy = $\sum_{i,j} P(i, j) \log P(i, j)$

Thus a group of 20 distinct features are extracted from the blocks of document image using mat lab. The feature vectors are generated using 519 segmented blocks of 15 document pages and the training dataset is generated with 519 instances.

III. Linear Genetic Programming-Genetic Programming Model

In artificial intelligence, genetic programming (GP) is an evolutionary algorithm based methodology impressed by biological evolution to seek out computer programs that carry out a user-defined task. It is an area of genetic algorithms (GA) where each entity is a computer code. It is a machine learning technique used to optimize a population of computer programs according to a fitness landscape determined by a program's ability to perform a given computational task[14].

A. GP algorithm

GP algorithm involves few essential control parameters for solving the problem. The control parameters needed are population size, maximum number of generations, solution fitness and operators. The algorithm begins by initializing the desired population size and evaluates fitness until best solution found. Crossover, reproduction and mutation are the common operators used in GP.

B. GP operators

Crossover operator works by choosing two parents from the population. Two random sub trees are selected from every parent and swapped to make children. In mutation single node in parent tree is chosen and replaced with a random node of same sort. E.g. a function node is replaced by a function node of same sort and a terminal node is replaced by a randomly selected terminal node. In reproduction operator an individual is chosen and copied on to the new generation with none changes or modifications to that.

C. Classification Using Genetic Programming

The GP based classification methods are categorized into three types. First kind of method illustrates the evolution of classification algorithms like decision trees, neural networks or other rule based algorithms. This system accurately depicts the use of GP for program evolution. The second system contains evolution of classification rules or expressions. Rules are evolved in the type of logical expressions with logical operators. In the third system the expressions are evolved in the type of arithmetic expressions or functions [15].

D. Fitness function for classification

One among the most common fitness function for classification task is the classification accuracy. The accuracy tells the amount of instances properly classified by a classifier. Another measure is to minimize the classification error that is reciprocal of classification accuracy. Fitness measure for GP classification task is uncomplicated. It is determined by the classification accuracy that is expressed as the following formula.

$$fitness = \frac{Correctly\ classified\ instances}{Total\ no\ of\ instances} \times 100$$

E. Linear Genetic Programming

Linear genetic programming is one type of representation of GP. In Linear genetic programming LGP each program is a sequence of register instructions and usually expressed in human-readable form. The registers are assigned zero before program execution. The features representing the objects to be classified are loaded into registers. The program is executed in an imperative manner and represents a directed acyclic graph. Fig 4 shows the program generated by linear genetic programming based discipulus. Discipulus has a feature called self tuning and self parameterizing which selects the GP control parameters based on its problem to be solved.

```
  if (cflag) f[0]  = f[0];
  f[0]  = - f[0];
  tmp  = f[0];  f[0]  = f[0]; f[0] =  tmp;
  f[0] *  =  Input8;
  tmp  =  f[1]; f[1]  =  f[0];  f[0]  =  tmp;
  cflag = (Double.isNaN(f[0]) || Double.isNaN(f[0])) ? true :
(f[0] < f[0]);
  tmp = f[0]; f[0] = f[0]; f[0] = tmp;
   f[0]* = Math.pow(2,trunc(f[1]));
  cflag = (Double.isNaN(f[0]) || Double.isNaN(f[1])) ? true :
(f[0] < f[1]);
  f[0] = Math.cos(f[0]);
  f[0] = Math.sin(f[0]);
  f[0] = Math.abs(f[0]);
   f[0]+ = 1.366016626358032f;
   f[0]* = -1.924433708190918f;
  f[0] = Math.abs(f[0]);
   f[0]+ = f[0];
  f[0]* = -1.924433708190918f;
  f[0]- = Input17;
  f[1]- = f[0];
  cflag = (Double.isNaN(f[0]) || Double.isNaN(f[0])) ? true :
(f[0] < f[0]);
  f[0]+ = Input1;
  f[0]- = -0.494312047958374f;
   f[0]* = -1.924433708190918f;
  f[0]/ = Input17;
  f[0]/ = -0.494312047958374f;
   f[0] = Math.abs(f[0]);
  f[1]* = f[0];
  f[0]/ = -0.494312047958374f;
   f[0]/ = Input3;
  f[0]/ = Input1;
  f[0]/ = Input2;
  f[0]* = 1.252994060516357f;
  f[0]- = -0.494312047958374f;
  cflag = (Double.isNaN(f[0]) || Double.isNaN(f[0])) ? true :
(f[0] < f[0]);
  tmp = f[0]; f[0] = f[0]; f[0] = tmp;
  cflag = (Double.isNaN(f[0]) || Double.isNaN(f[0])) ? true :
(f[0] < f[0]);
  tmp = f[0]; f[0] = f[0]; f[0] = tmp;
  cflag = (Double.isNaN(f[0]) || Double.isNaN(f[0])) ? true :
(f[0] < f[0]);
  f[0]* = f[0];
  return (float) f[0];
}
```

Fig. 1. Program Generated By LGP Using Discipulus Tool

Where input[i] represents the input variables, f[i] represents binary class values used in LGP. The function set of the system consists of arithmetic operators (=,,-,/,*), conditional operator (if condition) and trigonometric measures (sin, cos, tan, log, sqrt etc). The output of the program is decided by considering the highest value hold by f[i] register.

IV. EXPERIMENT AND RESULTS

The document image classification model is generated by implementing supervised learning algorithm. The documents used for creating the dataset are collected from various journals and here 15 document pages have been used. The dataset consists of 519 segmented blocks of which 196 are text blocks, 123 are drawing blocks, 92 are image blocks and 108 are tabular blocks. The features describing the properties of the blocks are extracted and the size of each feature vector is 20. The dataset with 519 instances is trained by genetic programming using discipulus tool.

Discipulus is the world's first and fastest commercial Genetic programming and data analysis software. Discipulus writes computer programs automatically in Java, C, C Sharp, Delphi and Intel assembler code, all on a desktop computer. Discipulus handles only binary classification. But, the problem of document segmentation is a multiclass classification. Hence, it has been solved by extending binary classification into multiclass classification using one against one method.

A. One Against One

In this case, for 4 classes (text, image, drawings and table) 6 multiclassifiers have been built, one for each pair of classes. The prediction of class for a new data point is based on voting scheme. Fig 5 shows the architecture of the prediction scheme for the four classes.

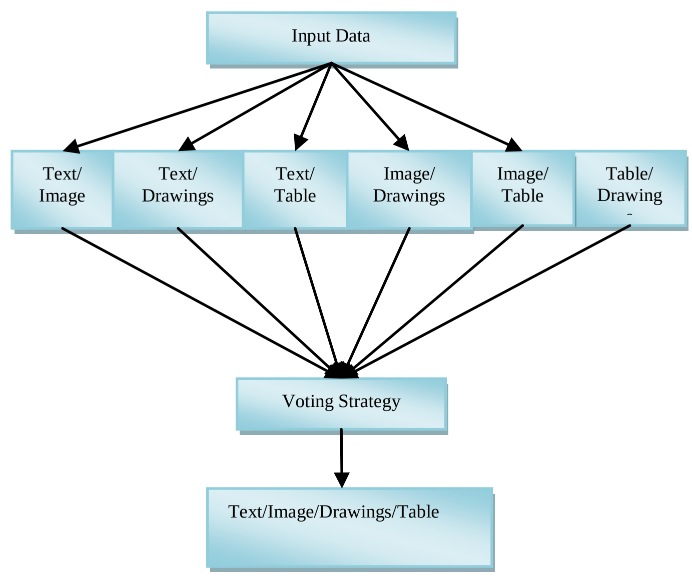


Fig. 5. Prediction Using One Against One Multiclassification

B. Training and testing in Discipulus

Discipulus is bundled with Notitia, which is data preparation, cleaning, and import software. Notitia lets you import, clean-up, transform and split data for use in Discipulus. Fig 6 and Fig 7 shows data import, data preparation using Notitia and data export to Discipulus respectively. For each binary model, hits then fitness approach is used as fitness function for classification.

In hits then fitness approach, different weights (or costs) are assigned to positive and negative hits to obtain hit rate. Positive hit is the hit rate assigned to positive instance (1). Negative hit rate is the hit rate assigned to negative instances (0).

The hit-rate is defined as the percentage of correctly classified instances. Threshold value is used to classify the outputs of binary models. If program's output for an instance is greater than or equal to the classification threshold, that output is counted as class one output. Else if the program's output for an instance is less than the classification threshold, that output is counted as class zero output. Threshold value is set automatically halfway between the target outputs for classes zero and one.

The dataset is used for learning six binary classifiers are built as per one against one multiclassification scheme. The java codes corresponding to these six classifiers or classification models have been saved using Discipulus. These six models have been invoked in a java program and voting strategy is implemented to test the class label of the given input vector. In this manner a test dataset with 40 instances, 10 in each class (text, image, drawings, and table) is tested and classification accuracy is calculated. It is observed that about 97.5% of classification accuracy is shown by genetic programming based document segmentation and region classification. Fig 8 to 13 depict the training and testing of GP based multiclassification in Discipulus

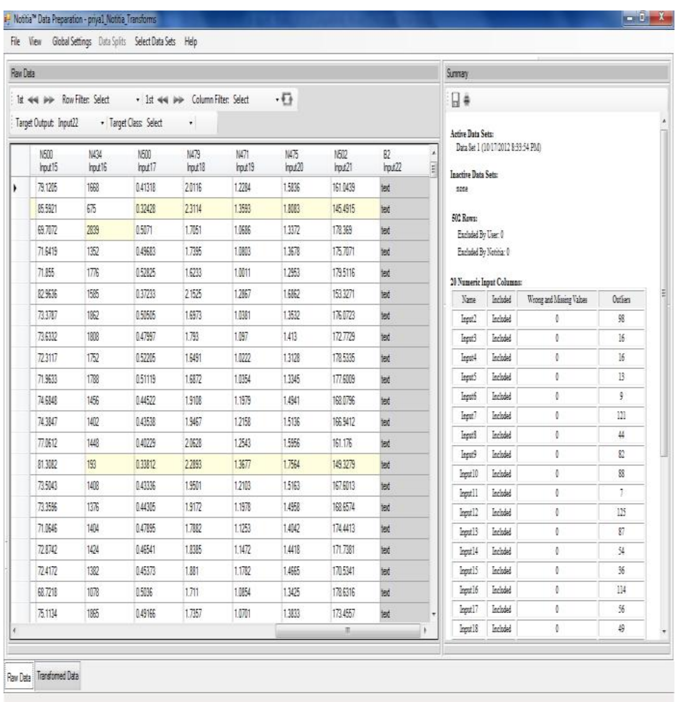


Fig. 6. Data Import And Preparation Using Notitia

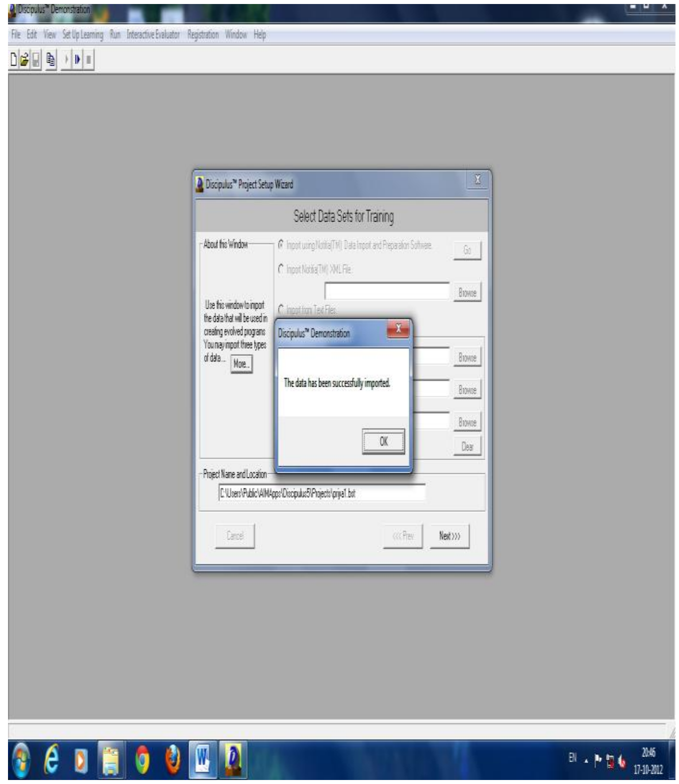


Fig. 7. Data Export Using Notitia To Discipulus

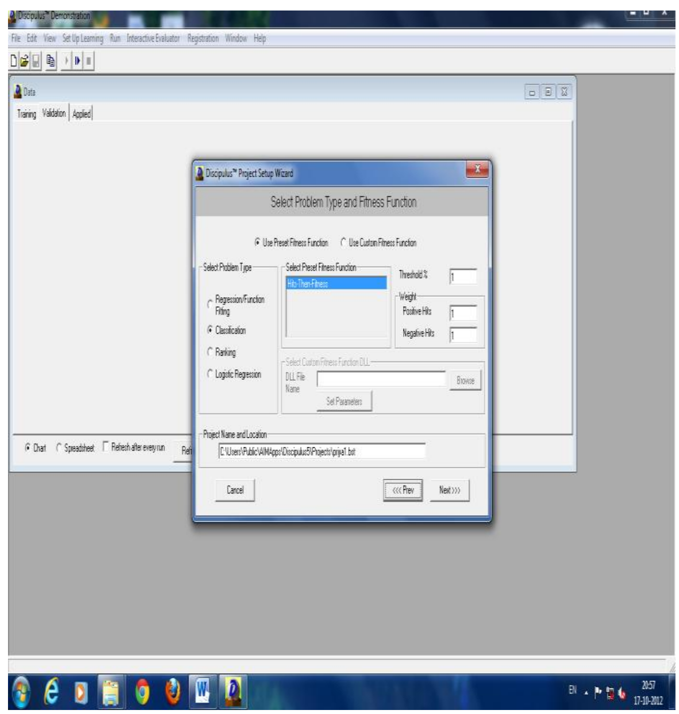


Fig. 8. Classification On Discipulus

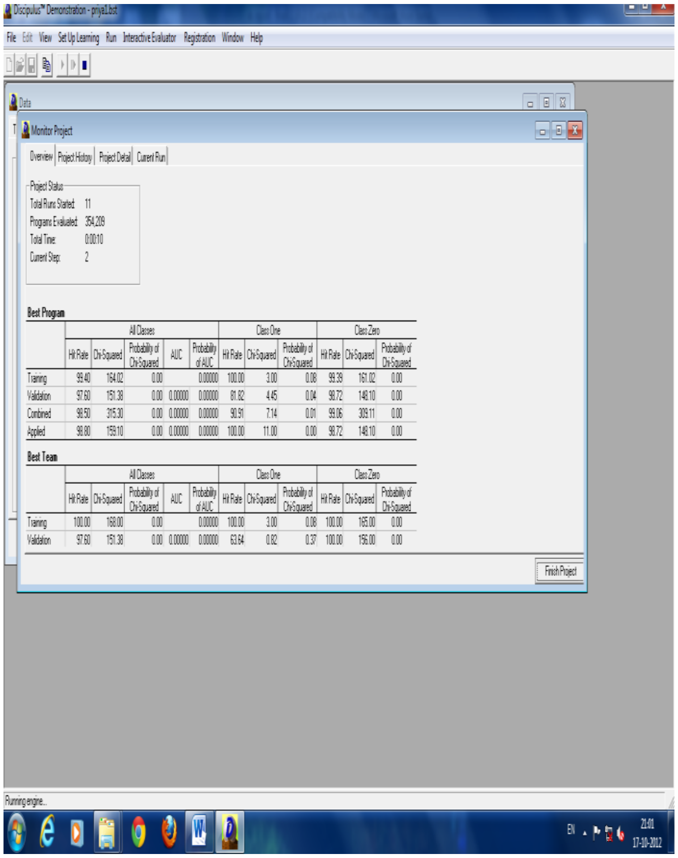


Fig. 9. Project Status On Discipulus

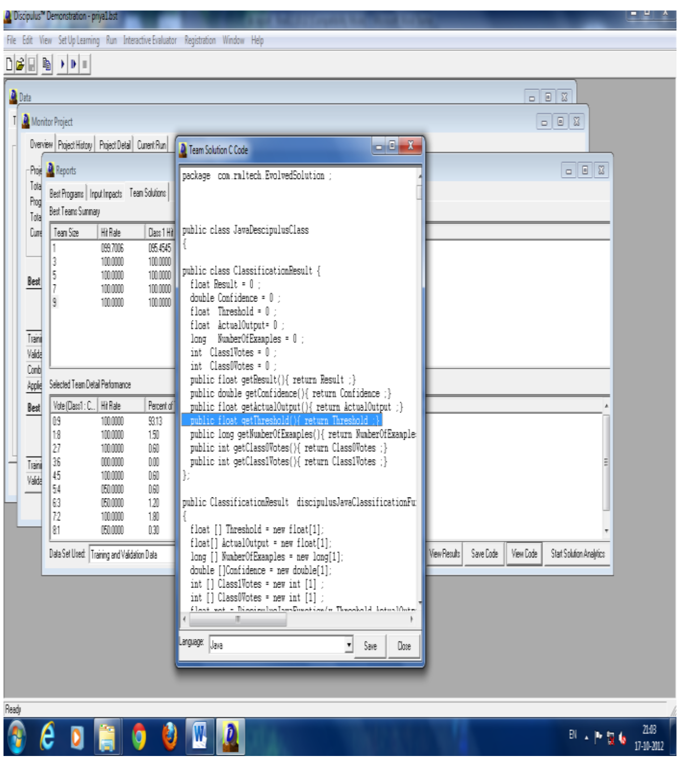


Fig. 10. Java code for classification model created using Discipulus

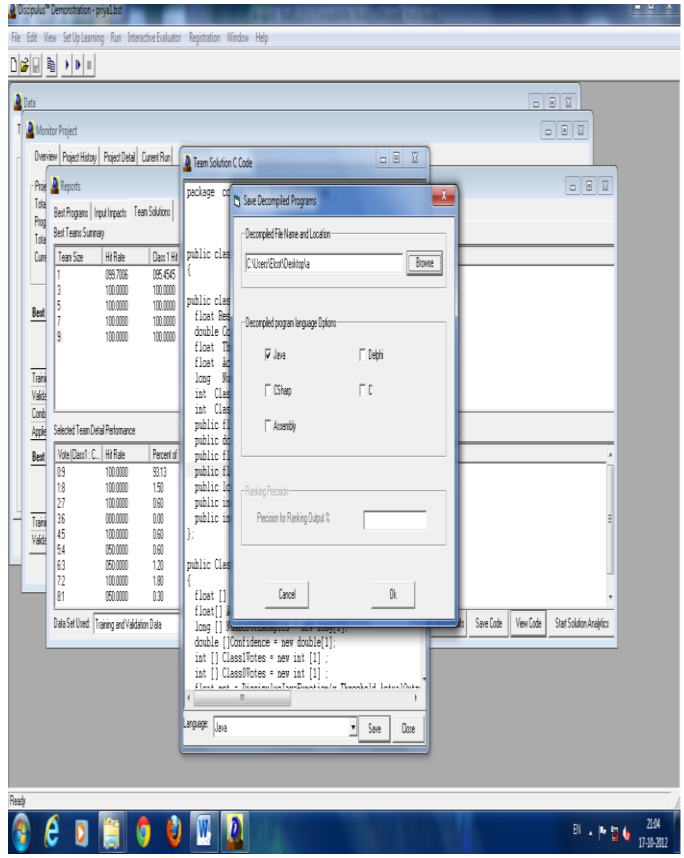


Fig. 11. Saving Of Java Code

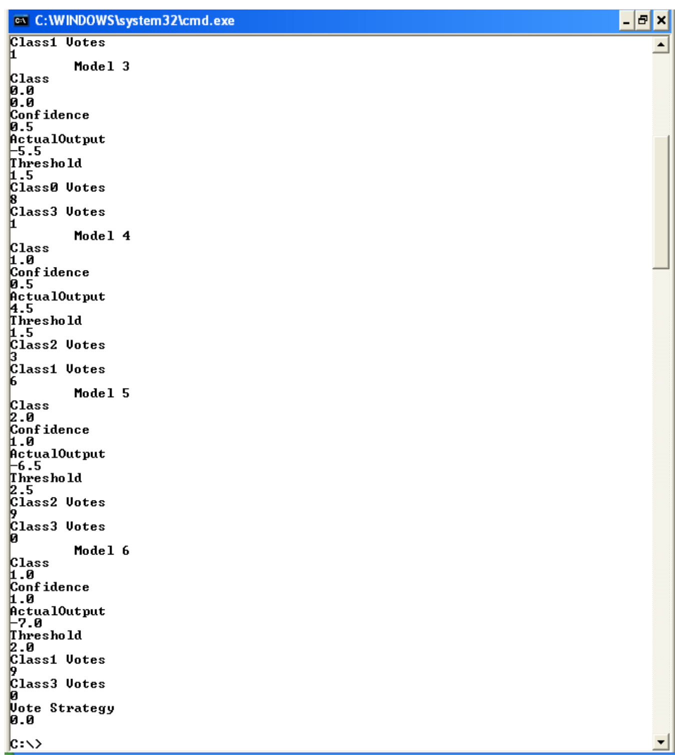


Fig. 12. Result Of Testing New Dataset In Java

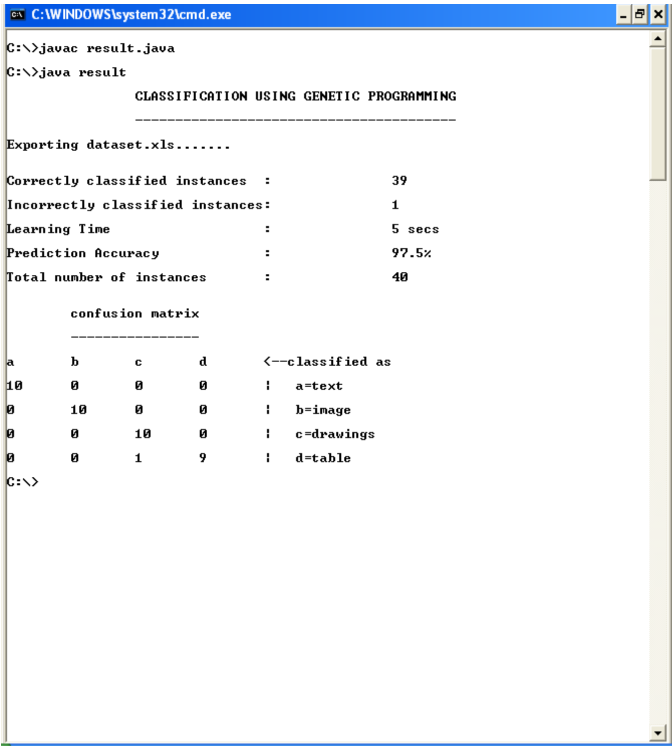


Fig. 13. Cross Validation Of GP

V. CONCLUSION

This paper demonstrates the modeling of document segmentation as classification task and describes the implementation of genetic programming approach for classifying various regions. Discipulus have been applied for generating LGP based classification models. It is observed that classification implemented by genetic programming in this paper is more efficient than other machine learning algorithms because the commercial GP software Discipulus uses automatic induction of binary machine code to achieve better performance.

Document segmentation model constructed in this research work can be incorporated into "Form" processing system to extract the data which are presented in text, image, table, drawing format from any kind of "Forms". Examples of "Forms" may include a student application form, bank application form.

REFERENCES

[1] Rangachar Kasturi, Lawrence O'gorman and Venu govindaraju. "Document Image Analysis: A Primer".

[2] Okun O, Doermann D and M. Pietikainen. "Page segmentation and zone classification". The state of the art. In UMD, 1999.

[3] Jain A and Yu B. "Document representation and its application to page decomposition". IEEE trans. On Pattern Analysis and Machine Intelligence, 20 (3): 294–308, March 1998.

[4] Frank Y shih, shy-shyan chen, Douglas Hung DC and Peter A Ng. "A Document segmentation, classification and recognition system"

[5] Wahl F, Wong K, and Casey R. "Block segmentation and text extraction in mixed text/image documents". CGIP, 20:375–390, 1982.

[6] Wang D and Srihari S. "Classification of newspaper image blocks using texture analysis". CVGIP, 47:327–352, 1989.

[7] Pavlidis T and Zhou J. "Page segmentation by white streams". Proceedings 1st International Conference Document Analysis and Recognition (ICDAR), International journal of Pattern Recognition, pages 945– 953, 1991.

[8] Tan C and Zhang Z. "Text block segmentation using pyramid structure". SPIE Document Recognition and Retrieval, San Jose, USA, 8:297–306, January 24-25 2001.

[9] Le D X, Thomas G, Weschler H. "Classification of Binary Document Images into Texture or Non-textual Data Blocks Using Neural Network Models".

[10] Andy Song. "Texture Classification: A Genetic Programming Approach".

[11] Jain AK and Bhattacharjee S. "Text segmentation using Gabor filters for automatic document processing". Machine Vision and Applications, 5 (3): 169–184, 1992.

[12] O'Gorman, Kasturi L. "Document Image Analysis". IEEE Computer Society Press, Los Alamitos, California, 1995.

[13] Ha T, Bunke H. "Image processing methods for document image analysis". Handbook of character recognition and document image analysis. World Scientific 1–47, (1997).

[14] http://en.wikipedia.org/wiki/Genetic_programming.

[15] Hajira Jabeenand Abdul Rauf Baig. (2010) "*Review of Classification Using Genetic programming*". International Journal of Engineering Science and Technology .Vol 2, no 2, pages 94-103.

[16] Kise k, Sato A and Iwata M. "Segmentation of page Images using the area Voronoi Diagram". Computer Vision and Image Understanding 70: 370-382, 1998.

AUTHORS PROFILE

N. Priya dharshini is pursing Master of Philosophy in Computer Science in PSGR Krishnammal college for women under the guidance of MS.Vijaya. Her research interests are data mining, image processing, and pattern recognition.

MS. Vijaya is presently working as Associate Professor in GR Govindarajulu School Of Applied Computer Technology, PSGR Krishnammal college for women, Coimbatore, India. She has 22 years of teaching experience and 8 years of research experience. She has completed her doctoral programme in the area of Natural Language Processing. Her areas of interest include Data Mining, Support Vector Machine, Machine learning, Pattern Recognition, Natural Language Processing and Optimization Techniques. She has presented 22 papers in National conferences and she has to her credit 17 publications in International conference proceedings and Journals.

She is a member of Computer Society of India, International Association of Engineers (Hong Kong), International Association of Computer Science and Information Technology (IACSIT – Singapore). She is also a reviewer of International Journal of Computer Science and Information Security.

Moving Domestic Robotics Control Method Based on Creating and Sharing Maps with Shortest Path Findings and Obstacle Avoidance

Utilization of Place Indentifier: PI

Kohei Arai [1]
Graduate School of Science and Engineering
Saga University
Saga City, Japan

***Abstract*—Control method for moving robotics in closed areas based on creation and sharing maps through shortest path findings and obstacle avoidance is proposed. Through simulation study, a validity of the proposed method is confirmed. Furthermore, the effect of map sharing among robotics is also confirmed together with obstacle avoidance with cameras and ultrasonic sensors.**

Keywords-domestic robotics; obstacle avoidance; place identifierl ultrasonic sensor; web camera

I. Introduction

Domestic robotics utilizing services are available in hospitals, group homes, private homes, etc. Domestic robot has camera image acquisition and voice output capability. Patients in hospitals wear a single Head Mount Display: HMD and an eye looking camera mounted glass. Camera acquired their eye image is transmitted to mobile phone through Bluetooth communication interface. Many robot helpers have been investigated by researchers. The ARM9-based Car controlled remotely has been developed by Wang Shaokun et all [1]. The embedded Linux system was installed under ARM9-structure processor for real time robot operation. It also optimized and improved the versatility and rapid data transmission of wireless remote car. The robots collected the data sensor and relay it to main PC station over WIFI network. Another robot also has been developed by Ding Chengjun et all [2].

Based on embedded WinCE5.0 operating system, they created remote control for mobile robot. The low power consumption and perfect real-time controller is the main goal of this. The data was sent using TCP/IP protocol over WIFI network. Another research concern in remote robot has been developed by Niu Zhigang and Wu Yanbo [3]. They developed a wireless remote control special design for Coal Mine Detection Robot. They investigated the embedded motion control system and apply it for Coal Mine Detection Robot's control system and wireless remote control.

The scenery around the robot in coal mine environment was transmitted to the main station, used it for controlling the robot movement such as forward, backward, turning left, turning right and tipping over the front arm. Ofir H et all have evaluated the telerobotic interface components for teaching robot operation [4]. They evaluated the control method of the robotic arm and the use of three alternative interface designs for robotic operation in the remote learning. Another system has been proposed by He Qingyun et all [5]. They created an embedded system of video capture and the transmission for monitoring wheelchair-bed service robots remotely. The embedded linux, S3C2410AL microprocessor, AppWeb 3.0 server, and block-matching motion estimation were taken into account for obtaining better video compression data. The remote control robot system with a hand-held controller has been proposed by Dmitry Bagayev et all [6]. The dog robot for accompanying the elderly has been proposed by Wei-Dian Lai [7]. The improved interaction technique between users and the robot has been developed for making elderly easy to use.

Application software of the mobile phone allows estimation of line of sight of their eye when they are looking at a certain location of a key in the keyboard which is displayed on the HMD screen. Thus they can create sentences and can communicate with the surrounding peoples of the robot because voice output capability is implemented on the robot. Also the patients select functions, move forward, turn right, turn left and the other functions by looking at the function keys.

At the same time, acquired image with the mounted at the tip of the robot is transmitted to the mobile phone through WiFi networks. Therefore, they can look at the outside scenery of the robot which is displayed on HMD screen. Such this domestic robot is designed and developed. Using this robot, patients enjoy virtual trip in the hospital, communication with the surrounding peoples, make orders to nurses, medical doctors, etc.

Domestic robots have to be moved safely with avoiding obstacles and have to find a shortest pass. This paper deals with the methods for obstacle avoidance, and finding a shortest pass based on Dijkstra algorithm [8]. The following section describes the proposed domestic robot followed by the methods for obstacle avoidance and finding a shortest pass. Then experiments and simulation study is described followed by conclusion with some discussions.

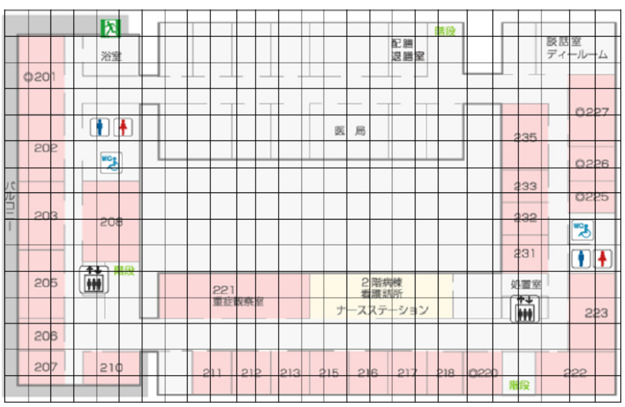
II. PROPOSED DOMESTIC ROBOT HELPING HOSPITALIZED PATIENTS

A. *Fundamental Ideas of the Proposed Robot*

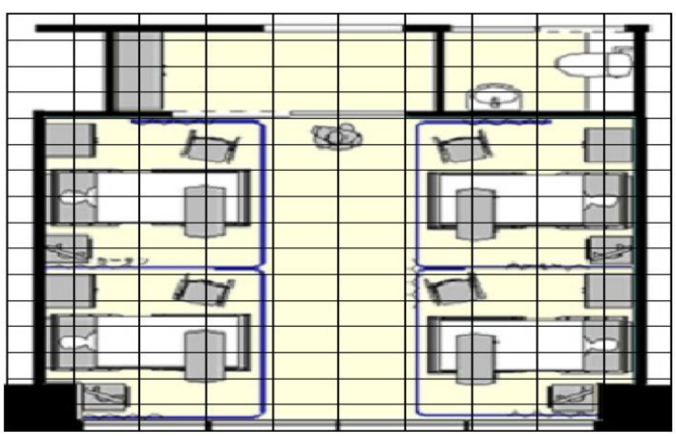
There is a concept of Place Identifier: PI. One of the examples shown below,

tag:hospital.net,2009:pi/2floor/31-2/view1/flag/hh.mm.ss

This is a kind of tag expressions which is known as tag scheme (RFC4151)[1] which allows identifying spatial reference coordinate system with DNS representations. Tag information says, the name of the hospital, floor, room, interior, the data acquisition time and so on as shown in Figure 1. Figure 1 (a) shows an example of floor layout of the floor in the hospital while Figure 1 (b) shows an example of room layout. Therefore, any specific piece of the items in the interior, in the room, on the floor in the hospital can be identified with PI. These examples have mesh. Every cross points of the mesh has the spatial reference coordinate system. Therefore, DNS representation of tag scheme is on the corresponding spatial reference coordinate system.



(a)Floor Layout



(b)Room Layout

Fig. 1. Examples Of Floor And Room Layout

B. *Process Flow of the Proposed Robot Control*

Figure 2 shows the process flow of the proposed helper robotics control.

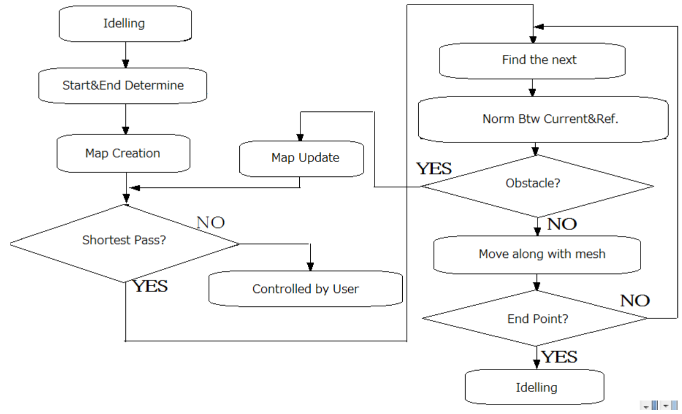
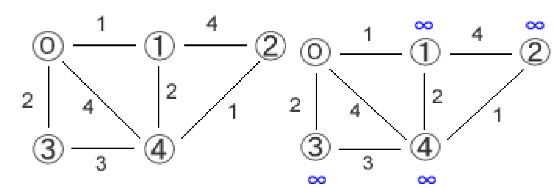


Fig. 2. Process Flow Of The Proposed Helper Robotics Control

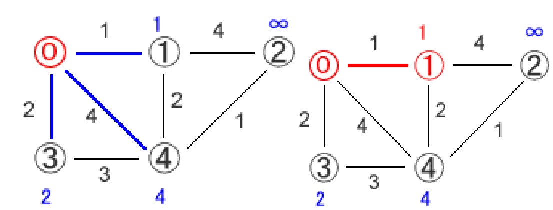
Along with the mesh which is shown in Figure 1, robotics are controlled and moved from the start point to the end point (destination). Shortest pass can be found based on Dijkstra algorithm and obstacles are detected by the norm image between the current and the reference images. All the robots on the same floor and room has shared map which is composed with the mesh data. Using floor and room layout, static obstacles are found and stored in the map database in a real time basis. Therefore, collision may not be happened.

C. *Shortest Pass Finding with Dikstra Algorithm*

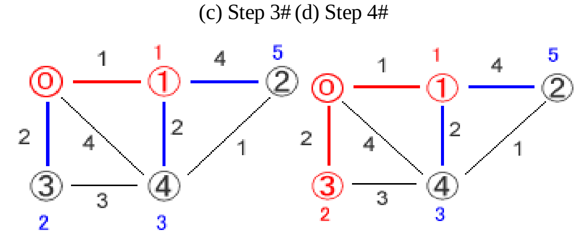
Djikstra algorithm can be illustrated as shown in Figure 3.In the Figure 3, there are just five nodes. Pass lengths are indicated between the adjacent nodes. From Figure 3 (a) to (h), the shortest pass is found subsequently. Thus the shortest pass, from node No.0 to node No.2 through node No.1 and No.4 can be found.



(a)Step #1 (b) Step #2

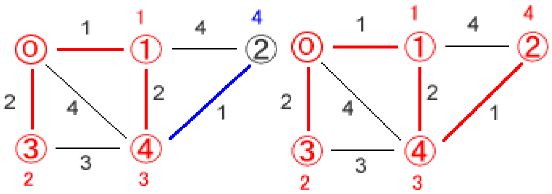


(c) Step 3# (d) Step 4#



(e) Step 5# (f) Step #6

[1] http://www.ietf.org/rfc/rfc4151.txt



(g) Step #7 (h) Step #8

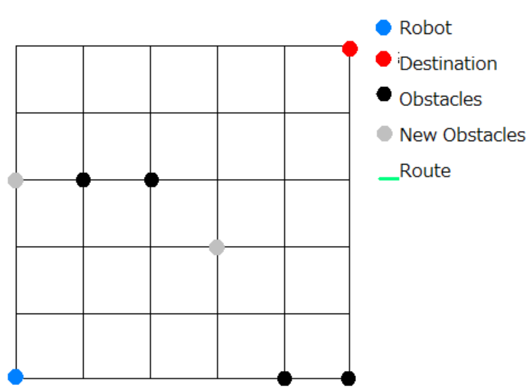
Fig. 3. Dijkstra Algorithm For Shortest Path Findings

D. *Method for Obstacle Avoidance*

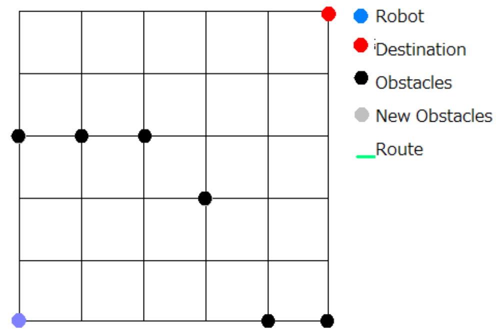
Using camera acquired image which are mounted at the tip of the domestic robot, patients can identify obstacles. One of the examples of the acquired images with camera is shown in Figure 4 together with the outlook image which is taken from the hand held camera tracing to the robot.

Fig. 4. Acquired Image with the Camera Mounted At the Tip of Helper Robot and the Image of Helper Robot from the Hand Held Camera

Patients can avoid the obstacles which are found by looking at the camera acquired image through finding a route for avoiding the obstacles. It is also possible to avoid obstacles because robots are sharing their own created map as shown in Figure 5.



(a)The first robot finds new obstacles



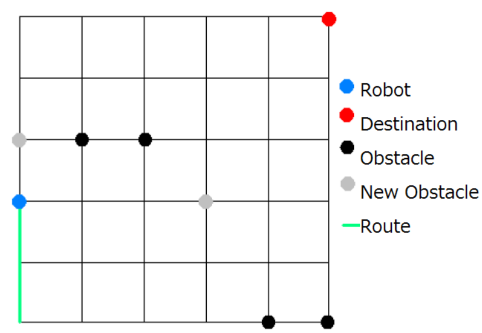
(b)The second robot may avoid the obstacles found by the first robot with the map sharing among the robots

Fig. 5. Obstacle Avoidance with Map Sharing Among Robots

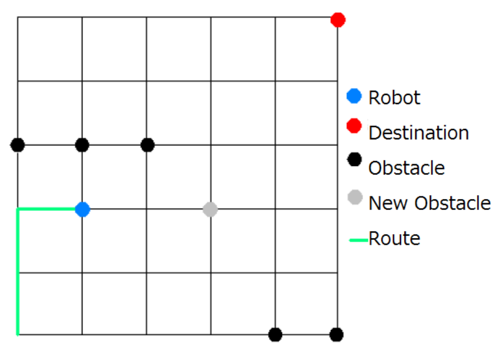
III. EXPERIEMENTS

A. *Preliminary Experiment*

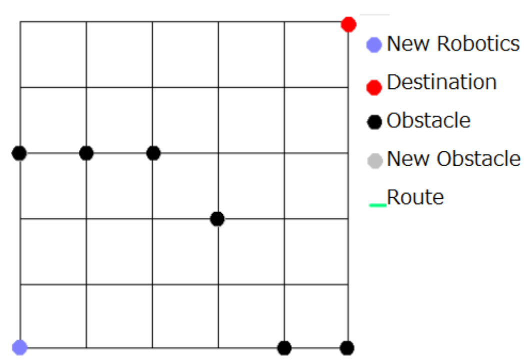
Figure 6 shows the example of process flow for obstacle avoidance.



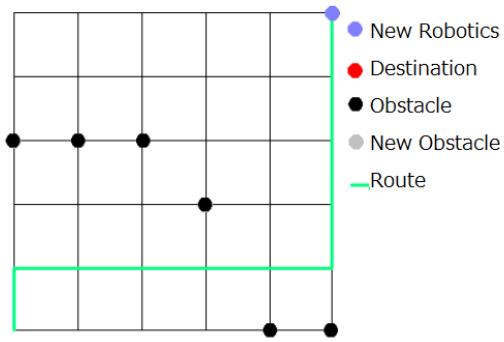
(a)Step #1



(b) Step #2



(c) Step #3



(d) Step #4

Fig. 6. Example Of Process Flow For Obstacle Avoidance.

Original situation is shown in Figure 5 (a). Starting from the bottom left corner, robot moves to the destination which is situated at the top right corner. Two robots share floor and room layout maps. At the step #1 and #2, the first robot creates a map. The map is shared by the second robot. Therefore, the second robot can avoid transient obstacles because the first robot meet the transient obstacles then the maps are updated with the location of the transient obstacles.

At the step #1, new obstacles (grey colored objects) appear. There are three types of obstacles, static, dynamic, and transient obstacles. Wall, door, interior, etc. is a kind of static obstacle while moving obstacles such as walking nurse, medical doctor, and patient is a dynamic obstacle. Suddenly appeared and disappeared obstacles, on the other hands, are transient obstacles. In the step #1, transient obstacles appear suddenly. Therefore, the robot has to change the route for avoiding static and transient obstacles in accordance with the Dijkstra algorithm.

At the step #2, the first robot meet the another obstacle. Then the robot changes route in accordance with Dijkstra algorithm. At the step #3, the second robot departs from the start point for the same destination of the top right corner in accordance with the updated maps. Therefore, the second robot can determine the shortest pass as shown in Figure 6 (d) of the step #4.

B. *Obstacle Finding and Avoidance*

Obstacles can be found by comparing between current and the reference image without obstacle. At the cross points of location, of meshed data of floor and room layout, forward looking images are acquired a prior basis as reference images. Therefore, obstacle can be identified when the norm between the current and the reference images is greater than zero. Reducing influences due to illumination condition changes, shadows and shades by using Near Infrared: NIR camera, the current and the reference images are acquired. Also influence due to geometric distortion on the calculation of norm can be eliminated with image matching (or template matching) between the current and the reference images. In the image matching process, Affine transformation is assumed for geometric distortion.

Figure 7 (a) and (b) shows the reference and the current images which are acquired with the forward looking NIR camera mounted at the tip of the robot. There is an obstacle in the current image. Also Figure 8 shows norm image between Figure 7 (a) and (b). Obstacle can be detected as shown in Figure 8. Figure 8 shows the norm image between the current and the reference images at the cross point of the meshed data for floor and room layout.

(a)Reference image

(b)Current image

Fig. 7. Examples of the Reference and the Current Images

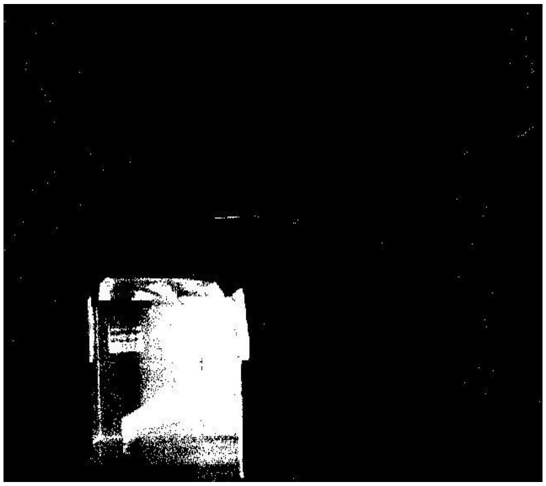


Fig. 8. Norm Image between the Reference and the Current Images Results in Obstacle Finding.

C. *Realistic Simulation*

Much realistic situation is taken into account with the following simulation parameters,

Mesh size: 30 by 20

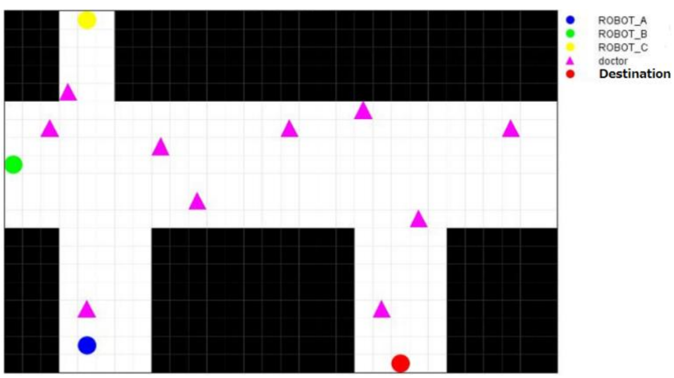
The number of robotics: 3

The number of moving obstacles: 0 to 15

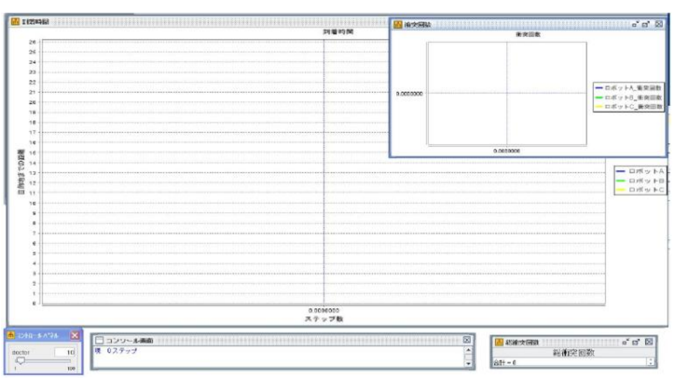
Floor and room layout, initial location of robotics, moving obstacles (medical doctors, nurses, patients) start and end points (destination) is shown in Figure 9. The simulations are taken place with a variety of initial conditions of the initial location of robotics and moving obstacles. Floor and room layout as well as mesh size and the number of robotics are fixed. The initial locations are determined with uniformly distributed random number. The number of trials is set at 10 times.

Figure 9 (a) and (b) shows situation and location of robotics as well as log data of three robotics locations at the initial situation while Figure 9 (c) and (d) shows those at the intermediate situation. Meanwhile, Figure 9 (e) and (f) shows those for the final situation.

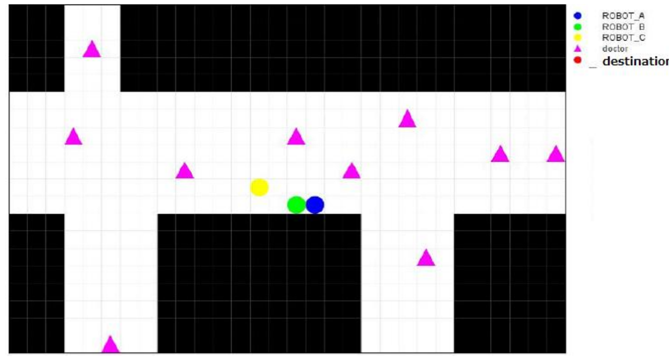
When collision occurs between or among the robotics, then it is assumed to be delayed 10 unit steps for recovering from the collision situation. A comparative study on the required number of steps (time) for getting the destination from the start point is conducted between with and without map sharing among robotics.



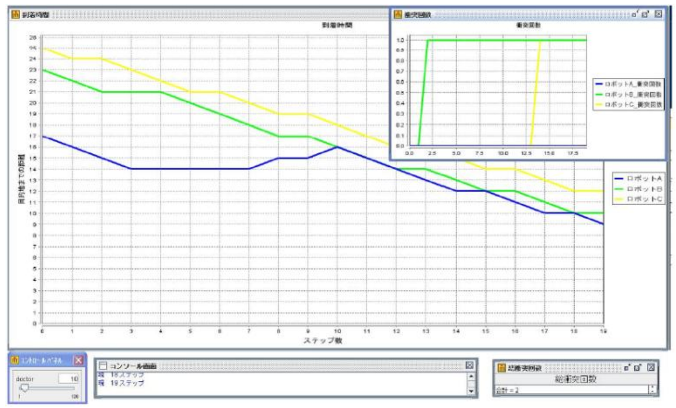
(a)Initial situation of robotics and moving obstacles



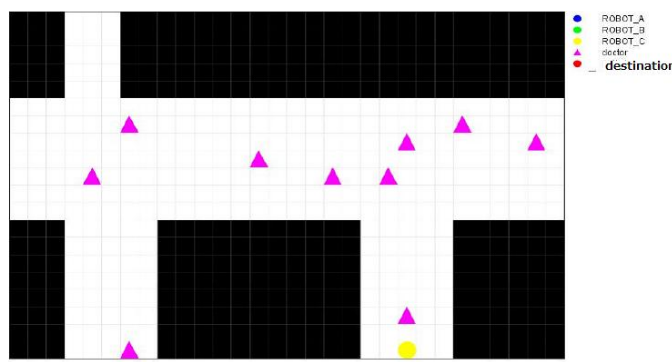
(b)Log data of the location of three robotics as function of the number of steps



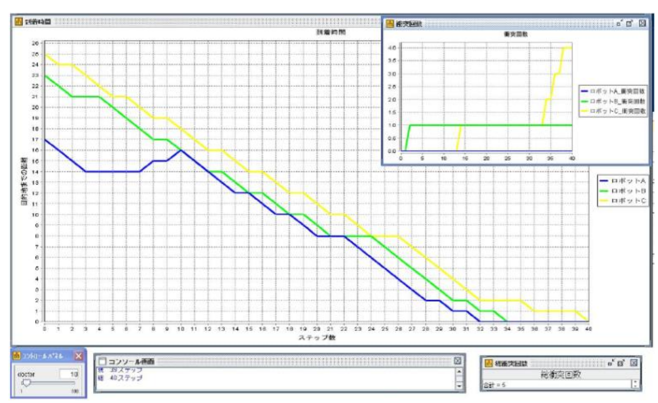
(c)Intermediate situation



(d)Log data of three robotics location at intermediate situation



(e)Final situation



(f)Log data of three robotics locations at the final situation

Fig. 9. Situation and Location of Robotics at the Initial, Intermediate, and Final Situations

Simulation results show that the number of collision increased with increasing of the number of moving obstacles as shown in Figure 10. Also, a relation between the number of moving obstacles and the number of unit time steps for getting the destination from the start point is shown in Figure 11. It is obvious that the number of unit time steps required for getting the destination is increased in accordance with increasing the number of moving obstacles because the number of collisions increased with increasing of the number of moving obstacles.

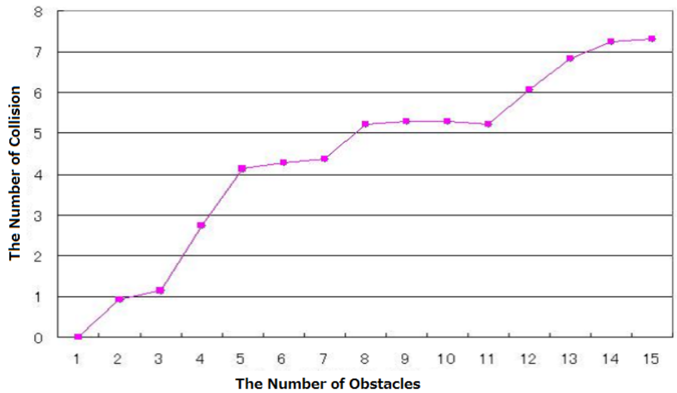


Fig. 10. Relation Between The Number Of Collision And The Number Of Moving Obstacles

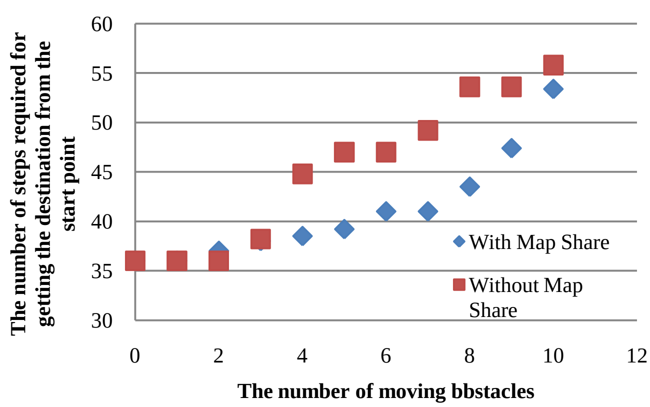


Fig. 11. Relation Between The Number Of Moving Obstacles And The Number Of Steps Required For Getting The Destination From The Start Point.

As shown in Figure 11, the effect of map sharing for reducing the time required for reaching the destination from the start point is quite obvious. In this simulation study, the time required for reaching the destination from the start point is almost same at the number of moving obstacle of 10. Because that the routes is getting complex in accordance with the number of moving obstacles, the effect of the map sharing is saturated is the number of moving obstacles is greater than 10 in this case.

IV. Conclusion

Control method for moving robotics in closed areas based on creation and sharing maps through shortest path findings and obstacle avoidance is proposed. Through simulation study, a validity of the proposed method is confirmed. Furthermore, the effect of map sharing among robotics is also confirmed together with obstacle avoidance with cameras and ultrasonic sensors.

For the small size simulation cell such as 30 by 20, the effect of the map sharing for reducing the time required for reaching the destination from the start point is significant and is saturated in accordance with the number of moving obstacles.

Acknowledgment

The author would like to thank Mr. Kazuyuki Fukuda for his effort to conduct simulation studies.

References

[1] Wang Shaokun, Xiao Xiao, and Zhao Hongwei, "The Wireless Remote Control Car System Based on ARM9," in Instrumentation, Measurement, Computer, Communication and Control, International Conference on , October 2011 , pp. 887-890.

[2] Ding Chengjun, Yan Bingsen, and Duan Ping, "The Remote Control of Mobile Robot Based on Embedded Technology," in Measuring Technology and Mechatronics Automation, International Conference on , January 2011, pp. 907-910.

[3] Niu Zhigang and Wu Yanbo, "Research on Wireless Remote Control for Coal Mine Detection Robot," in Digital Manufacturing and Automation, International Conference on , December 2010 , pp. 315-318.

[4] Ofir H Goldstain, Irad Ben-Gal, and Yossi Bukchin, "Evaluation of Telerobotic Interface Components for Teaching Robot Operation," in IEEE TRANSACTIONS ON LEARNING TECHNOLOGIES, 2011, pp. 365-376.

[5] Niu Zhigang and Wu Yanbo, "Research on Wireless Remote Control for Coal Mine Detection Robot," in Digital Manufacturing and Automation, International Conference on, December 2010, pp. 315-318.

[6] Evsyakov Artem Dmitry Bagayev, "System remote control of the robotized complex - Pegas," in East-West Design & Test Symposium , September 2010 , pp. 358-361.

[7] Wei-Dian Lai, "Zigbee remote control interface for a robot dog design for older users," in Proceedings of the 5th International Confernece on Ubiquitous Information Management and Communication, February 2011 , pp.1-5.

[8] Dijkstra, E.W. (1959). *A note on two problems in connexion with graphs*. In *Numerische Mathematik*, 1 (1959), S. 269 ～ 271.

Authors Profile

Kohei Arai, He received BS, MS and PhD degrees in 1972, 1974 and 1982, respectively. He was with The Institute for Industrial Science, and Technology of the University of Tokyo from 1974 to 1978 also was with National Space Development Agency of Japan (current JAXA) from 1979 to 1990. During from 1985 to 1987, he was with Canada Centre for Remote Sensing as a Post Doctoral Fellow of National Science and Engineering Research Council of Canada. He was appointed professor at Department of Information Science, Saga University in 1990. He was appointed councilor for the Aeronautics and Space related to the Technology Committee of the Ministry of Science and Technology during from 1998 to 2000. He was also appointed councilor of Saga University from 2002 and 2003 followed by an executive councilor of the Remote Sensing Society of Japan for 2003 to 2005. He is an adjunct professor of University of Arizona, USA since 1998. He also was appointed vice chairman of the Commission "A" of ICSU/COSPAR in 2008. He wrote 30 books and published 332 journal papers.

Eye-Base Domestic Robot Allowing Patient to be Self-Services and Communications Remotely

Helping Patients to Make Order, Virtual Travel, and Communication Aid

Kohei Arai [1]
Graduate School of Science and Engineering
Saga University
Saga City, Japan

Ronny Mardiyanto [2]
Department of Electronics Engineering
Institute of Technology, Surabaya
Surabaya City, Indonesia

***Abstract*—Eye-based domestic helper is proposed for helping patient self-sufficient in hospital circumstance. This kind of system will benefit for those patient who cannot move around, it especially happen to stroke patient who in the daily they just lay on the bed. They could not move around due to the body malfunction. The only information that still could be retrieved from user is eyes. In this research, we develop a new system in the form of domestic robot helper controlled by eye which allows patient self-service and speaks remotely. First, we estimate user sight by placing camera mounted on user glasses. Once eye image is captured, the several image processing are used to estimate the sight. Eye image is cropped from the source for simplifying the area. We detect the centre of eye by seeking the location of pupil. The pupil and other eye component could be easily distinguished based on the color. Because pupil has darker color than others, we just apply adaptive threshold for its separation. By using simple model of eye, we could estimate the sight based on the input from pupil location.**

Next, the obtained sight value is used as input command to the domestic robot. User could control the moving of robot by eye. Also, user could send the voice through text to speech functionality. We use baby infant robot as our domestic robot. We control the robot movement by sending the command via serial communication (utilizing the USB to serial adapter). Three types of command consist of move forward, turn left, and turn right are used in the system for moving the robot. In the robot, we place another camera for capturing the scenery in the front of robot. Between robot and user, they are separated by distance. They are connected over TCP/IP network. The network allows user control the robot remotely. We set the robot as server and user's computer as client. The robot streams the scenery video and receives command sending by the client. In the other place, client (user) receives video streaming from server and control the robot movement by sending command via the network. The user could control the robot remotely even in the long distance because user could see the scenery in the front of robot. We have tested the performance of our robot controlled over TCP/IP network. An experiment measuring the robot maneuverability performance from start point avoiding and passing obstacles has been done in our laboratory. By implementing our system, patient in hospital could self-service by them self.

Keywords-eye-based Human-Computer Interaction (HCI); domestic robot; communication aid; virtual travel

I. INTRODUCTION

Recently, the development of technology for handicap and elderly is growing rapidly. The need for handicap and elderly person in the hospital environment has been given much attention and actively developed by many researchers. According to the United State census in 2000, there are 35 million elderly (12%, with the age more than 65 years). It also reported that the aging has made the percent of people needing help with everyday activities by age become linearly increased. The functional limitations of persons 65 years that has been investigated by US census consist of walking (14.3%), getting outside (15.9%), bathing or showering (9.4%), transferring (9%), dressing (3.9%), using toilet (2.6%), eating (2.1%), preparing meal (8.6%), managing money (7.1%), using the telephone (7.1%), doing light housework (11.4%).

Many robot helpers have been investigated by researchers. The ARM9-based Car controlled remotely has been developed by Wang Shaokun et all [1]. The embedded Linux system was installed under ARM9-structure processor for real time robot operation. It also optimized and improved the versatility and rapid data transmission of wireless remote car. The robots collected the data sensor and relay it to main PC station over WIFI network. Another robot also has been developed by Ding Chengjun et all [2].

Based on embedded WinCE5.0 operating system, they created remote control for mobile robot. The low power consumption and perfect real-time controller is the main goal of this. The data was sent using TCP/IP protocol over WIFI network. Another research concern in remote robot has been developed by Niu Zhigang and Wu Yanbo [3]. They developed a wireless remote control special design for Coal Mine Detection Robot. They investigated the embedded motion control system and apply it for Coal Mine Detection Robot's control system and wireless remote control.

The scenery around the robot in coal mine environment was transmitted to the main station, used it for controlling the robot movement such as forward, backward, turning left, turning right and tipping over the front arm. Ofir H et all have evaluated the telerobotic interface components for teaching robot operation [4].

They evaluated the control method of the robotic arm and the use of three alternative interface designs for robotic operation in the remote learning. Another system has been proposed by He Qingyun et all [5]. They created an embedded system of video capture and the transmission for monitoring wheelchair-bed service robots remotely. The embedded linux, S3C2410AL microprocessor, AppWeb 3.0 server, and block-matching motion estimation were taken into account for obtaining better video compression data. The remote control robot system with a hand-held controller has been proposed by Dmitry Bagayev et all [6]. The dog robot for accompanying the elderly has been proposed by Wei-Dian Lai [7]. The improved interaction technique between users and the robot has been developed for making elderly easy to use.

In this research, we develop a new system in the form of domestic robot helper controlled by eye which allows patient self-service and speaks remotely. The main goal of this research is how we could develop a domestic robot helper that could be used by a handicap person in hospital environment for helping them to do not always depend on the nurse or the assistance.

First, we estimate user sight by placing camera mounted on user glasses. Once eye image is captured, the several image processing are used to estimate the sight. Eye image is cropped from the source for simplifying the area. We detect the centre of eye by seeking the location of pupil. The pupil and other eye component could be easily distinguished based on the color. Because pupil has darker color than others, we just apply adaptive threshold for its separation. By using simple model of eye, we could estimate the sight based on the input from pupil location. Next, the obtained sight value is used as input command to the domestic robot. User could control the moving of robot by eye.

Also, user could send the voice through text to speech functionality. We use baby infant robot as our domestic robot. We control the robot movement by sending the command via serial communication (utilizing the USB to serial adapter). Three types of command consist of move forward, turn left, and turn right are used in the system for moving the robot. In the robot, we place another camera for capturing the scenery in the front of robot. Between robot and user, they are separated by distance. They are connected over TCP/IP network. The network allows user control the robot remotely. We set the robot as server and user’s computer as client. The robot streams the scenery video and receives command sent by the client. In the other place, client (user) receives video streaming from server and control the robot movement by sending command via the network. The user could control the robot remotely even in the long distance because user could see the scenery in the front of robot.

II. Proposed System

A. System Configuration

Our proposed eye-based domestic robot is shown in Figure 1.

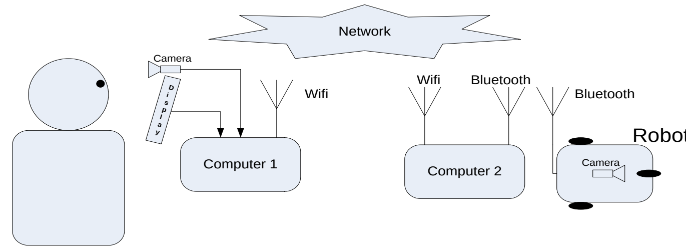


Fig. 1. Block Diagram Of Eye-Based Domestic Robot

The entire system consists of two parts: Client (User) and Server (Robot). Between client and server are separated by different place. The user lay on the bad while robot commutes to everywhere following user control. The signal control is initiated from user eye. We place the camera mounted on user glasses acquiring the eye image. Based on the acquired image and several image processing, we estimate the user sight. We estimate the resulted sight point on display to produce the signal control. The entire system consists of two ways/directions of communication:

(1) from user to robot and
(2) from robot to user.

The first direction of communication sends the command to the robot. After sight estimation, system translate it into the robot command such as go forward, turning left, turning right, and stop. We do not use the backward for safety reason (Moving in backward could cause robot in dangerous situation because user could not see the scenery in the backside. Also, the client could send voice by user typing a word and translate it into voice using text to voice function. Over WIFI network, the client and server connected each other. In the other hands, robot broadcast the scenery video via web camera placed in the robot. After acquiring the scenery, robot stream it to client using TCP/IP network. It will enable robot to be controlled as long as the covered area of the network. In locally, robot is controlled by minicomputer placed on the robot body. They connected using Bluetooth interface.

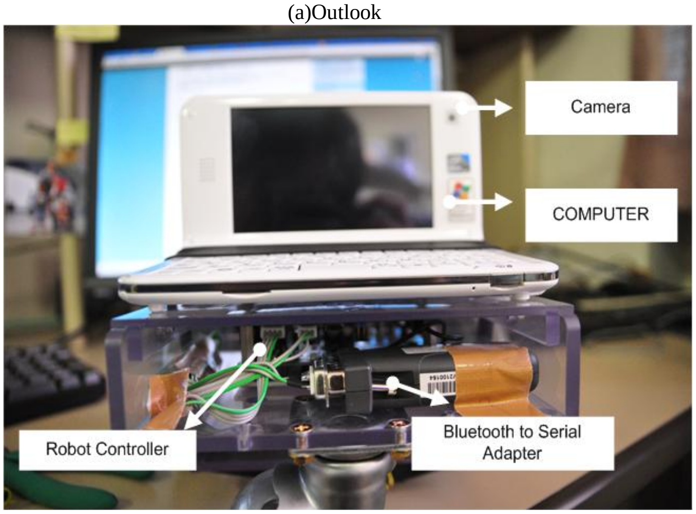
B. System Components

We implement our system by utilizing baby infant type robot with netbook PC as main controller. In the inside of baby infant, there is a controller (tiny power type) enabling us to control its movement via serial communication.

This robot is driven by two motors in the right and left sides. These two motors enable robot moves forward, turn right and left, and also backward. To make the robot turning, we just adjust the combination speed of these two motors. This robot was powered by 11.1 V 2200 mAh lithium battery.

We modify the Bluetooth to serial adapter by adding new power supply taken from tiny power controller. Also we change the mode communication of GBS301 by making a direct connection between RTS and DTS pins. The 5V voltage was taken from this microcontroller to supply the power of Bluetooth to serial adapter. The hardware of our robot is shown in Figure 2.

(a)Outlook



(b)Components

Fig. 2. Outlook And Components Of The Proposed Robot

We make software to allow communication between two PCs or more over TCP/IP network. We utilize Winsock library to establish the connection. We make Client-Server application that will be used to control the robot over the TCP/IP Network. The application software has been created and the communication has been established. By utilizing the WIFI connection, robot could commute to other rooms. The screenshot of this program is shown in Figure 3. Also, the type of streaming data and the direction is shown in Figure 4.

There is automatic control operation mode. In the automatic control mode, user can look at moving pictures of images which are acquired with the camera attached at the tip of the robot as shown in Figure 5 (a). There are five keys which allow switch between menu and automatic modes, left/right/forward selection of arrows, and switching to key-in mode. Therefore, user can move the robot to forward direction, and to turn left or right.

Also, in the key-in mode, sentences can be input. Furthermore, sentences are read-out using voice output software tools. Then user can return to the automatic control mode again.

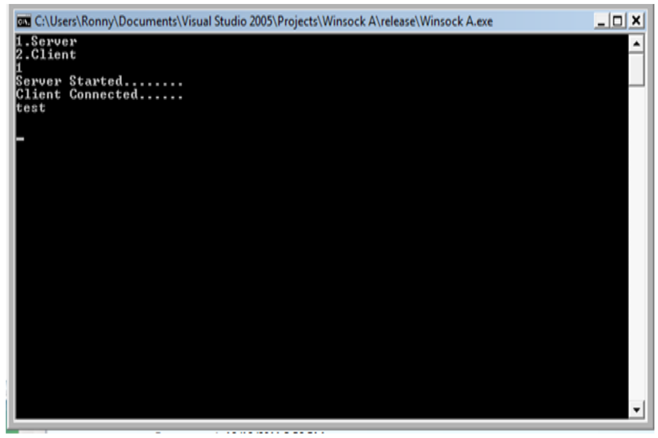


Fig. 3. The Client-Server Application

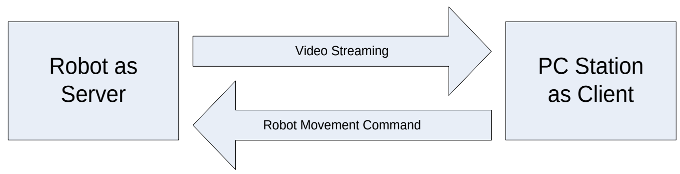
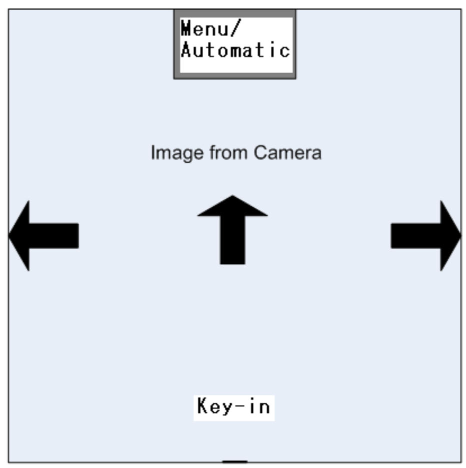
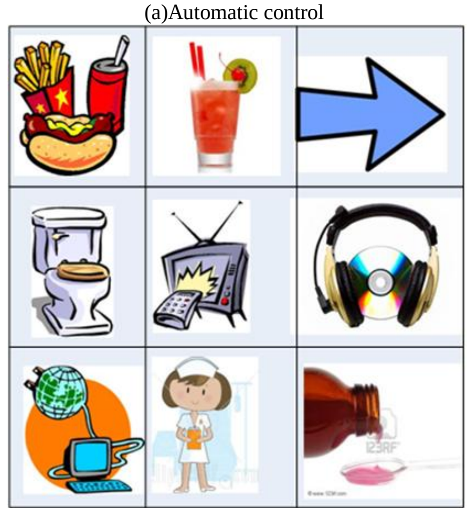


Fig. 4. Client-Server Streaming Connection



(a)Automatic control



(b)Menu control

Fig. 5. Menu For Helper Robot Control

At the initial stage, main menu of Figure 5 (b) appears. From the menu, user chooses one of the menu functions, I need meal, I need drink, move to the automatic control mode, I need toilet, I will watch TV, I would like to listen audio programs, I would like to use Internet, I need a help from nurse, I need medicines.

C. Hardware

There are three major components of the proposed system. One is robotics and the other one is cameras which are mounted on the tip of the robotics and mounted on the glass of which user is wearing. The Head Mount Display: HMD of TAC-EYE LT is also attached to the camera mounted glass. All these three hardware specifications are shown in Table 1, 2, and 3, respectively.

TABLE I. SPECIFICATION OF OKATECH INFANT MINI ROBOT PLATFORM

Size	W213xD233xT105mm
Weight	1.7kg
Max.payload	5kg
Wheel_rotation	8000rpm
Speed	18.12cm/s(0.65km/h)
Motor	12V,1.5W(with_encoder)
Tire	100mm
Caster	60mm
Battery	Lithum_polimer(11.1V,2200mAh)

TABLE II. SPECIFICATION OF THE WEB CAMERA USED

Resolution	1,300,000pixels
Minimum distance	20cm
Frame rate	30fps
Minimum illumination	30lux
Size	52mm(W)x70mm(H)x65mm(D)
Weight	105g

TABLE III. SPECIFICATION OF THE HEAD MOUNT DISDPLAY USED

Pixel_size	SVGA+(852x600)
Field_of_view	30degree
Format	SVGA,NTSC,RS-170,PAL
Specification	MIL-STD_810F
Dry_Battery	USB_Interface(2_batteries)

III. EXPERIMENTS

A. Experimental Conditions

We tested the robot control performance by controlling it moves forward about two meters and return again while we were recording the time. Before experiment, first we have to connect the robot to local network via WIFI connection. Also, the PC station network has to be established. After the connection between them was established, we began to start the server program and video streaming. Also, we start the program on client to receive the video and enable sending robot commands.

The Map used for experiment is shown in Figure 6.

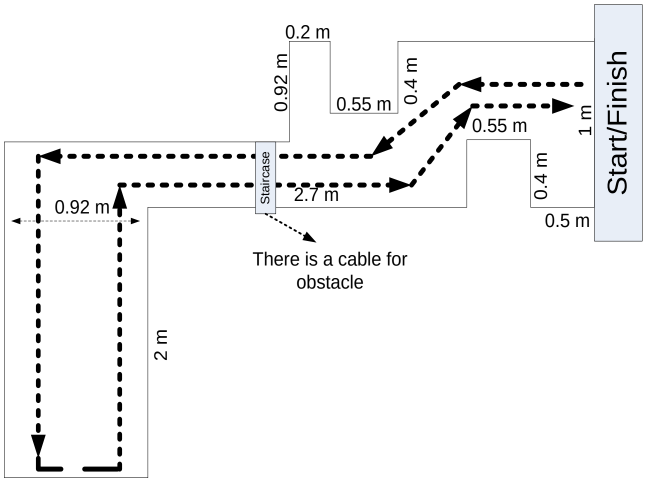


Fig. 5. Route For Domestic Robot In The Experiments

B. Experimental Results

Screen shot image of moving picture of image which is acquired with the camera attached at the tip of robot is shown in Figure 6. Also, the acquired image of hand held camera of which one of the authors traced to the robot is shown in Figure 6 (bottom right). Using this camera acquired image, user can avoid obstacles. Also, the robotics can avoid obstacles using the attached ultrasonic sensor, even if the obstacles are transparent (invisible with the camera).

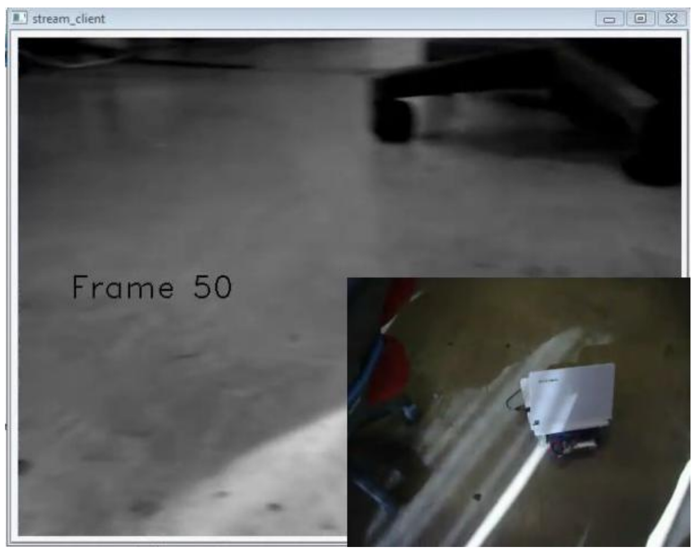


Fig. 6. Acquired Image With The Camera Mounted At The Tip Of Helper Robot And The Image Of Helper Robot From The Hand Held Camera

The robotics control speed is evaluated. The time required for the route stating from the start location to the finished location is measured with five trials. The results are shown in Table 4

TABLE IV. TIME REQUIRED FOR MOVING ALONG THE TRACK

Trial No	Required Time (seconds)
1	177
2	202
3	192
4	182
5	198

The required time ranges from 177 to 202 seconds for traveling the route of the approximately 16 m of travel length. Therefore, robot speed ranges from 7.92 to 9.04 cm/s. It is about half speed in comparison to the specification. This would be good enough for the robot which helps patients to make order, to travel in the hospital virtually, and enjoy conversations with the other persons when the robot meets with them.

IV. CONCLUSION

The proposed eye-based robot helper has been successfully implemented. The real-time remote controller has been successfully implemented and tested in our laboratory with the average time 182 seconds for robot travelling along the 5.75 meters. Also, our robot has good maneuver to avoid the obstacles and pass it. The user easily controls the robot via the transmitted scenery image. By implementing our robot in the real application, it could help the handicap patient when they are sick in the hospital.

REFERENCES

[1] Wang Shaokun, Xiao Xiao, and Zhao Hongwei, "The Wireless Remote Control Car System Based on ARM9," in Instrumentation, Measurement, Computer, Communication and Control, International Conference on , October 2011 , pp. 887-890.

[2] Ding Chengjun, Yan Bingsen, and Duan Ping, "The Remote Control of Mobile Robot Based on Embedded Technology," in Measuring Technology and Mechatronics Automation, International Conference on , January 2011, pp. 907-910.

[3] Niu Zhigang and Wu Yanbo, "Research on Wireless Remote Control for Coal Mine Detection Robot," in Digital Manufacturing and Automation, International Conference on , December 2010 , pp. 315-318.

[4] Ofir H Goldstain, Irad Ben-Gal, and Yossi Bukchin, "Evaluation of Telerobotic Interface Components for Teaching Robot Operation," in IEEE TRANSACTIONS ON LEARNING TECHNOLOGIES, 2011, pp. 365-376.

[5] Niu Zhigang and Wu Yanbo, "Research on Wireless Remote Control for Coal Mine Detection Robot," in Digital Manufacturing and Automation, International Conference on, December 2010, pp. 315-318.

[6] Evsyakov Artem Dmitry Bagayev, "System remote control of the robotized complex - Pegas," in East-West Design & Test Symposium , September 2010 , pp. 358-361.

[7] Wei-Dian Lai, "Zigbee remote control interface for a robot dog design for older users," in Proceedings of the 5th International Confernece on Ubiquitous Information Management and Communication, February 2011 , pp.1-5.

AUTHORS PROFILE

Kohei Arai, He received BS, MS and PhD degrees in 1972, 1974 and 1982, respectively. He was with The Institute for Industrial Science, and Technology of the University of Tokyo from 1974 to 1978 also was with National Space Development Agency of Japan (current JAXA) from 1979 to 1990. During from 1985 to 1987, he was with Canada Centre for Remote Sensing as a Post Doctoral Fellow of National Science and Engineering Research Council of Canada. He was appointed professor at Department of Information Science, Saga University in 1990. He was appointed councilor for the Aeronautics and Space related to the Technology Committee of the Ministry of Science and Technology during from 1998 to 2000. He was also appointed councilor of Saga University from 2002 and 2003 followed by an executive councilor of the Remote Sensing Society of Japan for 2003 to 2005. He is an adjunct professor of University of Arizona, USA since 1998. He also was appointed vice chairman of the Commission “A” of ICSU/COSPAR in 2008. He wrote 30 books and published 332 journal papers

Ronny Mardiyanto: He was born in 1981. He received his M.T degree from Institut Teknologi Bandung, Indonesia in 2006. He has been working as lecturer of Institut Teknologi Sepuluh Nopember Surabaya since 2003 and now is a PhD student in Saga University Japan. His current research interests are robot vision.

Comparison of Supervised and Unsupervised Learning Algorithms for Pattern Classification

R. Sathya
Professor, Dept. of MCA,
Jyoti Nivas College (Autonomous), Professor and Head,
Dept. of Mathematics,
Bangalore, India.

Annamma Abraham
Professor and Head, Dept. of Mathematics
B.M.S.Institute of Technology,
Bangalore, India.

Abstract: **This paper presents a comparative account of unsupervised and supervised learning models and their pattern classification evaluations as applied to the higher education scenario. Classification plays a vital role in machine based learning algorithms and in the present study, we found that, though the error back-propagation learning algorithm as provided by supervised learning model is very efficient for a number of non-linear real-time problems, KSOM of unsupervised learning model, offers efficient solution and classification in the present study.**

Keywords – Classification; Clustering; Learning; MLP; SOM; Supervised learning; Unsupervised learning;

I. INTRODUCTION

Introduction of cognitive reasoning into a conventional computer can solve problems by example mapping like pattern recognition, classification and forecasting. Artificial Neural Networks (ANN) provides these types of models. These are essentially mathematical models describing a function; but, they are associated with a particular learning algorithm or a rule to emulate human actions. ANN is characterized by three types of parameters; (a) based on its interconnection property (as feed forward network and recurrent network); (b) on its application function (as Classification model, Association model, Optimization model and Self-organizing model) and (c) based on the learning rule (supervised/ unsupervised /reinforcement etc.,) [1].

All these ANN models are unique in nature and each offers advantages of its own. The profound theoretical and practical implications of ANN have diverse applications. Among these, much of the research effort on ANN has focused on pattern classification. ANN performs classification tasks obviously and efficiently because of its structural design and learning methods. There is no unique algorithm to design and train ANN models because, learning algorithm differs from each other in their learning ability and degree of inference. Hence, in this paper, we try to evaluate the supervised and unsupervised learning rules and their classification efficiency using specific example [3].

The overall organization of the paper is as follows. After the introduction, we present the various learning algorithms used in ANN for pattern classification problems and more specifically the learning strategies of supervised and unsupervised algorithms in section II.

Section III introduces classification and its requirements in applications and discusses the familiarity distinction between supervised and unsupervised learning on the pattern-class information. Also, we lay foundation for the construction of classification network for education problem of our interest. Experimental setup and its outcome of the current study are presented in Section IV. In Section V we discuss the end results of these two algorithms of the study from different perspective. Section VI concludes with some final thoughts on supervised and unsupervised learning algorithm for educational classification problem.

II. ANN LEARNING PARADIGMS

Learning can refer to either acquiring or enhancing knowledge. As Herbert Simon says, Machine Learning denotes changes in the system that are adaptive in the sense that they enable the system to do the same task or tasks drawn from the same population more efficiently and more effectively the next time.

ANN learning paradigms can be classified as supervised, unsupervised and reinforcement learning. Supervised learning model assumes the availability of a teacher or supervisor who classifies the training examples into classes and utilizes the information on the class membership of each training instance, whereas, Unsupervised learning model identify the pattern class information heuristically and Reinforcement learning learns through trial and error interactions with its environment (reward/penalty assignment).

Though these models address learning in different ways, learning depends on the space of interconnection neurons. That is, supervised learning learns by adjusting its inter connection weight combinations with the help of error signals where as unsupervised learning uses information associated with a group of neurons and reinforcement learning uses reinforcement function to modify local weight parameters.

Thus, learning occurs in an ANN by adjusting the free parameters of the network that are adapted where the ANN is embedded.

This parameter adjustment plays key role in differentiating the learning algorithm as supervised or unsupervised models or other models. Also, these learning algorithms are facilitated by various learning rules as shown in the Fig 1 [2].

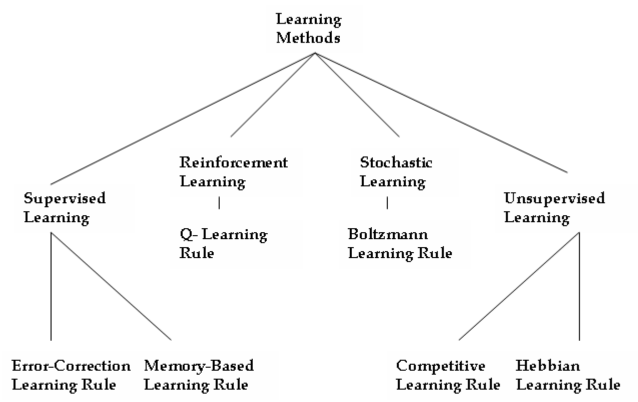


Fig. 1. Learning Rules Of ANN

A. Supervised Learning

Supervised learning is based on training a data sample from data source with correct classification already assigned. Such techniques are utilized in feedforward or MultiLayer Perceptron (MLP) models. These MLP has three distinctive characteristics:

1. One or more layers of hidden neurons that are not part of the input or output layers of the network that enable the network to learn and solve any complex problems
2. The nonlinearity reflected in the neuronal activity is differentiable and,
3. The interconnection model of the network exhibits a high degree of connectivity

These characteristics along with learning through training solve difficult and diverse problems. Learning through training in a supervised ANN model also called as error back-propagation algorithm. The error correction-learning algorithm trains the network based on the input-output samples and finds error signal, which is the difference of the output calculated and the desired output and adjusts the synaptic weights of the neurons that is proportional to the product of the error signal and the input instance of the synaptic weight. Based on this principle, error back propagation learning occurs in two passes:

Forward Pass: Here, input vector is presented to the network. This input signal propagates forward, neuron by neuron through the network and emerges at the output end of the network as output signal: $y(n) = \varphi(v(n))$ where $v(n)$ is the induced local field of a neuron defined by $v(n) = \Sigma\ w(n)y(n)$. The output that is calculated at the output layer $o(n)$ is compared with the desired response $d(n)$ and finds the error $e(n)$ for that neuron. The synaptic weights of the network during this pass are remains same.

Backward Pass: The error signal that is originated at the output neuron of that layer is propagated backward through network. This calculates the local gradient for each neuron in each layer and allows the synaptic weights of the network to undergo changes in accordance with the delta rule as:

$\Delta w(n) = \eta * \delta(n) * y(n)$.

This recursive computation is continued, with forward pass followed by the backward pass for each input pattern till the network is converged [4-7].

Supervised learning paradigm of an ANN is efficient and finds solutions to several linear and non-linear problems such as classification, plant control, forecasting, prediction, robotics etc [8-9]

B. Unsupervised Learning

Self-Organizing neural networks learn using unsupervised learning algorithm to identify hidden patterns in unlabelled input data. This unsupervised refers to the ability to learn and organize information without providing an error signal to evaluate the potential solution. The lack of direction for the learning algorithm in unsupervised learning can sometime be advantageous, since it lets the algorithm to look back for patterns that have not been previously considered [10]. The main characteristics of Self-Organizing Maps (SOM) are:

1. It transforms an incoming signal pattern of arbitrary dimension into one or 2 dimensional map and perform this transformation adaptively
2. The network represents feedforward structure with a single computational layer consisting of neurons arranged in rows and columns.
3. At each stage of representation, each input signal is kept in its proper context and,
4. Neurons dealing with closely related pieces of information are close together and they communicate through synaptic connections.

The computational layer is also called as competitive layer since the neurons in the layer compete with each other to become active. Hence, this learning algorithm is called competitive algorithm. Unsupervised algorithm in SOM works in three phases:

Competition phase: for each input pattern x, presented to the network, inner product with synaptic weight w is calculated and the neurons in the competitive layer finds a discriminant function that induce competition among the neurons and the synaptic weight vector that is close to the input vector in the Euclidean distance is announced as winner in the competition. That neuron is called best matching neuron, i.e. $x = \arg\min \|x - w\|$.

Cooperative phase: the winning neuron determines the center of a topological neighborhood h of cooperating neurons. This is performed by the lateral interaction d among the cooperative neurons. This topological neighborhood reduces its size over a time period.

Adaptive phase: enables the winning neuron and its neighborhood neurons to increase their individual values of the discriminant function in relation to the input pattern through suitable synaptic weight adjustments, $\Delta w = \eta h_{(x)}(x - w)$.

Upon repeated presentation of the training patterns, the synaptic weight vectors tend to follow the distribution of the input patterns due to the neighborhood updating and thus ANN learns without supervisor [2].

Self-Organizing Model naturally represents the neuro-biological behavior, and hence is used in many real world applications such as clustering, speech recognition, texture segmentation, vector coding etc [11-13].

III. CLASSIFICATION

Classification is one of the most frequently encountered decision making tasks of human activity. A classification problem occurs when an object needs to be assigned into a predefined group or class based on a number of observed attributes related to that object. There are many industrial problems identified as classification problems. For examples, Stock market prediction, Weather forecasting, Bankruptcy prediction, Medical diagnosis, Speech recognition, Character recognitions to name a few [14-18]. These classification problems can be solved both mathematically and in a non-linear fashion. The difficulty of solving such problem mathematically lies in the accuracy and distribution of data properties and model capabilities [19].

The recent research activities in ANN prove, ANN as best classification model due to the non-linear, adaptive and functional approximation principles. A Neural Network classifies a given object according to the output activation. In a MLP, when a set of input patterns are presented to the network, the nodes in the hidden layers of the network extract the features of the pattern presented. For example, in a 2 hidden layers ANN model, the hidden nodes in the first hidden layer forms boundaries between the pattern classes and the hidden nodes in the second layer forms a decision region of the hyper planes that was formed in the previous layer. Now, the nodes in the output layer logically combines the decision region made by the nodes in the hidden layer and classifies them into class 1 or class 2 according to the number of classes described in the training with fewest errors on average. Similarly, in SOM, classification happens by extracting features by transforming of m-dimensional observation input pattern into q-dimensional feature output space and thus grouping of objects according to the similarity of the input pattern.

The purpose of this study is to present the conceptual framework of well known Supervised and Unsupervised learning algorithms in pattern classification scenario and to discuss the efficiency of these models in an education industry as a sample study. Since any classification system seeks a functional relationship between the group association and attribute of the object, grouping of students in a course for their enhancement can be viewed as a classification problem [20-22]. As higher education has gained increasing importance due to competitive environment, both the students as well as the education institutions are at crossroads to evaluate the performance and ranking respectively. While trying to retain its high ranking in the education industry, each institution is trying to identify potential students and their skill sets and group them in order to improve their performance and hence improve their own ranking. Therefore, we take this classification problem and study how the two learning algorithms are addressing this problem.

In any ANN model that is used for classification problem, the principle is learning from observation. As the objective of the paper is to observe the pattern classification properties of those two algorithms, we developed Supervised ANN and Unsupervised ANN for the problem mentioned above. A Data set consists of 10 important attributes that are observed as qualification to pursue Master of Computer Applications (MCA), by a university/institution is taken. These attributes explains, the students' academic scores, priori mathematics knowledge, score of eligibility test conducted by the university. Three classes of groups are discovered by the input observation [3]. Following sections presents the structural design of ANN models, their training process and observed results of those learning ANN model.

IV. EXPERIMENTAL OBSERVATION

A. Supervised ANN

A 11-4-3 fully connected MLP was designed with error back-propagation learning algorithm. The ANN was trained with 300 data set taken from the domain and 50 were used to test and verify the performance of the system. A pattern is randomly selected and presented to the input layer along with bias and the desired output at the output layer. Initially each synaptic weight vectors to the neurons are assigned randomly between the range [-1,1] and modified during backward pass according to the local error, and at each epoch the values are normalized.

Hyperbolic tangent function is used as a non-linear activation function. Different learning rate were tested and finally assigned between [0.05 - 0.1] and sequential mode of back propagation learning is implemented. The convergence of the learning algorithm is tested with average squared error per epoch that lies in the range of [0.01 – 0.1]. The input patterns are classified into the three output patterns available in the output layer. Table I shows the different trial and error process that was carried out to model the ANN architecture.

TABLE I: SUPERVISED LEARNING OBSERVATION

No. of hidden neurons	No. of Epochs	Mean-squared error	Correctness on training	Correctness on Validation
3	5000 - 10000	0.31 – 0.33	79%	79%
4	**5000 - 10000**	**0.28**	**80% - 85%**	**89%**
5	5000 - 10000	0.30 – 0.39	80% - 87%	84%

B. Unsupervised ANN

Kohonen's Self Organizing Model (KSOM), which is an unsupervised ANN, designed with 10 input neurons and 3 output neurons. Data set used in supervised model is used to train the network. The synaptic weights are initialized with 1/√ (number of input attributes) to have a unit length initially and modified according to the adaptability.

Results of the network depends on the presentation pattern of the input vector for small amount of training data hence, the training patterns are presented sequentially to the NN.

Euclidean distance measure was calculated at each iteration to find the winning neuron. The learning rate parameter initially set to 0.1, decreased over time, but not decreased below 0.01. At convergence phase it was maintained to 0.01 [11]. As the competitive layer is one dimensional vector of 3 neurons, the neighborhood parameter has not much influence on the activation. The convergence of the network is calculated when there were no considerable changes in the adaptation. The following table illustrates the results:

TABLE II. UNSUPERVISED LEARNING OBSERVATION

Learning rate parameter	No. of Epochs	Correctness on training	Correctness on Validation
.3 - .01	1000 - 2000	85%	86%
.1 - .01	**1000 - 3000**	**85% - 89%**	**92%**

V. RESULTS AND DISCUSSION

In the classification process, we observed that both learning models grouped students under certain characteristics say, students who possess good academic score and eligibility score in one group, students who come from under privileged quota are grouped in one class and students who are average in the academics are into one class.

The observation on the two results favors unsupervised learning algorithms for classification problems since the correctness percentage is high compared to the supervised algorithm. Though, the differences are not much to start the comparison and having one more hidden layer could have increased the correctness of the supervised algorithm, the time taken to build the network compared to KSOM was more; other issues we faced and managed with back-propagation algorithm are:

1. *Network Size*: Generally, for any linear classification problem hidden layer is not required. But, the input patterns need 3 classifications hence, on trail and error basis we were confined with 1 hidden layer. Similarly, selection of number of neurons in the hidden layer is another problem we faced. As in the Table I, we calculated the performance of the system in terms of number of neurons in the hidden layer we selected 4 hidden neurons as it provides best result.
2. *Local gradient descent*: Gradient descent is used to minimize the output error by gradually adjusting the weights. The change in the weight vector may cause the error to get stuck in a range and cannot reduce further. This problem is called local minima. We overcame this problem by initializing weight vectors randomly and after each iteration, the error of current pattern is used to update the weight vector.
3. *Stopping Criteria*: Normally ANN model stops training once it learns all the patterns successfully. This is identified by calculating the total mean squared error of the learning. Unfortunately, the total error of the classification with 4 hidden neuron is 0.28, which could not be reduced further. When it is tried to reduce minimum the validation error starts increasing. Hence, we stopped the system on the basis of correctness of the validation data that is shown in the table 89%. Adding one more neuron in the hidden layer as in the last row of Table I increase the chance of over fitting on the train data set but less performance on validation.
4. The only problem we faced in training of KSOM is the declaration of learning rate parameter and its reduction. We decreased it exponentially over time period and also we tried to learn the system with different parameter set up and content with 0.1 to train and 0.01 at convergence time as in Table II.

Also, unlike the MLP model of classification, the unsupervised KSOM uses single-pass learning and potentially fast and accurate than multi-pass supervised algorithms. This reason suggests the suitability of KSOM unsupervised algorithm for classification problems.

As classification is one of the most active decision making tasks of human, in our education situation, this classification might help the institution to mentor the students and improve their performance by proper attention and training. Similarly, this helps students to know about their lack of domain and can improve in that skill which will benefit both institution and students.

VI. CONCLUSION

Designing a classification network of given patterns is a form of learning from observation. Such observation can declare a new class or assign a new class to an existing class. This classification facilitates new theories and knowledge that is embedded in the input patterns. Learning behavior of the neural network model enhances the classification properties. This paper considered the two learning algorithms namely supervised and unsupervised and investigated its properties in the classification of post graduate students according to their performance during the admission period. We found out that though the error back-propagation supervised learning algorithm is very efficient for many non- linear real time problems, in the case of student classification KSOM – the unsupervised model performs efficiently than the supervised learning algorithm.

REFERENCES

[1] L. Fu., Neural Networks in Computer Intelligence, Tata McGraw-Hill, 2003.

[2] S. Haykin, Neural Networks- A Comprehensive Foundation, 2nd ed., Pearson Prentice Hall, 2005.

[3] R. Sathya and A. Abraham, "Application of Kohonen SOM in Prediction", In Proceedings of ICT 2010, Springer-Verlag Berlin Heidelberg, pp. 313–318, 2010.

[4] R. Rojas, The Backpropagation Algorithm, Chapter 7: Neural Networks, Springer-Verlag, Berlin, pp. 151-184, 1996.

[5] M. K. S. Alsmadi, K. B. Omar, and S. A. Noah, "Back Propagation Algorithm: The Best Algorithm Among the Multi-layer Perceptron Algorithm", International Journal of Computer Science and Network Security, Vol.9, No.4, 2009, pp. 378 – 383.

[6] X. Yu, M. O. Efe, and O. Kaynak, "A General Backpropagation Algorithm for Feedforward Neural Networks Learning", IEEE Trans. On Neural Networks, Vol. 13, No. 1, 2002, pp. 251 – 254.

[7] Jin-Song Pei, E. Mai, and K. Piyawat, "Multilayer Feedforward Neural Network Initialization Methodology for Modeling Nonlinear Restoring Forces and Beyond", 4th World Conference on Structural Control and Monitoring, 2006, pp. 1-8.

[8] Awodele, and O. Jegede, "Neural Networks and Its Application in Engineering", Proceedings of Informing Science & IT Education Conference (InSITE) 2009, pp. 83-95.

[9] Z. Rao, and F. Alvarruiz, "Use of an Artificial Neural Network to Capture the Domain Knowledge of a Conventional Hydraulic Simulation Model", Journal of HydroInformatics, 2007, pg.no 15-24.

[10] T. Kohonen, O. Simula, "Engineering Applications of the Self-Organizing Map", Proceeding of the IEEE, Vol. 84, No. 10, 1996, pp.1354 – 1384

[11] R. Sathya and A. Abraham, "Unsupervised Control Paradigm for Performance Evaluation", International Journal of Computer Application, Vol 44, No. 20, pp. 27-31, 2012.

[12] R. Pelessoni, and L. Picech, "Self Organizing Maps - Applications and Novel Algorithm Design, An application of Unsupervised Neural Netwoks in General Insurance: the Determination of Tariff Classes".

[13] P. Mitra, "Unsupervised Feature Selection Using Feature Similarity", IEEE Transaction on Pattern Analysis and Machine Intelligence", Vol. 24, No. 4, pp. 2002.

[14] Moghadassi, F. Parvizian, and S. Hosseini, "A New Approach Based on Artificial Neural Networks for Prediction of High Pressure Vapor-liquid Equilibrium", Australian Journal of Basic and Applied Sciences, Vol. 3, No. 3, pp. 1851-1862, 2009.

[15] R. Asadi, N. Mustapha, N. Sulaiman, and N. Shiri, "New Supervised Multi Layer Feed Forward Neural Network Model to Accelerate Classification with High Accuracy", European Journal of Scientific Research, Vol. 33, No. 1, 2009, pp.163-178.

[16] Pelliccioni, R. Cotroneo, and F. Pung, "Optimization Of Neural Net Training Using Patterns Selected By Cluster Analysis: A Case-Study Of Ozone Prediction Level", 8th Conference on Artificial Intelligence and its Applications to the Environmental Sciences", 2010.

[17] M. Kayri, and Ö. Çokluk, "Using Multinomial Logistic Regression Analysis In Artificial Neural Network: An Application", Ozean Journal of Applied Sciences Vol. 3, No. 2, 2010.

[18] U. Khan, T. K. Bandopadhyaya, and S. Sharma, "Classification of Stocks Using Self Organizing Map", International Journal of Soft Computing Applications, Issue 4, 2009, pp.19-24.

[19] S. Ali, and K. A. Smith, "On learning algorithm selection for classification", Applied Soft Computing Vol. 6, pp. 119–138, 2006.

[20] L. Schwartz, K. Stowe, and P. Sendall, "Understanding the Factors that Contribute to Graduate Student Success: A Study of Wingate University's MBA Program".

[21] B. Naik, and S. Ragothaman, "Using Neural Network to Predict MBA Student Success", College Student Journal, Vol. 38, pp. 143-149, 2004.

[22] V.O. Oladokun, A.T. Adebanjo, and O.E. Charles-Owaba, "Predicting Students' Academic Performance using Artificial Neural Network: A Case Study of an Engineering Course", The Pacific Journal of Science and Technology, Vol. 9, pp. 72–79, 2008.

Hybrid of Rough Neural Networks for Arabic/Farsi Handwriting Recognition

Elsayed Radwan
Computer Science Department, Faculty of Computer and Information Sciences,
Mansoura University, Egypt, P.O. Box: 35516
Assistant Prof., Institute of Scientific Research and Revival of Islamic Heritage,
Umm Al-Qura University, KSA.

***Abstract*—Handwritten character recognition is one of the focused areas of research in the field of Pattern Recognition. In this paper, a hybrid model of rough neural network has been developed for recognizing isolated Arabic/Farsi digital characters. It solves the neural network problems; proneness to overfitting, and the empirical nature of model development using rough sets and the dissimilarity analysis. Moreover the perturbation in the input data is violated using rough neuron. This paper describes an evolutionary rough neural network based technique to recognize Arabic/Farsi isolated handwritten digital characters. This method involves hierarchical feature extraction, data clustering and classification. In contrast with conventional neural network, a comparative study is appeared. Also, the details and limitations are discussed.**

Keywords- Rough Sets; Rough Neural Network; Arabic/Farsi Digit Recognition; Dissimilarity Analysis; and Classification.

I. INTRODUCTION

There are various ways where computers directly take input from the human information system like Optical Character Recognition (OCR), speech recognition, symbolic (Icons, windows) interactive communication etc. Such systems are difficult to design and do not provide complete error free operation. General solution for speech or character recognition is very complex. Even fastest computers would need much larger computational time compared to human response time to perform the same job. To get working solution for the above class of problems, domain specific solutions should be much more efficient [9]. Also, combing different techniques is needed to overcome the shortcomings in each other. In hand-written character recognition problem, the domain is reduced to a small subset of characters of limited number of written characters using specific style. This subset is then further classified to smaller subsets, where each group represents a character [6, 9].

According to previous research [6, 15], the complexity of the handwritten character recognition is greatly increased by the noise problem. Moreover, it should be influenced by the almost infinite inconsistency of handwriting as the result of the writer and the nature of the writing. These characteristics make the progress of handwritten more complex and difficult than typewritten. Thus, a good and effective tool to deal with vagueness and uncertainty of information is needed. Rough Sets [10]is used in the pre-processing stage in this paper that give the ability for dealing with inconsistency and noise reduction on the handwritten characters. This comes from the fact that Rough Sets can handle the missing feature value and inconsistency on the pattern [10, 12]. Moreover, rough sets are able to get the most essential part of the knowledge with minimum number of features based on its concept of reduct. The feature quality representations that are yielded from rough sets of a data dominate in typewritten character recognition, though it is well suited to all data analysis methods. Because of the nature of handwriting, the perturbation in the specific input data, pairwise representation, should be measured and treated, so a cluster based technique is needed. However the quality representation can be translated to this type of data into a pairwise representation using rough clustering method and problem dependent similarity measure. The main goal of this translation type is that a pairwise representation captures the structure that is captured by the quality data [1, 11]. By transferring quality data into a pairwise, the interpretation the significance of the individual feature is lost, so it is implicitly imbedded by the measure of dissimilarity result.

Since Real life data sets are relatively erroneous, there is an increasing need for effective tools that are able to deal with non-linear problems. Also, incorporating architectural changes result in improving accuracy of approximation. So far, nonlinear problems have been dealt with ANN (Artificial Neural Networks) [15]. ANN was used for pattern recognition for its advantages of, parallel processing and certain fault tolerance [15, 9]. But, this method also has its weaknesses, just as the learning time will be soaring and easy to sink into a local minimum point with the increase of the dimensions [4].

In order to optimize the ANN structure to improve the learning efficiency, a hybrid model based on Rough Sets and Artificial Rough Neural Network (RS-RNN) is proposed. RS-RNN has a drawback that input neurons with zero activation energy negatively affect processing time and space so using rough set reduction algorithm is essential for reducing these superfluous neurons. Moreover, the perturbation in the sample data set is defined as a lower approximation and upper approximation to introduce the idea of the rough neuron [5]. This method takes Rough Set as the modified disposal system of ANN to simplify the structure of it, and to reduce the attribute index and the sample numbers. So a practical method with theoretical support and methodological guarantee is provided efficiently to establish the Arabic/Farsi hand writing recognition system.

Although conventional neural network achieves a good ability to detect all possible interactions between predicted variables, and the availability of multiple training algorithms, it still suffer for different problems such as proneness to over fitting, and the empirical nature of model development. Combining Rough Sets solves the above mentioned problems of ANN and enhance its performance by discovering and removing input neurons with zero weights. In addition, rough neuron treats the problem of data perturbation. In contrast with ANN, a comparative study is mentioned. Database of handwritten Arabic/Farsi sample , IFHCDB (Isolated Farsi Handwritten Character Database) which is created at Amirkabir University of Technology (AUT) and isolated typewritten characters represented by 10×10 pixels [14], is used for more compromise results.

This paper is organized as follows; in section 2 a brief introduction of important fields (rough neural network, rough set) is discussed. Section 3 describes the fundamentals of our method where the method RS-RNN and its performance are given. In Section 4 examine the application and guide the user using it. Then we conclude with section 5 the purpose of that paper and its results.

II. Preliminaries

This section briefs on the basic notions of rough sets that is used in this paper and the detailed definitions can be referred to some related papers [1, 3, 10, 12].

A. Rough Sets theory

Rough set theory is a new mathematical approach to imperfect knowledge [10, 12]. The principle notion of Rough Sets is that lowering the principle in data representation makes it possible to uncover patterns in the data, which may otherwise be obscured by too many details. At the basis of Rough Sets theory is the analysis of the limits of discernibility of subsets ***X*** of objects from the universe of discourse ***U***.

Let U be a set of objects (universe of discourse), A be a set of attributes. A decision system is an attribute value table, in which objects of the universe and columns label rows by the attribute. It takes the form $DT = (U, A \cup \{d\})$, where $d \notin A$ is the decision attribute. The elements of A are called conditions attributes and an attribute $a \in A$ can be regarded as a function from the domain $\boldsymbol{U}$ to some value set $V_a, a : U \to V_a$.

With every subset of attribute $B \subseteq A$ one can easily associate an equivalence relation I_B on U: $I_B = \{(x, y) \in U : \text{for every } a \in B, a(x) = a(y)\}$. Then $I_B = \cap_{a \in B} I_a$. If $X \subseteq U$, the sets $\{x \in U : [x]_B \subseteq X\}$ and $\{x \in U : [x]_B \cap X \neq \phi\}$ where $[x]_B$ denotes the equivalence class of the object $x \in U$ relative to I_B, are called the B-lower and B-upper approximation of X in S and denoted by $\underline{B}X, \overline{B}X$. The aim of Rough Sets is to obtain irreducible but essential parts of the knowledge encoded by the given information system; these would constitute *reducts* of the system. So one is, in effect, looking for maximal sets of attributes taken from the initial set (*A*, say) which induce the same partition on the domain as *A*. In other words, the essence of the information remains intact, and superfluous attributes are removed. Reducts have nicely characterized in [12] by discernibility matrices and discernibility functions.

A principle task is the method of rule generation is to compute relative to a particular kind of information system, the decision system. R-reducts and d-discernibility matrices are used for this purpose [13]. The methodology is described below.

Let $S = < U, A >$ be a decision table, $A = C \cup D$, with C and $D = \{d_1, \ldots, d_l\}$ its sets of condition and decision attributes respectively. Divide the decision table $S = < U, A >$ into l tables $S_i = < U_i, A_i >$, $i = 1, \ldots, l$ corresponding to the l decision attributes $d_1, \ldots, d_l$ where $U = U_1 \cup \ldots \cup U_l$ and $A_i = C \cup \{d_i\}$.

Let $\{x_{i1}, .. x_{ip}\}$ be the set of those objects of U_i that occur in $S_i, i = 1, \ldots, l$. Now for each d_i -reduct $B = \{b_1, \ldots, b_k\}$, a discernibility matrix is defined as follows

$$c_{ij} = \{a \in B : a(x_i) \neq a(x_j)\} \;,\; i, j = 1, \ldots, n \tag{1}$$

for each object $x_j \in x_{i1}, \ldots, x_{ip}$ the discernibility function

$$f_{d_i}^{x_j} = \wedge\{\vee(c_{ij}) : 1 \leq i, j \leq n, j < i, c_{ij} \neq \phi\} \tag{2}$$

where $\vee(c_{ij})$ is disjunction of all members of c_{ij}. Then $f_{d_i}^{x_j}$ is brought to its conjunctive normal form (c.n.f). One of thus obtains as a dependency rule $r_i : P_i \to d_i$ where P_i is disjunctive normal form (d.n.f) of $f_{d_i}^{x_j}$, $j \in i_1, \ldots, i_p$. The dependency factor γ_i for r_i is given by

$$\gamma_i = \frac{card(POS_i(d_i))}{card(U_i)}, \tag{3}$$

Where $POS_i(d_i) = \cup_{X \in Id_i} l_i(X)$ and $l_i(X)$ is the lower approximation of X with respect to I_i . The Coefficient $\bar{\gamma}_i = 1 - \gamma_i$ is called the inconsistency degree of DT[13].

B. Rough Neuron

Rough Neuron [5, 16] was developed with an aim of classifying a set of objects into three parts based on a given condition i.e. into the lower, the upper and the negative regions. Rough neural networks [5] consist of both conventional as well as the rough neurons in a fully connected fashion. The rough neuron consists of two individual neurons called the upper bound neuron and the lower bound neuron, which have a mode of sharing information as demonstrated in Figure 1. The lower bound neuron, deals only with the definite or certain part of the input data and generates its output signal called as the lower boundary-signal. The second neuron called the upper boundary neuron processes only that part of the input data which lies in the upper boundary region evaluated based on the concepts of rough sets and generates the output called upper boundary Signal. This interpretation of upper and lower boundary regions is limited only to the learning or training stage of the neural network.

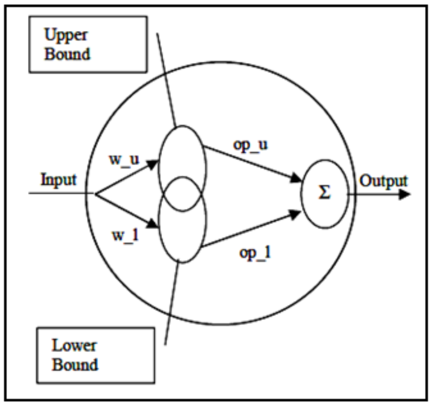


Fig. 1. Rough neuron structure

Tough the neurons (lower and upper) have a conventional sigmoid transfer; the actual output of the lower and upper neurons is given as the following functions

$$output_{\bar{r}} = \max(transfer(x_{\bar{r}}), transfer(x_{\underline{r}})) \quad (4)$$

$$output_{\underline{r}} = \min(transfer(x_{\bar{r}}), transfer(x_{\underline{r}})) \quad (5)$$

Where

$$x_{*} = input_i = \sum_{j:j\ connected\ \ with\ i} w_{ij} \, .\, output_j \quad (6)$$

Training in rough neural is similar to conventional neural network [16]. During training the network use inductive learning principle to learn from the training set. In supervised training the desired output from output neurons in the training set is known, the weight is modified using learning equation. Neural network use back propagation technique for training. Training using rough Back Propagation performs gradient descent in weight space on an error function.

III. Handwriting Recognition based on Rough Neuron

Arabic/Farsi handwriting recognition is widely accepted as the means of document authentication, authorization and personal verification in the modern societies of Middle East. For legality most documents like bank cheques, travel passport and academic certificate need to have authorized real time handwritten verification. Thus there is a need for automatic verification, although the difficulties faced in visual assessment of different types and different fonts [9].

In order to recognize handwritten characters, we need to digitize the position of the path of a drawing tool on a writing surface. A primary requirement of a handwriting recognition system is to allow handwriting anywhere on a writing surface and characters of any size.

The program requires the user to draw one or more copies of each character. As characters are drawn, bounding rectangles are calculated for each character. A hand-drawn character can be of any size because the bounding rectangle is used to normalize the image of the character to fit into a small two-dimensional grid that is used for input to a neural network [15]. As the result of normalization, missing values and noise should be appeared. Moreover inconsistency among different patterns for the same user should be discovered. Thus, a good and effective tool to deal with vagueness and uncertainty of information is needed to extract the local features from pattern. Also, an innovative classifier which treats the inconsistency and the perturbation among patterns should be used. Hence, Rough Sets [18] are used in noise reduction and discovering the most admissible local feature, core feature. Also, depending on rough sets methodology of dissimilarity analysis [17], the differences among patterns are localized, i.e. the architecture of the neural network for the whole patterns is designed. Based on the dis-similarity measure, the location where data are perturbed, to define the position of rough neuron, is defined. As the result, the optimal architecture of rough neural networks is detected. Wherever, the superfluous neuron is removed by data reduction of rough sets, the rough neurons in the input pattern are located by the dissimilarity analysis. Then, the outputs of rough neuron (r) are calculated from equations (4, 5). Finally, the conventional neurons in the input pattern are located in correspondence with the other local features without perturbation. The output of the conventional neuron is a function of the output combined from the rough neuron as equation (7)

$$output = \frac{output_{\bar{r}} - output_{\underline{r}}}{ave\,(output_{\bar{r}}, output_{\underline{r}})} \quad (7)$$

The above function uses the difference between outputs of the upper and lower neurons and normalizes it by the average of the outputs of upper and lower neurons. As the result, each input grid point (local feature) represents the value of an input neuron in a neural network.

In training stage, hand-drawn characters can be of any size, but they are down-sampled to a specific size of grid. The program must determine when the program user is done drawing an individual character. This determination is made by recording the current time in milliseconds when each mouse down movement is recorded. After no mouse down movements has occurred for 400 milliseconds, it is assumed that the training character is no longer being drawn.

A. Rough Sets in the Pre-Processing

Hand-drawn characters can be of any size, but they are down-sampled to a specific size of grid. In this stage, we concentrate on Arabic/Farsi digits that can be represented by eleven features, typewritten digits, as demonstrated in Figure 2.

Some of the attributes that represent Arabic/Farsi digits are superfluous. The superfluous attribute can be reduced using rough set reduction algorithm (using *discernibility Matrix*) and the result is then clustered and applied to rough neural network.

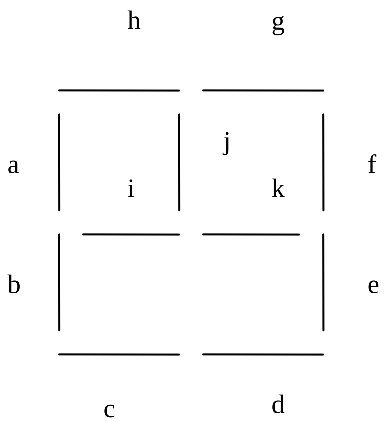


Fig. 2. Template For Arabic/Farsi Digits

Now we will use rough sets for data analysis. Rough Sets theory provides tools for expressing inexact dependencies within data. A Minimum Description Length Principle (MDL-Principle) gives us the reason of why we will use rough sets to reduce the input features of the data. It states, generally speaking, that rules of the most simplified construction, which preserve consistency with data, are likely to classify so far unseen objects with the lowest risk of error. Therefore, to enable classified more objects with high accuracy, it needs to neglect features being the source of redundant information, i.e. to use what is called reduct of attributes. A reduct is a subset of attributes such that it's enough to consider only the features that belong to this subset and still have the same amount of information. If the Decision table has *n* objects, so by $\Theta(DT)$ we denoted an $n \times n$ matrix (c_{ij}), where c_{ij} is defined by equation (1), called the discernibility matrix of *DT*. The main idea of reduct algorithm is that if a set of attributes satisfies the consistency criterion (i.e. be sufficient to discern all the required objects), it must have a non-empty intersection with non-empty elements of the discernibility matrix. One can prove that $a \in CORE(C)$, if and only if there exist two objects, which have the same value for each attribute from C except a. This statement may be expressed by mean of matrix elements (c_{ij}), as given in Figure 3.

The input data to the model will be quantized first, i.e. the features defined the problem should be identified and labeled. If the input data u_i are given, one has to divide the data into distinct sets and introduce new logical input variable s_k such that:

$$If\ (u_i \in X_j) THEN (s_k = label(u_i) = true), i \in I, j \in J \qquad (8)$$

"Min" performs an operation that is analogical to checking for prime implicates of a Boolean function. The returned value is true if the argument R does not contain redundant attributes. Depending on the reduced set of attribute the robustness of the neural networks will be increase, according to the following theorem [2];

***Theorem 1*:** Let $F(u_1,u_2,\ldots\ldots\ldots.,u_n)$ be an arbitrary linearly separable function π is the hyper plane separating vertices. Then with decreasing the dimensionality from (n) to (n-1) , the distance of vertices from π in (n-1) dimensions cannot decrease. □

Thus;

Corollary 2:

Decreasing number of input neurons increase pattern robustness and reduce tolerance

Proof: it is proved by [2].

At present the reduction algorithm always focuses on reducing attributes and aims at obtaining the best attributes reduction. Potential knowledge contained in data is always targeted when we analyse the database. The complexity of the information system can be reduced by attributes reduction, although not all attribute values of each rule are necessary in the reduced information table, so the dis-similarity analysis among different objects is needed [10]

Input: decision table DT of signature images in binary form for particular person with set of rules describing rough neuron
Output: DT with minimal set of rules describing rough neuron such that the superfluous rules is considered to be conventional neuron (distinguish rough and crisp neurons)

- Compute indiscernibility matrix $\Theta(DT) = (\theta_{ij})$ using equation (1)
- Eliminate any empty or non-minimal elements of $\Theta(DT)$ and create a discernibility list , $k = (k_1, k_2, \ldots, K_l)e$, where e is the number of any non empty element in Θ(DT)
- Build families of sets $R_0, R_1, \ldots, R_e$ in the following way

$\circ$Set $R_0 = \phi, i = 1$
$\circ$ *while* $i \le e$ do
- $S_i = \{R \in R_{i-1} : R \cap k_i \neq \phi\}$
- $T_i = \bigcup_{k \in k_i} \bigcup_{R \in R_{i-1} : R \cap k_i \neq \phi} \{R \cup \{k\}\}$
- $M_i = \{R \in T_i : Min(R, k, i) = true\}$
- $R_i = S_i \cup M_i$
- $i = i + 1$

Fig. 3. Reduct Computation

B. Rough Sets in Pattern Clustering

Since the nature of human writer differs from one another, the challenge is to discover pattern inherent among different patterns. Because changes in general lead to uncertainty, the appropriate approaches for uncertainty modelling in order to capture , model and predicate the perspective phenomena is considered in dynamic environment. As a consequence, the combination of dynamic data mining and soft computing is very promising. Rough clustering [3] offers strong tool to detect such changing data structure and aggregate each pixel in its correct cluster.

The appropriate clusters are considered in accordance with the feature reduced by rough set, reduct set. When the new data h in each pattern i is considered to be located at cluster k if it is near to the existing cluster center υ_k, which is defined by the following equation

$$d_{kh} \leq \frac{1}{2}\min\left(d(\upsilon_j,\upsilon_k)\right) \tag{9}$$

Where $d(\upsilon_j,\upsilon_k)$ is the distance between adjacent cluster centers υ_j and υ_k and d_{kh} is the distance between pixel data h and the class centerυ_k.

If the new data h in each pattern i does not fit well into the existing cluster where these pixels are far away from current cluster center υ_k, new clusters should be formed. Whenever, a little of new pixels are discovered far away from existing clusters centers, they might form a noise that should be removed from our pattern. The number of noise pixels F that are far away from existing cluster center should at most less than the average number of pixels in the lower approximation of the existing clusters weighted by the multiplier f

$$F < f\,\frac{\underline{M}^i}{K^i} = f\,\frac{\sum_{k=1}^{K^i}\underline{M}_k^i}{K^i} \tag{10}$$

Where $\underline{M}^i = \sum_{k=1}^{K^i}\underline{M}_k^i$ is the total number of pixels in the lower approximations, the strict inequality of equation (9), in the pattern i and $K^i$is the total number of clusters at pattern i. This criterion requires the setting of multiplier f. The smaller f is, the smaller number of noise pixels and thus the greater the number of pixels needs to be established a new cluster.

C. Dissimilarity in Data Analysis

Without knowledge of the domain and specifically the data set description, finding an appropriate weighting to give reasonable result would be computationally expensive. Since rough sets is able to measure dissimilarity between records of boolean value and to compute the knowledge that represent each pattern, attributes of the same value for all records are disregarded and measure to be zero distance. On the other hand, attributes with different values are considered to be dissimilar. There are many dissimilarity / similarity measures that can be used for the comparison of objects studied in the space of measured variables [7, 11].

To fully understand the descriptor which is based on the centroid distance function, it is essential to foremost understand how one computes the centroid. In this section, we note that the formulas were found from [19]. The position of the centroid, the center of gravity in a pattern i, is fixed in relation to the shape in cluster k. The centroid can be calculated by taking the average of all the points that are defined inside a cluster. Under the assumption that the shape in a cluster k is simply connected, we can compute the centroid simply by using only the boundary points. Let Nbe the total number of points on the border of our clusterk. Here, $n \in [0, N-1]$. The x and y coordinates of the centroid, denoted by ${g_x}^k$ and ${g_y}^k$ respectively, are given by:

$$\begin{cases} {g_x}^k = \frac{\sum_{i=0}^{N-1}(x_i + x_{i+1})(x_i y_{i+1} - x_{i+1} y_i)}{6A_k} \\ {g_y}^k = \frac{\sum_{i=0}^{N-1}(y_i + y_{i+1})(x_i y_{i+1} - x_{i+1} y_i)}{6A_k} \end{cases} \tag{11}$$

Here, the area of the shape, A, is given by the following equation:

$$A_k = \left(\frac{1}{2}\right)\left|\sum_{i=0}^{N-1}(x_i y_{i+1} - x_{i+1} y_i)\right| \tag{12}$$

The centroid distance function expresses the distances of the boundary points from the centroid (g_x^k, g_y^k) of a cluster center$\upsilon_k = (x_{\upsilon_k}, y_{\upsilon_k})$. It is given by the following formula

$$d_k^i = \sqrt{\bar{\gamma}_k[(x_{\upsilon_k} - g_x^k)^2 + (y_{\upsilon_k} - g_y^k)^2]} \tag{13}$$

For each pattern i, d_k^i represents the perturbation of the current data in cluster k that takes into account the cluster significance, where $\bar{\gamma}_k$ is the inconsistency coefficient of cluster k. Thus the level of significance, tolerance, for cluster k at pattern i is given by

$$\alpha_k^i = \frac{d_k^i}{\max_j(d(\upsilon_j,\upsilon_k))} \tag{14}$$

Where $d(\upsilon_j,\upsilon_k)$ is the distance between two adjacent cluster centers υ_j and υ_k. Hence the accepted variations captured in each pattern i forms an interval valued feature values;

$$\left[1 - \tfrac{1}{2}\alpha_k^i, 1 + \tfrac{1}{2}\alpha_k^i\right] \tag{15}$$

Hence, superfluous neuron, produced by the rough sets, is considered to be zero input and the other are considered to be rough neuron. If d_k^i is approximately zero then the corresponding neuron is a conventional neuron with input to be one, otherwise the lower and upper neurons inputs are dominated by the interval bounds of equation (15).

D. Adapting the Rough Neural Network

The proposed system uses a Back propagation (BP) RNNS for classification process. In which, the captured characters are trained by the rough neural network using back propagation technique. Back propagation, or propagation of error, is a common method of teaching artificial neural networks how to perform a given task. [4, 8].

The structure of RNN consists of input layer, hidden layer, and output layer [5]. Neurons in the input layer are rough neurons. The number of nodes in the input layer differs according to the feature vector dimensionality. The number of neurons in the input layer is equal to the number of clusters corresponding to each digit pattern. Each input neuron is given its value by equation (15). The hidden layer consists of conventional neurons; the number of neurons in the hidden layer is approximately double the input layer size. The input of the hidden layer neuron is given by equation (7). The output layer consists of four crisp neuron that represent the binary representation corresponding to each Arabic/Farsi digit.

It is a supervised learning method, and is an implementation of the Delta rule. It requires a teacher that knows, or can calculate, the desired output for any given input. It is most useful for feed-forward networks (networks that have no feedback, or simply, that have no connections that loop). The term is an abbreviation for "backwards propagation of errors". Back propagation requires that the activation function used by the artificial neurons (or "nodes") is differentiable. A generalized sigmoid function equation (16) has been chosen in order to accommodate any non-linearity during the modeling process

$$F(x) = \frac{\alpha}{1+e^{\beta x+\zeta}} \tag{16}$$

Where $x$the input is applied to the node and $\{\alpha, \beta, \zeta\}$ are the node parameters. Further, input to a succeeding layer (N) can be given as a linear combination of all outputs of neurons belonging to the preceding layer(M). The linear combination is based on the weighted connections between the respective neurons

$$input_{N_{pi}} = \sum_{i=1}^{K} w_{M_i N_p}.(output_{M_i}) \tag{17}$$

For learning phase, the algorithm changes the weights till a specified number of epochs or a get a zero error free. Where the new weight is adjusted according to the formula

$$w_{M_i N_p}^{new} = w_{M_i N_p}^{old} + \eta . err_{N_p} F`(input_{N_p}) \tag{18}$$

Where $F`$ is the first order derivative of equation (16) , η is the is the learning constant The learning rate is an important parameter in the learning process. It always lies between 0 to 1. In a rough neural network the training goes in two tiers, i.e., two parallel processes run through the network during parameter approximation, one through the lower neuron and the other through the upper neuron [20]. The learning rates of individual neurons are different and are generally time varying and decreases with the number of iterations. The learning rate of the lower neuron is expected to be more than the upper neuron as it is having significant information, which helps in discerning the patterns. Due to the properties of sigmoid function , thus

$$w_{M_i N_p}^{new} = w_{M_i N_p}^{old} + \eta . \left(d_{N_p} - y_{N_p}\right). input_{N_p} . y_{N_p} .(1\text{-}y_{N_p}) \tag{19}$$

Where y_{N_p} is the actual output and d_{N_p} is the desired output of the neuron p.

The neural network space N is defined as a mapping that transforms a neural network $Si \in N$ to a neural network $Sj \in N$. The main thing to be done is to find the connections between the way in which we make reduction and characteristics of the network that is constructed after reduction. The weights of the connections among input unit (rough or conventional neuron) are denoted by $w_i^h, i = 1,2,\ldots,n, h = 1,2,\ldots H$ and the weights of the connections between the hidden units and the output units by $w_h^0, h = 1,2,\ldots H, o = 1,2,\ldots,O$ where H is the number of hidden unit, n is the dimensionality of the input pattern and O is the number of output units.

Sine the output of the network is given by [4, 8, 20]

$$Y_0 = f(\textstyle\sum_{h=1}^{H} w_h^o\, f(\sum_{i=1}^{n} xiw_i^h)) \tag{20}$$

Where ***f*** is a sigmoid function. Since our goal is to find and eliminate as many unneeded network neurons as possible, it is important to identify the effect of violating connections of neurons to the output of the network.

Let Y_0 be considered as a function of single variable corresponding to the connection between the ***i***th input unit and the ***h***th hidden unit. The derivative of Y_0 with respect to the weights of the network is as follows

$$\frac{\partial Y_0}{\partial w_i^h} = Y_0 \times (1-Y_0) \times w_h^0 \times x_i \times \left(1 - f\left(\textstyle\sum_{i=1}^{n} x_i w_i^h\right)\right) f(\textstyle\sum_{i=1}^{n} x_i\, w_i^h) \tag{21}$$

$$\frac{\partial Y_0}{\partial w_h^0} = Y_0 \times (1-Y_0) \times f(\textstyle\sum_{i=1}^{n} x_i\, w_i^h) \tag{22}$$

By the Mean Value Theorem, let Y_0 be a function of single variable corresponding to the connection between the ***i***th input unit and the ***h***th hidden unit, thus;

$$Y_{0(w)} = Y_0\left(w_i^h\right) + \frac{\partial Y_0(w_i^h + \delta(w - w_i^h))}{\partial w_i^h} \times (w - w_i^h) \tag{23}$$

Where $0 < \delta < 1$. At w equals zero, thus

$$\left|Y_0(0) - Y_0(w_i^h)\right| \le \left|w_i^h \times \frac{\partial Y_0(w_i^h)}{\partial w_i^h}\right| \tag{24}$$

Consider the activation function is sigmoid function that belong to the interval [-0.5,0.5], thus

$$\left|\frac{\partial Y_i(w)}{\partial w_i^h}\right| \le \frac{w_h^0 x_i}{8} \tag{25}$$

From equation (24) and (25) if follow that

$$\left|Y_0(0) - Y_0(w_i^h)\right| \le {\left|w_i^h w_h^0 x_i\right|}/{8} \tag{26}$$

Inequality (26) illustrates an upper bound on the changes in the output of the network when the weight w_i^h is eliminated. Since$x_i \in [0,1]$, hence

$$\left|Y_0(0) - Y_0(w_i^h)\right| \le {\left|w_i^h w_h^0\right|}/{8} \tag{27}$$

Similarly, by considering Y_0 is a function of a single variable υ that corresponds to the connection between the h^{th} hidden unit and the o^{th} output unit, Hence

$$\left|\frac{\partial Y_i(v)}{\partial w_h^o}\right| \le 1/8 \tag{28}$$

Thus, the changes in the output of the network after the weight w_h^o is eliminated is bounded by

$$|Y_i(0) - Y_i(w_h^0)| \le {|w_h^o|}/{8} \tag{29}$$

Equations (27) and (29) show the maximum error that occurred in the network if a connection is eliminated from any layer in the model. Hence, removing the neuron and its connections that do not exist in the reduct set if the error, calculated by equation (27) and (29) are less than some threshold. When performing network learning is done in each iteration, a structure adaptation is consequently performed. The learning process finished whenever no significantly better results of classification with this network.

IV. Testing Handwriting Recognition

This section is aimed at providing an insight into results from the early section. It may also be regarded as space used to provide a reasonable justification for certain trends. This system has to phases, learning phase and test phase. In learning phase two stages are applied; the first stage is applied on a typewritten data set, the other is applied on the handwritten data set. First, the Arabic/Farsi typewritten instance digits are clustered in accordance to eleven features as demonstrated early in Figure (2) and can be represented by data attribute value Table (1)

Table 1: Data Attribute Table For Arabic/Farsi Digits

	a	b	c	d	e	f	g	h	i	j	k
٠	0	0	0	0	0	0	0	0	0	0	1
١	1	1	0	0	0	0	0	0	0	0	0
٢	1	1	0	0	0	0	1	1	0	0	0
٣	1	1	0	0	0	1	0	0	1	1	1
٤	1	1	1	0	0	0	0	1	1	0	0
٥	1	1	1	1	1	1	1	1	0	0	0
٦	0	0	0	0	1	1	1	1	0	0	0
٧	1	1	1	1	1	1	0	0	0	0	0
٨	1	1	0	0	1	1	1	1	0	0	0
٩	1	0	0	0	1	1	1	1	1	0	1

Using rough set discernibility matrix, these attributes are reduced in which $\{e\}$ is the most significant attribute associated with the reduct set to be $\{a, c, e, h, k\}$, where $\gamma_a = \gamma_c = \gamma_e = 0.4, \gamma_e = 0.6\ and\ \gamma_k = 0.2$. By Dissimilarity analysis of rough set [10], differences among patterns and the optimal structure of NN are discovered, as illustrated in Figure 4. The number of nodes in input layer are chosen in accordance with the redact set. The pattern of zero are differs than three pattern in one attribute is called a, while as zero pattern is characterized by only one attribute called k. Thus, the three pattern is characterized by the attributed $\{a, k\}$.

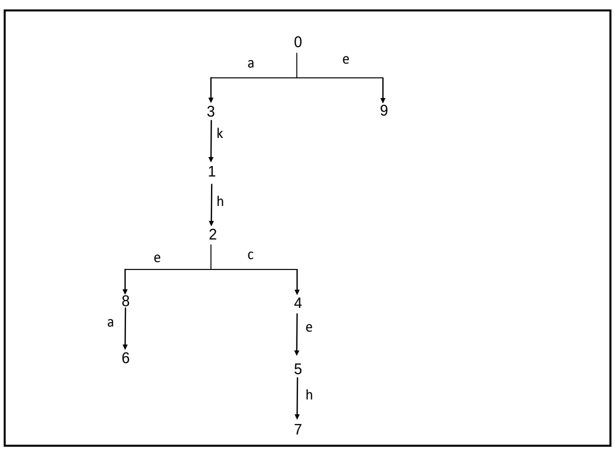


Fig. 4. The Dis-Similarity Diagram Among Different Arabic/Farsi Digits

Moreover, the center of gravity, calculated by equation (11), for typewritten data set represents v_k, center of the cluster. Using this reduction algorithm on a typewritten data reduces the number of learning epochs as a result of reducing number of input neurons as shown on Figure 5 which compare number of iterations before reduction and after it.

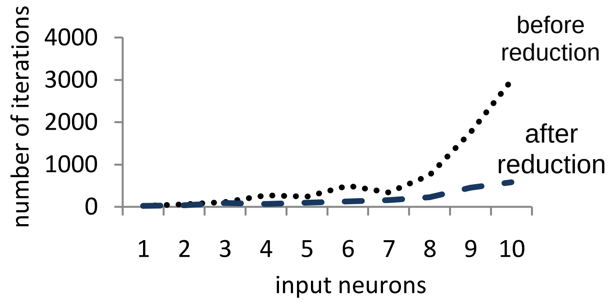


Fig. 5. Comparing Number Of Iterations Using Neural Network With Number Of Iterations After Applying Rough Sets

To evaluate the proposed hybrid model on isolated Arabic/Farsi handwritten characters, IFHCBD which is created at Amirkabir University of Technology has been used. The RS-RNN model is applied on Arabic/Farsi handwritten digits. IFHCDB contains a set of images for Arabic/Farsi digits that are divided into 88 training data and 30 test data. First, rough sets are used in clustering the handwritten data set, where the noise are discovered and removed by equation (10). Second, dissimilarity in data analysis is applied to compute the perturbation in the input data, by equation (15). Hence rough neurons are discovered and the optimal architecture is represented by rough sets' clusters. Finally, an implementation, as shown in Figure (6), for the Back propagation algorithm is performed where a structure adaptation is done by each learning epochs. The learning process is finished whenever the error function is at most 0.01.

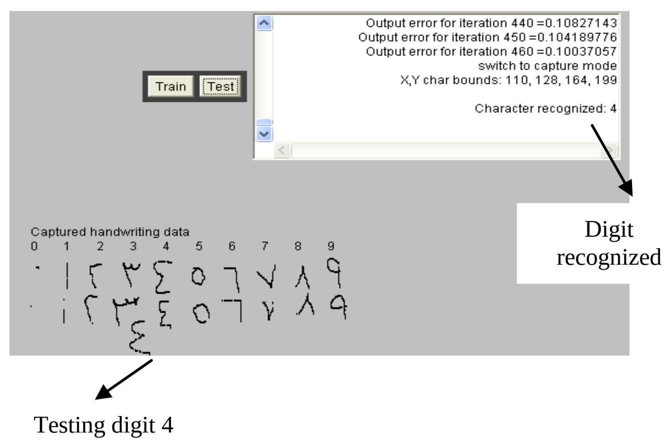


Fig. 6. The Implementation Of The RS-RNN

At the end of the study, we prove the ability of our System depending on the results for the recognition accuracy. Comparing errors for each digit pattern that are resulted from training by neural network against those are resulted from RS-RNN, as demonstrated in Figure (7).

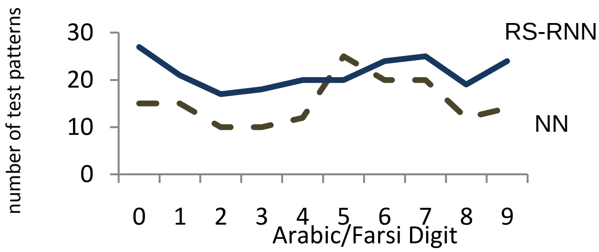


Fig. 7. Number Of Recognized Patterns By NN And RS-RNN

Moreover, a comparison among theories [15], such as genetic algorithms, simple object modeling, statistical method and rough sets with neural networks, and RS-RNN in terms of recognition accuracy is demonstrated in Figure(8). As mentioned RS-RNN approach is able to recognize the Arabic/Farsi handwritten digits more efficient than others.

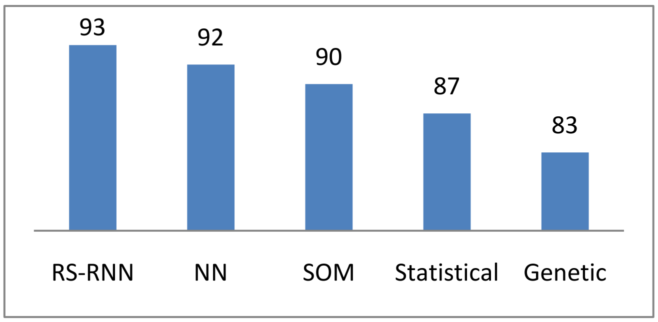


Fig. 8. the accuracy measure among different theoris and RS-RNN

V. Conclusions

Handwriting number recognition, a system for recognizing isolated digits as standard characters, is a challenging problem because different users have their own handwriting style. Moreover, it is affected by noise that established during acquisition and normalization. The main goal to this paper is to recognizing isolated Arabic/Farsi digits exist in different forms.

This paper presents a hybrid model that starts with acquiring and normalizing an image containing Arabic/Farsi digits. The digitized image was treated by rough sets. Rough sets played an important role in reducing the feature attribute, reduct, and discovering dissimilarity among different patterns. Also, rough sets segment the user pattern into different clusters in accordance with the Arabic/Farsi digit pattern. Moreover, the noise has been eliminated and the dissimilarity between the user cluster and its corresponding feature was measured.

By this paper, the optimal architecture of the rough neural network was discovered. Finally an adaptation of the rough neuron was applied during the learning phase using the back propagation algorithm.

The results were tested on standard data and proved the efficiency of our method. This approach efficiently chooses a segmentation method to fit our demands. Our approach successfully design and implement rough neural network which go without demands. After that RS-RNN are able to understand the Arabic/Farsi numbers that was manually written by the users.

References

[1] Duo Chen, Du-Wu Cui, Chao-Xue Wang, Zhu-Rong Wang," A Rough Set- Based Hierarchical Clustering Algorithm for Categorical Data", International Journal of Information Technology, vol. 12, no. 3, pp. 149-159, 2006.

[2] Elsayed Radwan, Omaima Nomir, and Eiichiro Tazaki, "A New Learning Method for Cellular Neural Networks Templates based on Hybrid of Rough Sets and Genetic Algorithms, " International Journal of Computer Science and Information Security, vol. 8, no. 3, pp. 155-164, August 2010.

[3] Georg Peters, Richard Weber, Rene Nowatzke, "Dynamic Rough Clustering and its Applications", Applied Soft Computing, vol. 12, pp. 3193-3207, 2012.

[4] Insung Jung, and Gi-Nam Wang," Pattern Classification of Back-Propagation Algorithm Using Exclusive Connecting Network", International Journal of Electrical and Electronics Engineering, vol. 2, no. 6, pp. 365-369, 2008.

[5] Lingras, P. J.," Rough Neural Networks", Proceedings of Sixth International Conference on Information Processing and Management of Uncertainty in Knowledge Based Systems, Granda, Spain, pp. 1445-1450, 1996.

[6] Luiz S. Oliveira and Robert Sabourin, "Support Vector Machines for Handwritten Numerical String Recognition", Proceedings of the 9th Int'l Workshop on Frontiers in Handwriting Recognition (IWFHR-9 2004).

[7] M.M. Richter: Similarity, Studies in Computational Intelligence (SCI), 73, 25–90, Springer-Verlag Berlin Heidelberg 2008

[8] Mark Watson, "Intelligent Java Applications for Internet and Intranets", Morgan Kaufmann Publisher,Sanfrancisco,California, chapter 4,10 ,1997.

[9] Nalini K. Ratha , Venu Govindaraju, "Advances in Biometrics -Sensors, Algorithms and Systems", Springer-Verlag London Limited, 2008.

[10] Pawlak, Z., " Rough Sets- Theoretical Aspects of Reasoning about Data", Kluwer Academic, Dordrecht, 1991.

[11] Piotr Zerzucha, and Beata Walczak," Concept of Dis-Similarity in Data Analysis", Trends in Analytical Chemistry, vol. 38, pp. 116-128, 2012.

[12] Polkowski, L., Skowron, A.(Eds), Rough Sets in Knowledge Discovery, Physica- Verlag, Heidelberg, vols: 1 and 2, 1998.

[13] Roman, W. Swiniarski, Andrzej Skowron," Rough Set Methods in Feature Selection and Recognition", Pattern Recognition Letters, vol. 24, pp. 833-849, 2003.

[14] S. Mozaffari, K. Faez, M. Zairatban, and S. M. Golzan, " A comprehensive Isolated Farsi/Arabic Chracter DataBase for Hanwritten OCR Research" 10th International WorkShop on Frontiers in Handwriting Recognition (IWFHR), La Baule, France, pp. 385-389, 2006.

[15] Saleh Ali K. Al-Omari, Putra Sumari, Sadik A. Al-Taweel and Anas J.A. Husain, " Digital Recognition using Neural Network", Journal of Computer Science, vol. 55, no. 6, pp. 427-434, 2009.

[16] Sandeep Chandana, Rene V. Mayorga," Rough Neuron based on Space Partitioning", Neurocomputing, vol. 74, pp. 2052-2061, 2011.

[17] Sheela Ramanna," Near Tolerance Rough Sets", Applied Mathematical Sciences, vol. 5, no. 38, 1895 – 1902, 2011.

[18] Shinya imai, che-wei lin, junzo watada and gwo-hshiung tzeng, "Rough Sets Approach to Human Resource Development of Information Technology Corporations", IJSSST, vol. 9, no. 2, pp. 3142, 2008.

[19] Yang Mingqiang, Kpalma Kidiyo and Ronsin Joseph, "A Survey of Shape Feature Extraction Techniques," Pattern Recognition Techniques, Technology and Applications, Peng-Yeng Yin (Ed.),pp. 43-90, 2008.

[20] Yassar Hassan, Eiichiro Tazaki, Sina Egawa, Kazuho Suyama," Rough Neural Classifier System", IEEE International Conference on Systems, Man and Cybernetics, 2002.

AUTHORS PROFILE

Elsayed Radwan, received his B.Sc. and M.Sc. from Mathematics Dept. Faculty of Science, Mansoura University, Egypt. He received his Ph.D. from Intelligent Lab., Graduate School of Engineering, Toin University of Yokohama, Japan. He became a Lecturer at Computer Science Dept., Faculty of Computer and Information Sciences, Mansoura University, Egypt on 2007. His areas of interest are Soft Computing, Evolutionary Computing, Pattern Recognition, and Cellular Neural Networks. He is now working as an Assistant Professor of Computer Science in Institute of Scientific Research and Revival of Islamic Heritage.

Voice Recognition Method with Mouth Movement Videos Based on Forward and Backward Optical Flow

Kohei Arai [1]
Graduate School of Science and Engineering
Saga University
Saga City, Japan

***Abstract*—Lip reading method with mouth movement videos based on backward optical flow is proposed. Through experiments with 10 of mouth movement videos, it is found that the proposed lip reading method is superior to the conventional optical flow based method.**

Keywords- lip reading; optical flow; Hiddeen Markov Model; mouth movement

I. Introduction

Although voice recognition is now world widely available, recognition performance is not good enough for normal conversations. For instance, voice recognition performance of the typical Hidden Markov Model: HMM based method [1] (this is referred to the conventional voice recognition hereafter) with the feature of Formant is less than 50 % when the signal to noise ratio is below 5dB. In other words, voice recognition performance is totally affected by noise. In normal conversation among us, not only voice but also mouth movement is used for recognitions. Mouth movement video analysis makes voice recognition much better performance. The proposed lip reading method is for improvement of voice recognition performance.

Usually, Hidden Markov Model based method or neural network based method is used for voice recognitions as well as optical flow [2]-[9] based analysis of the mouth movement videos. Forward direction (from the past to the future) of optical flow is usually used for mouth movement analysis. Voice recognition performance can be improved by adding backward direction (from the future to the past) of optical flow for correction of voice recognition errors through a confirmation of recognized results. In this process, two voice elements are treated as a unit for the proposed backward optical flow. The conventional forward direction of optical flow recognizes by voice element by voice element, though. In order to make sure the recognized results, two voice elements are much easier and efficient manner. This is because transient between voice element and voice element is so important for voice recognitions. This is the basic idea of the proposed lip reading method.

Experiments are conducted with 10 of mouth movement videos which are acquired by different peoples. Voice recognition performance, then is evaluated and is compared to the conventional forward direction of optical flow based method. The experimental results show that the proposed backward optical flow is superior to the conventional method.

The following section describes the proposed method followed by some experiments. Then conclusion is described together with some discussions.

II. Proposed Method

A. Overview of the Proposed Voice Recognitions

Process flow of the proposed voice recognition method is shown in Figure 1.

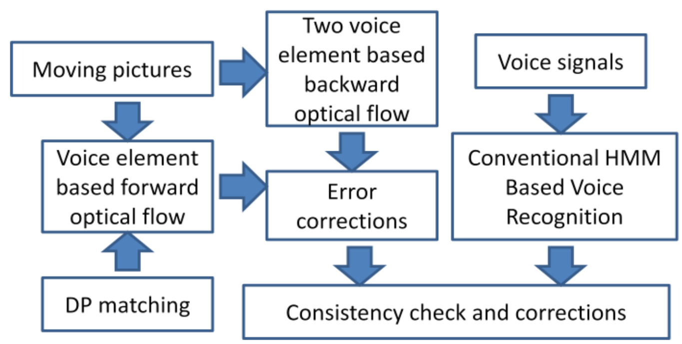


Fig. 1. Process Flow Of The Proposed Voice Recognition

Time series of moving pictures and voice signals are acquired first. Using the conventional HMM based voice recognition method, time series of voices are recognized. This is referred to voice based recognition, hereafter.

On the other hands, lip reading is performed based on forward optical flow with time series of moving pictures of mouth movement which are acquired at the same time of voice signals. This is done by voice element by voice element as usual. Meanwhile, two voice element based backward optical flow is applied to the time series of moving pictures of mouth movement. Then the result from the voice element based forward optical flow is corrected by using the two element based backward optical flow results. Through this voice element based optical flows, Dynamic Programming: DP matching based recognition is performed. Because extracted voice elements have missing portion of elements. Furthermore, recognition needs some insertions of voice elements. DP matching allows insertion and also recognition without some missing elements. This is referred to moving picture based recognitions, hereafter.

After all, the recognized results from moving picture based and voice signal based methods are compare and check a consistency between both results, then final recognition results are reduced.

B. Optical Flow

Optical flow is defined as object movement representations in vector form in the visual representations. From moving pictures, videos of digital images, optical flow can be extracted as vectors. There are the conventional block matching method and gradient method for extraction of optical flow. Block matching method is usually referred to “Block-based methods” which are minimizing sum of squared differences or sum of absolute differences, or maximizing normalized cross-correlation while the gradient method is used to be referred to “Differential methods” which are based on partial derivatives of the image signal and/or the sought flow field and higher-order partial derivatives. Other than these, there are “Phase correlation methods” which can get inversion of normalized cross power spectrum between two adjacent images and “Discrete optimization methods” of which the search space is quantized, and then image matching is addressed through label assignment at every pixel.

C. Input Data for Dynamic Programming: DP Matching

Figure 2 shows an example of one cut of the moving picture of mouth movements. Time series of images are acquired. Voice element can be extracted from the time series of images. From the piece of the time series of images, four feature points, top, bottom, right end, and left end are extracted as input data for DP matching.

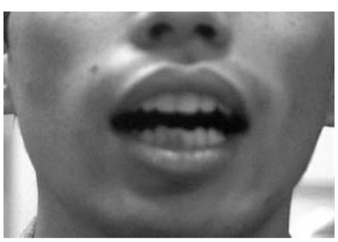


Fig. 2. Example of a piece of moving picture of time series of images of mouth movements

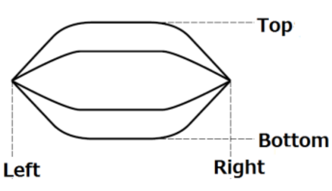


Fig. 3. Feature points as input data for DP matching

D. Fundamentals of Dynamic Programming: DP Matching

Similarity $D(A,B)$ between coded edge [A] and [B] is defined as follows,

$$D(A,B) = \min_{c[k]} \frac{\sum_{k=0}^{K-1} w[k]d[k]}{\sum_{k=0}^{K-1} w[k]} \tag{1}$$

where

$$\begin{aligned} A &= \{a[i](i = 0,\ldots,I-1)\} \\ B &= \{b[j](j = 0,1,\ldots,J-1)\} \end{aligned} \tag{2}$$

and d[k] denote distance, as well as $w[k]$ denotes weighting coefficient,

$$w[k] = i_k - i_{k-1} + j_k - j_{k-1} \tag{3}$$

when the coded edge (K denotes the total number of edges) is represented as shown in Figure.4.

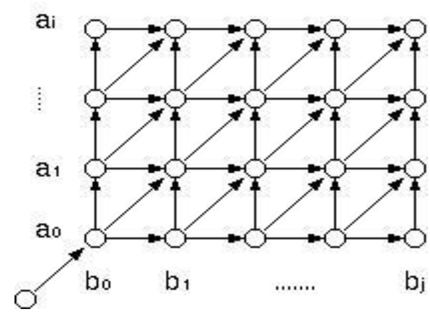


Fig. 4. Coded edge information

Subset summation of $s[c[m]]$ of numerator of equation (1) is expressed with equation (4) when k=m,

$$\begin{aligned} s[c[m]] = s[i_m, j_m] &= \min_{c[m]} \sum_{k=0}^{m} w[k]d[k] \\ &= \min(\min_{c[m-1]} \sum_{k=0}^{m-1} w[k]d[k] + w[m]d[m]) \\ &= \min(s[c[m-1]] + w[m]d[m]) \end{aligned} \tag{4}$$

If it is assumed that s is increased, then $s[c[m]]$ at k=m-1 is represented with equation (5),

$$s(i_m, j_n) = \min \begin{cases} s(i_m, j_{n-1}) + d(i_m, j_n) \\ s(i_{m-1}, j_{n-1}) + 2d(i_m, j_n) \\ s(i_{m-1}, j_n) + d(i_m, j_n) \end{cases} \tag{5}$$

Because a, b positions are at the one of (i_{m-1}, j_m), (i_{m-1}, j_{m-1}), (i_m, j_{m-1}) in Fig.1. Thus total summation of s and D can be calculated if the summation of $s[c[m]]$ is reached at (i_{K-1}, j_{K-1}).

Even if some of the coded edges are missing, similarity between two coded edges can be calculated results in edge image matching between the query image and the current image.

E. Detailes of Dynamic Programming: DP Matching

Initial condition is assumed to be $(x_0, x_0^{(l)})$, then $g_1^{(l)}(1,1)$ is minimum distance for $x_1 - x_1^{(l)}$ where x_i is input pattern data of voice elements while $x_i^{(l)}$ is reference voice elements. Then suffix of the input pattern data is incremented as follows,

$$g_j^{(l)}(i',i) = Min \left\{ \begin{array}{l} g_{j-1}^{(l)}(i'-1,i) + w_j d_j(x_{i'}, x_i^{(l)}) \\ g_{j-1}^{(l)}(i'-1,i-1) + w_j d_j(x_{i'}, x_i^{(l)}) \\ g_{j-1}^{(l)}(i',i-1) + w_j d_j(x_{i'}, x_i^{(l)}) \end{array} \right.$$

There are three possible solutions which minimize the distance between input pattern data and the reference pattern data.

Meanwhile, $(x_0, x_0^{(l)})$ is defined as inner product (dot product) of the

$$(x, x^{(l)}) = ||x|| \cdot ||x^{(l)}|| \cos\theta$$

Then distance between two x_i and $x_i^{(l)}$ are as follows,

$$d_s = \cos\theta = \frac{(x, x^{(l)})}{||x|| \cdot ||x^{(l)}||}$$

Where $-1 \leq \cos\theta \leq 1$

To find the minimum distance, if the d_s is minimum when the $l=l_0$, then the input pattern data is classified to l_0. If a distortion is considered for the input pattern data due to some reasons, then d_s is no longer can be calculated with $(x_0, x_0^{(l)})$. The reason for that is some of the voice elements will be missing, or some of voice element inserted accidentally as shown in Figure 5. Therefore, distorted input pattern data (Modified pattern) has to be represented as follows,

$$x = (x_1, x_2, L, x_{i'}, L, x_{I'})^T$$

Reference patter in Figure 5 is defined as reference patter for voice elements. In this case, the following function which represents the relation between d_s and $(x_0, x_0^{(l)})$.

$$F = c(1)c(2)Lc(k)Lc(K) \qquad (k = 1,2,L,K)$$

where

$$c(k) = (i'(k), i(k))$$

This is the k-th relation between $(x_0, x_0^{(l)})$

Then the distance is rewrite with the following equation,

$$d_s^{(l)}(x) = Min \left\{ \frac{\sum_{j=1}^{J} w_j d_j(x_{i'}, x_i^{(l)})}{\sum_{j=1}^{J} w_j} \right\}$$
$$(i' = 1,2,\cdots,I')(i = 1,2,\cdots,I)$$

Where wj denotes k-th weighting coefficient which allows adjustment, or normalization of the distance d_s from -1 to 1. Figure 6 shows an enlarged portion of Figure 5. Weighting coefficients can be determined as shown in Figure 6.

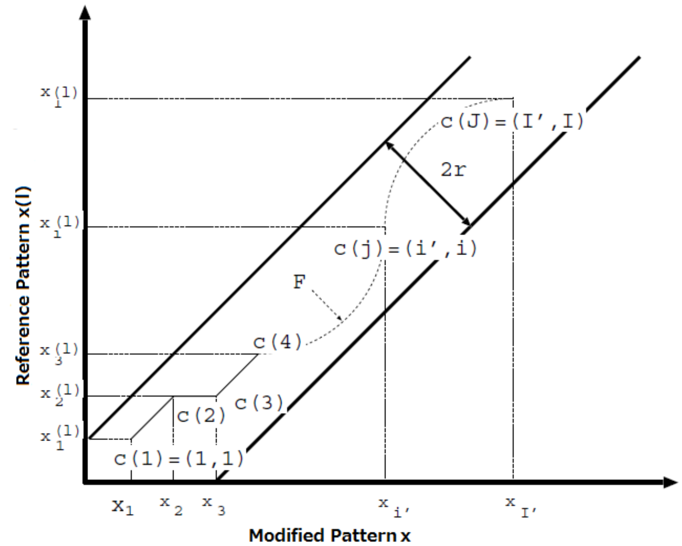


Fig. 5. Relation between reference pattern and input pattern data (Modified Pattern)

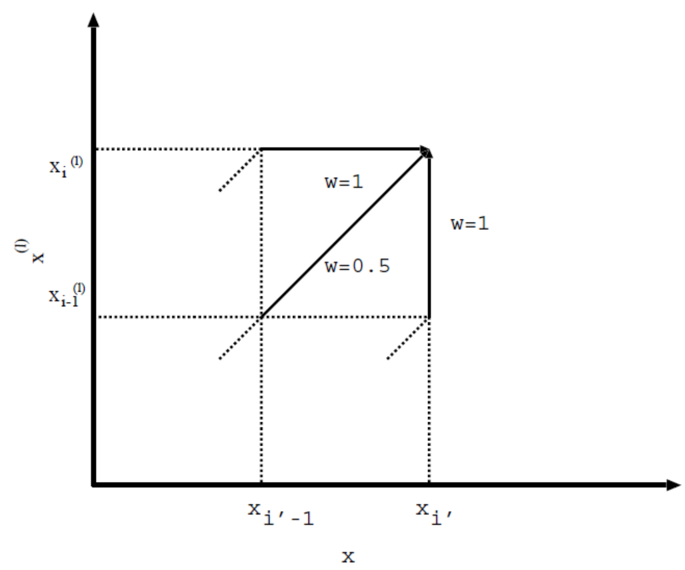


Fig. 6. Enlarged portion of Figure 5

There are some conditions for the distance definition,

Start and end of input pattern and reference pattern are corresponded,

The voice element orders have to be same for both input and reference patterns, The corresponding reference pattern exists near by the input pattern.

Then, as shown in Figure 6, F is calculated as follows,

$$F = \begin{cases} x_{i'-1} - x_i^{(l)} \ and \ x_{i'} - x_i^{(l)} \\ x_{i'-1} - x_{i-1}^{(l)} \ and \ x_{i'} - x_i^{(l)} \\ x_{i'} - x_{i-1}^{(l)} \ and \ x_{i'} - x_i^{(l)} \end{cases} \quad w_j = \begin{cases} -1 \\ 0.5 \\ 1 \end{cases}$$

Then the distance between input pattern data of voice element and the reference voice element pattern is represented as follows,

$$d^{(l)}(x) = \frac{1}{I+I'} Min \left\{ \sum_{j=1}^{J} w_j d_j(x_{i'}, x_i^{(l)}) \right\}$$

Where I and I' denotes the number of reference voice element patterns, respectively. Thus input voice element pattern is classified to the reference pattern, namely, if the d_s is minimum when the $l=l_0$, then the input pattern data is classified to l_0.

F. Voice Elements

In this paper, Japanese language recognition is focused. Japanese, in particular, the following 40 voice sounds are concerned.

Vowel: "a, i, u, e, o"

Consonant + Vowel: "ka, ki, ku, ke, ko, sa, si, su, se, so, ta, ti, tu, te, to, na, ni, nu, ne, no, ha, hi, hu, he, ho, ma, mi, mu, me, mo, ya, yu, yo,ra, ri, ru, re, ro, wa, and nn"

Voice element is defined as vowel and consonant, separately. Therefore, "a, i, u, e, o, k, s, t, n, h, m, y, r, w" are major concern. In this paper, voice recognition for these 14 vowels and consonants are concentrated.

III. EXPERIMENTS

First, the reference patters of the aforementioned 40 voice sounds are prepared with four different speakers. Sounds and moving pictures are prepared as the reference patterns.

For the optical flow based voice recognition, moving vectors of the aforementioned four features, top, bottom, left end, and right ends of mouth which are extracted from the moving pictures are used. Features are represented as the symbol. One small example of a portion of the time series of symbolized voice elements are shown in Figure 7.

In accordance with the distance, the first (L1), the second (L2), and the third (L3) candidates are determined. From the calculated distance, likelihood, or probability is also calculated for each candidate. The probability is calculated by voice element by voice element and also is evaluated for both vowels and consonants. The proposed method is based on forward and backward optical flow as explained in the second section. The probability evaluations have been done for the proposed method and compared to forward optical flow based method as well as the conventional voice recognition method.

Probability or likelihood is corresponding to the percent correct classification: PCC. If the PCC is evaluated with the first candidate only, then PCC for the conventional voice recognition method is not so good, below 43% for vowels and 14.3% for consonant + vowel while that for the proposed method with forward optical flow is 71.4% for vowel and 57.1% for consonant + vowel. Therefore, it is found that PCC is improved remarkably by taking moving picture analysis with the forward optical flow into consideration by the factor of approximately 30%.

$$\begin{array}{llll} 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 4 & -4 & 4 & 3 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 \\ \#a & & & \\ 4 & -2 & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 \\ \#ra & & & \end{array}$$

Fig. 7. Symbolized voice elements for "a", and "ra"

TABLE I. PROBABILITY EVALUATION FOR THE FIRST TO THIRD CANDIDATES FOR THE PROPOSED AND THE METHOD WITH FORWARD OPTICAL FLOW ONLY AS WELL AS THE CONVENTIONAL VOICE RECOGNITION METHOD METHOD

		L1	L2	L3
Conventional	Vowel	42.9	71.4	71.4
	Consonant + Vowel	14.3	42.9	42.9
Forward	Vowel	71.4	85.7	100
optical flow	Consonant + Vowel	57.1	57.1	100
Forward	Vowel	90.3	95.5	100
and Backward	Consonant + Vowel	77.4	82.7	100

Furthermore, the proposed method with backward optical flow for confirmation and correction of recognized results which are obtained from the proposed method with forward optical flow only is superior to the proposed method with forward optical flow only. This implies that PCC is improved remarkably by taking confirmation and correction of recognized results which are obtained from the proposed method with forward optical flow only into account by the factor of about 20%.

PCC of vowel is always better than that of consonant + vowel, obviously. In particular for the conventional voice recognition method, there is around 30# of difference between vowel PCC and PCC of vowel + consonant.

If PCC is evaluated with the first to the third candidates, both of the proposed method with forward optical flow only and that with forward and backward optical flow shows 100% of PCC. This implies that the effect of considering not only voice signals but also moving pictures on PCC of voice recognition is significant

As the results, it is found that the voice recognition performance can be improved by adding moving picture analysis to the voice signal analysis. This is same thing for human to human conversations. By looking at the speakers mouth movement, voice recognition is helped and reconfirmed recognized results at the same time.

ACKNOWLEDGMENT

The author would like to thank Mr. Shinji Matsuda for his effort to conduct the experiments.

REFERENCES

[1] Hongbing Hu, Stephen A. Zahorian, (2010) "Dimensionality Reduction Methods for HMM Phonetic Recognition," ICASSP 2010, Dallas, TX, http://bingweb.binghamton.edu/~hhu1/paper/Hu2010Dimensionality.pdf .(accessed on September 14 2012)

[2] Huston SJ, Krapp HG (2008). Kurtz, Rafael. ed. "Visuomotor Transformation in the Fly Gaze Stabilization System". *PLoS Biology* 6 (7): e173. doi:10.1371/journal.pbio.0060173. PMC 2475543. PMID 18651791. http://www.plosbiology.org/article/info:doi/10.1371/journal.pbio.0060173.(accessed on September 14 2012)

[3] Andrew Burton and John Radford (1978). *Thinking in Perspective: Critical Essays in the Study of Thought Processes*. Routledge. ISBN 0-416-85840-6. http://books.google.com/?id=CSgOAAAAQAAJ&pg=PA77&dq=%22optical+flow%22+%22optic+flow%22+date:0-1985.(accessed on September 14 2012)

[4] David H. Warren and Edward R. Strelow (1985). *Electronic Spatial Sensing for the Blind: Contributions from Perception*. Springer. ISBN 90-247-2689-1. http://books.google.com/?id=-I_Hazgqx8QC&pg=PA414&dq=%22optical+flow%22+%22optic+flow%22+date:0-1985.(accessed on September 14 2012)

[5] S. S. Beauchemin , J. L. Barron (1995). *The computation of optical flow*. ACM New York, USA http://portal.acm.org/ft_gateway.cfm?id=212141&type=pdf&coll=GUIDE&dl=GUIDE&CFID=72158298&CFTOKEN=85078203.(accessed on September 14 2012)

[6] David J. Fleet and Yair Weiss (2006). "Optical Flow Estimation". In Paragios et al.. *Handbook of Mathematical Models in Computer Vision*. Springer. ISBN 0-387-26371-3. http://www.cs.toronto.edu/~fleet/research/Papers/flowChapter05.pdf.(accessed on September 14 2012)

[7] John L. Barron, David J. Fleet, and Steven Beauchemin (1994). "Performance of optical flow techniques". *International Journal of Computer Vision* (Springer). http://www.cs.toronto.edu/~fleet/research/Papers/ijcv-94.pdf.(accessed on September 14 2012)

[8] B. Glocker, N. Komodakis, G. Tziritas, N. Navab & N. Paragios (2008). *Dense Image Registration through MRFs and Efficient Linear Programming*. Medical Image Analysis Journal. http://vision.mas.ecp.fr/pub/mian08.pdf.(accessed on September 14 2012)

[9] Christopher M. Brown (1987). *Advances in Computer Vision*. Lawrence Erlbaum Associates. ISBN 0-89859-648-3. http://books.google.com/?id=c97huisjZYYC&pg=PA133&dq=%22optic+flow%22++%22optical+flow%22. (accessed on September 14 2012)

AUTHORS PROFILE

Kohei Arai, He received BS, MS and PhD degrees in 1972, 1974 and 1982, respectively. He was with The Institute for Industrial Science, and Technology of the University of Tokyo from 1974 to 1978 also was with National Space Development Agency of Japan (current JAXA) from 1979 to 1990. During from 1985 to 1987, he was with Canada Centre for Remote Sensing as a Post Doctoral Fellow of National Science and Engineering Research Council of Canada. He was appointed professor at Department of Information Science, Saga University in 1990. He was appointed councilor for the Aeronautics and Space related to the Technology Committee of the Ministry of Science and Technology during from 1998 to 2000. He was also appointed councilor of Saga University from 2002 and 2003 followed by an executive councilor of the Remote Sensing Society of Japan for 2003 to 2005. He is an adjunct professor of University of Arizona, USA since 1998. He also was appointed vice chairman of the Commission “A” of ICSU/COSPAR in 2008. He wrote 30 books and published 332 journal papers

AutoBeeConf

A swarm intelligence algorithm for MANET administration

Luca Caputo, Cristiano Davino, Filomena de Santis,Vincenzo Ferri
Dipartimento di Informatica
Facoltà di Scienze, Università degli Studi di Salerno
Fisciano (SA), Italy

***Abstract*— In a mobile ad-hoc network (MANET) nodes are self-organized without any infrastructure support: they move arbitrarily causing the network to experience quick and random topology changes, have to act as routers as well as forwarding nodes, some of them do not communicate directly with each other. Routing and IP address auto-configuration are among the most challenging tasks in the MANET domain. Swarm Intelligence is a property of natural and artificial systems involving minimally skilled individuals that exhibit a collective intelligent behavior derived from the interaction with each other by means of the environment. Colonies of ants and bees are the most prominent examples of swarm intelligence systems. Flexibility, robustness, and self-organization make swarm intelligence a successful design paradigm for difficult combinatorial optimization problems, such as routing and IP address allocation in MANET. This paper proposes *AutoBeeConf*, a new IP address auto-configuration algorithm based on a bee swarm labor that may be applied to large scale MANET with low complexity, low communication overhead, even address distribution, and low latency. Both the protocol description and the simulation experiments are presented to demonstrate the advantages of AutoBeeConf over two known algorithms, namely *Buddy* and *Antbased* protocols. Eventually, future research directions are established, especially toward the principle that swarm intelligence paradigms may be usefully employed in the redefinition or modifications of each layer in the TCP/IP suite in such a way that it can efficiently work even in the infrastructure-less and dynamic nature of MANET environment.**

***Keywords*—MANET; Routing protocols; IP address auto-configuration; Swarm intelligence.**

I. INTRODUCTION

Advances in wireless communication technology have strongly encouraged the use of low-cost and powerful wireless transceivers in mobile applications. As compared with wired networks, mobile networks exhibit unique features: recurrent network topology changes, link capacity fluctuations, critical bounds to their performances. Mobile networks can be classified into infrastructure networks and mobile ad-hoc networks, [1]. In an infrastructure mobile network, mobile nodes communicate through wired access points that work in the node transmission range and create the backbone of the network. In a mobile ad-hoc network (MANET) nodes, acting potentially both as routers and hosts, are generally equipped with either omnidirectional or directional antennas for sending and receiving information. They have a packet-forwarding capability in order to communicate via shared and limited radio channels. Communication may be performed by one-to-one transmissions (single-hop) or using other nodes as relay stations (multi-hop). In both cases each sender node must adjust its emission power in order to reach the respective receiver node. In cases where energy is supplied by batteries, the network lifetime is limited by the batteries of the wireless devices. Therefore, energy saving is critical in all network operations. Ad-hoc networks are suitable for situations where only temporary communication is needed, and establishing a communication infrastructure is either not possible or not desirable. As an example for an ad-hoc network, we can imagine a meeting in which the members want to interchange data. The participants do not want to make high efforts for the network configuration since; perhaps, the users are not technically skilled. Notwithstanding, users wish a convenient way for their cooperation.

A challenging task in the MANET domain is routing where a path between a source and its destination must be found, possibly in an efficient way. *Proactive* routing, *reactive* routing and *hybrid* routing [2] are the most popular classes of MANET routing protocols. In a *proactive* routing protocol nodes continuously evaluate routes towards all reachable nodes and maintain consistent, up-to-date routing information even though network topology changes occur (e.g. Destination Sequenced Distance Vector, DSDV, [3]) . In a *reactive* routing protocol, routing paths are searched only when needed by means of a route discovery operation established between the source and destination node (e.g. Dynamic Source Routing, DSR, [4]). *Hybrid* routing protocols combine the merits of both proactive and reactive protocols and overcome their shortcomings (e.g. Core Extraction Distributed Ad-Hoc Routing, CEDAR, [5]). However, before a path between the nodes can be found, the nodes must be identified according to an uniform address scheme and an unique address assignment policy in sight of an IP (Internet Protocol) correct operation [6]. The strong centralization of DHCP (Dynamic Host Configuration Protocol) and the local broadcast of IPv4 Link-Local Addresses are not suited for MANETs, where topology changes, network partitioning and merging cannot assure that every mobile node will be connected at a given time neither predict the topology or size of the network. Several approaches have been proposed to solve this problem, generally classified into categories reflecting the allocation features of protocols. *Stateful*, *stateless*, and *hybrid* approaches are the most popular classes of MANET address assignment protocols. For *statefull* approaches, the state of each address is held in such a way the network have a vision of assigned and non assigned IPs, so address duplication could be avoided. For *stateless* approaches, each node randomly chooses its own address and performs a duplicate address detection test to ensure that the chosen address is not already used. *Hybrid* approaches combine mechanisms from both stateful and stateless approaches, in order to improve reliability and scalability. The price is a more complex protocol.

Swarm Intelligence (SI) is an novel distributed paradigm for the solution of hard problems taking insight from biological examples such as colonies of ants, bees, and termites, schools of fish, flocks of birds, [7]. The most interesting property of SI is the involvement of multiple individuals that interact with each other and the environment, exhibit a collective intelligent behavior, and are able to solve complex problems. Many applications, mainly in the contexts of computer networks, distributed computing and robotics are nowadays being designed using SI, [8], [9]. The basic idea behind this paradigm is that many tasks can be more efficiently completed by using multiple simple autonomous agents instead of a single sophisticated one. Regardless of the improvement in performance, such systems are usually much more adaptive, scalable and robust than those based on a single, highly capable, agent. An artificial swarm can generally be defined as a decentralized group of autonomous agents having limited capabilities. Due to the adaptive and dynamic nature of MANETs, the swarm intelligence approach is considered a successful design paradigm to solve the routing and the IP address auto-configuration problems.

The rest of this paper is organized as follows. Section 2 briefly reviews references on the swarm paradigm, specifically based on ant and bee behaviors, with a glance at their use for the solution of the MANET routing problem. Section 3 first defines the IP address auto-configuration problem for ad-hoc networks, then describes two well kwon protocols, such as the *Buddy* protocol and the *AntConf* protocol, developed with a stateful approach based on the *binary split* idea of [10], and with a swarm intelligence based model targeted at network administration [11], respectively. Section 4 contains the description of the *AutoBeeConf* protocol, our proposal for the IP address auto-configuration for MANET that integrates the advantages deriving from the classical approaches with the benefits arising from the most typical activities of a bee swarm. Section 5 presents the simulations carried on to test and compare the performances of the three before mentioned protocols. Eventually, section 6, after reviewing the main features of AutoBeeConf, sketches potential future extensions to the work.

II. THE SWARM PARADIGM

Many ant species *(Argentine ant, Linepithema humile)* are able to discover the shortest path to a food source and to share that information with other ants through *stigmergy* [12]. In ant colonies, indeed, an odor substance, the pheromone, is used as an indirect communication medium. When a source of food is found, the ants lay some pheromone to mark the path. The quantity of the laid pheromone depends upon the distance, quantity and quality of the food source. While an isolated ant that moves at random detects a laid pheromone, it is very likely that it will decide to follow its path. This ant will itself lays a certain amount of pheromone, and hence enforces the pheromone trail of that specific path. Accordingly, the path that has been used by more ants will be more attractive to follow. The local intensity of the pheromone field, which is the overall result of the repeated and concurrent *path sampling* experiences of the ants, encodes a spatially distributed *measure of goodness* associated with each possible move. This form of distributed control, based on indirect communication among agents which locally modifies the environment and reacts to these modifications, is called *stigmergy*. These basic ingredients have been reverse-engineered in the framework of Ant Colony Optimization (ACO), which exploits the ant behavior to define a nature-inspired meta-heuristic for combinatorial optimization. ACO has been applied with success to a variety of combinatorial problems, such as traveling salesman, routing, scheduling, and has been shown to be an effective tool in finding good solutions.

Bee colonies (*Apis mellifera),* show structural characteristics similar to those of ant colonies, such as the presence of a population of minimalist social individuals, and must face analogous problems such as distributed foraging, nest building and maintenance. A honey bee colony consists of morphologically uniform individuals with different temporary specializations. The benefit of such an organization is an increased flexibility to adapt to the changing environments. Thousands of worker bees perform all the maintenance and management jobs in the hive. There are two types of worker bees, namely *scouts* and *foragers*. The scouts start from the hive in search of a food source randomly keeping on this exploration process until they are tired. When they return back to the hive, they convey to the foragers information about the odor of the food, its direction, and distance with respect to the hive by performing dances. A *round dance* indicates that the food source is nearby whereas waggle dances indicate that the food source is far away. Waggling is a form of dance in eight-shaped circular direction. It is repeated again and again; its intensity and direction gives information about the food source quality and location, respectively. The better is the quality of food; the greater is the number of foragers recruited for harvesting. In analogy with ACO, the Bee Colony Optimization (BCO) meta-heuristic has been defined and satisfactorily tested on many combinatorial problems [13].

While referring to the specialized literature for an exhaustive coverage of swarm-inspired algorithms, in the sequel we will limit our attention to a short description of a few routing algorithms, namely modeled on both ant and bee behaviors, which can help in appreciating equivalent solutions in the IP address auto-configuration domain.

A. AntNet and AdHocNet

The first ACO routing algorithm, *AntNet* [14], [15] was designed for wired packet-switched networks. It is a proactive algorithm where each node periodically sends a *forward ant* to a random destination. The forward ant records its path as well as the time needed to arrive at each intermediate node. The timing information recorded by the forward ant, which is forwarded with the same priority as data traffic, is returned from the destination to the source by means of a high priority *backward ant*. Each intermediate node updates its routing tables with the information from the backward ant. Routing tables contain per destination next hop biases so that faster routes are used with greater likelihood. The algorithm exhibits a number of interesting properties which are also desirable for MANET: it can work in a fully distributed way, is highly adaptive to network and traffic changes, uses mobile agents for active path sampling, is robust to agent failures, provides multipath routing, and automatically takes care of data load spreading. However, the fact that it crucially relies on repeated path sampling can cause significant overhead.

AntHocNet is a hybrid multipath algorithm *for* routing in mobile ad-hoc networks consisting of reactive and proactive components, [16], [17]. It does not maintain routes to all possible destinations at all times (like *AntNet*), but only sets up paths when they are needed at the start of a data session. This is done in a *reactive route setup* phase, where the *reactive forward ants* are launched by the source in order to find multiple paths to the destination, and the *backward ants* return to the source to set up the paths. According to the common practice in ACO algorithms, the paths are set up in the form of pheromone tables indicating their respective quality. After the route setup, data packets are routed stochastically over the different paths following these pheromone tables. While the data session is going on, new ants, the *proactive forward ants, monitor*, maintain and improve paths. This allows to adapt to changes in the network, and to construct a mesh of alternative paths between source and destination. The proactive behavior is supported by a lightweight information bootstrapping process. Link failures, detected by unicast transmissions or expected hello messages crashes, and are coped with either a local route repair or by warning preceding nodes on the paths.

Antnet and *AntHocNet* have been evaluated on the basis of a relatively large number of simulation experiments using a custom network simulator. The algorithms have been tested on a variety of different scenarios based on different topologies with a variable number of nodes, and considering UDP traffic patterns with different geographical and generation characteristics. The reported experiments show that they robustly outperform several different dynamic state-of-the-art algorithms in terms of throughput and delay.

B. BeeHive and BeeAdHoc

BeeHive is a proactive algorithm that models bee agents in packet switching networks for routing purposes, [18], [19]. Since in nature the majority of forager's exploits food sources nearby the hive whereas a minority visits food sites far away from it, the algorithm provides for two types of agents: short distance bees and long distance bees which collect and disseminate routing information in the neighborhood of their source and in the entire network, respectively. They differ in their life time that is the number of hops they can travel across. Nodes periodically send a bee agent, by broadcasting replicas of it to each neighbor. When a replica of a particular bee agent arrives at a site, it updates routing information before being flooded again. This process continues until the life time of the agent expires, or if a same replica had been received already at a site. Short and long distance bees allow to partition the network in *foraging zones* and *foraging regions* so that each node maintains current routing information to reach all nodes in its zone and only the address of a *region representative node* to reach nodes located outside its zone. The next hop for a data packet is selected in a probabilistic manner according to a quality measures assigned to the current node. As a result, not all packets follow "best" paths. This will help in maximizing the system performance though a data packet may not follow a best path, a concept directly borrowed from a principle of bee behavior: a bee could only maximize her colony profit if she refrains from broadly monitoring the dance floor to identify the single most desirable food.

BeeAdHoc is a reactive source routing algorithm based on the use of four different bee-inspired types of agents: *packers, scouts, foragers, and bee swarms*. [20], [21]. *Packers* mimic the task of a food-storekeeper bee, reside inside a network node, receive and store data packets from the upper transport layer. Their main task is to find a forager for the data packet at hand. Once the forager is found and the packet is handed over, the packer will be killed. *Scouts* discover new routes from their launching node to their destination node. A scout is broadcasted to all neighbors in range using an expanding time to live (TTL). At the start of the route search, a scout is generated; if after a certain amount of time the scout is not back with a route, a new scout is generated with a higher TTL in order to incrementally enlarge the search radius and increase the probability of reaching the searched destination. When a scout reaches the destination, it starts a backward journey on the same route that it has followed while moving forward toward the destination. Once the scout is back to its source node, it recruits foragers for its route by *dancing*. A dance is abstracted into the number of clones that could be made of the same scout. *Foragers* are bound to the bee hive of a node. They receive data packets from packers and deliver them to their destination in a source-routed modality. To attract data packets foragers use the same metaphor of a waggle dance as scouts do. Foragers are of two types: delay and lifetime. From the nodes they visit, delay foragers gather end-to-end delay information, while lifetime foragers gather information about the remaining battery power. Delay foragers try to route packets along a minimum-delay path, while lifetime foragers try to route packets in such a way that the lifetime of the network is maximized. A forager is transmitted from node to node using a unicast, point-to-point modality. Once a forager reaches the searched destination and delivers the data packets, it waits there until it can be piggybacked on a packet bounded for its original source node. In particular, since TCP (Transport Control Protocol) acknowledges received packets, BeeAdHoc piggybacks the returning foragers in the TCP acknowledgments. This reduces the overhead generated by control packets, saving at the same time energy. *Bee swarms* are the agents that are used to explicitly transport foragers back to their source node when the applications are using an unreliable transport protocol like UDP (User Datagram Protocol). The algorithm reacts to link failures by using special hello packets and informing other nodes through Route Error Messages (REM). In BeeAdHoc, each MANET node contains at the network layer a software module called *hive*, which consists of three parts: the *packing floor*, the *entrance floor*, and the *dance floor*. The entrance floor is an interface to the lower MAC layer; the packing floor is an interface to the upper transport layer; the dance floor contains the foragers and the routing information.

Beehive and *BeeAdHoc* have been implemented and evaluated both in simulation and in real networks. Results demonstrate a very substantial improvement with respect to congestion handling, for example due to hello messages overhead and flooding, and proved both the algorithm far superior to common routing protocols, both single and multipath.

III. IP ADDRESS AUTO-CONFIGURATION

The most important constraint of ad-hoc addressing schemes is to guarantee the uniqueness of node addresses so that no uncertainty appears in communication. This is not a trivial task because of the dynamic topology of ad-hoc networks. A MANET, indeed, can be split into several parts, and several MANETs can merge into one, and an indefinite number of nodes coexisting in a single network may participate concurrently in the configuration process. Moreover, the wireless nature, such as limited bandwidth, power, and high error rate make the problem even more challenging. Besides handling a dynamic topology, the protocols must take into account scalability, robustness, and effectiveness. Finally, in IPv6, a protocol is expected not only to deal with the local addressing, but also the global addressing. Since 1998, several address auto-configuration protocols for IPv4 and IPv6 have been proposed, each of them attempting to achieve a level of optimization for a particular aspect [6], [22].

In the sequel we will describe two well known IP auto-configuration protocols, namely *Buddy* and *AntConf*, that we implemented with the aim to compare their performances with those of the proposed algorithm *AutoBeeConf*.

A. The Buddy Protocol

Buddy is a stateful protocol where every node stores a disjoint set of IP addresses which it can assign to a new node without consulting any other node in the network. At the beginning, only one node in the network **has** the entire pool of IP addresses; this node detects no neighbors, thus it auto-assigns itself with the first IP of the pool, entitles the network with an ID (Identifier), and becomes the *network initiator*. A new node, that wants to join the network, periodically sends broadcast messages reclaiming an IP address. The initiator assigns an address to it, divides the pool of IP addresses into two sets, gives one half to the requesting node, and keeps the other half with itself; the protocol agreement makes the requesting node to auto-assign itself with the first address in the received set. This process continues and eventually all the nodes in the network have a set of addresses to assign to other nodes. As a consequence, a requesting node can also receive one or more responses; in such a case, it will choose the first node that replies. If a node receives a request and has no available addresses, it should request its neighbors. Three different scenarios are possible: it searches its IP address table for possible one hop neighbor candidates and increment by one the radius of search if it finds no address availability; it sends a broadcast message to its one hop neighbors and a 2 hop broadcast if it receives no reply; it searches its IP address table for the node with the biggest block and contacts it directly. The synchronization of the address tables makes each node to periodically broadcast its address table. The detection of address leaks is accomplished by buddy nodes: if one node detects that another is missing, it merges its IP pool with its own IP pool. When networks merge, conflicting nodes have to give up their address space and acquire a new set of addresses. The protocol guarantees address uniqueness, does not generate unnecessary address changes, and is distributed, but it is complex to implement, produces a scarce balanced address assignment, and requires a consistent flooding that strongly increases the network overhead, [10].

B. The AntConf Protocol

AntConf is a stateful protocol based on the Ant Colony meta-heuristic, where every node creates and propagates through the network at least one *originator ant*. The node may destroy, reproduce or duplicate the originator ant that, on its own, has the exclusive right to initiate any change involving its parent IP address when a conflict is detected. The ants, usually identified by means of the Medium Access Control (MAC) of their originator nodes, spread their own node information, collect other node information, and induce feedback within the network using the environment as interchange means. The environment is usually realized as a small segment of memory that nodes and ants hold and employ during their mutual updating interactions. Basically, the memory segments contain the MAC address, the IP address and a timestamp for each of the currently known nodes. Timestamp reflects the time elapsed since the node initialization; in order to deal with a totally distributed control, nodes do not need synchronization. When the process begins, each memory segment would have only one entry pointing to itself; as the algorithm progresses information about other nodes will be brought in, and the environment will be dynamically built. At the boot time, a set of IP addresses is available for auto-configuration; each node randomly picks up a unique address, and creates its originator ant that starts its journey through the network. At each step the next hop is chosen with respect to the optimization criterion suggesting to reach the least recently updated node. The exchange of information between a node and an ant is based on the timestamps the ants carry on a per entry basis. On a network with *n* nodes, the ants carry *n* IP addresses, one for each node, usually the most recent ones according to its knowledge. When information exchange between the node and the arriving ants takes places, either of them updates itself based on the timestamps. Whenever an ant during the process of its journey detects a conflict for the node it has originated from, it takes responsibility to inform and have it changed. A conflict is detected when two or more nodes have chosen the same IP address. Conflict resolution mechanism is based on mechanisms followed in Zero-Configuration networks. The node that has the least MAC address takes the responsibility to have its node change its IP address to a different one. This is not a one step process but the result of various interactions among the swarms. The conflict resolution mechanism will continue until a state wherein all the nodes have unique IP address is reached. Due to the completely distributed control and feedback flow, the swarm based system guarantees that, even in case of node or link failure, only a partial component of information is lost so that the system can quickly recover from it. An important feature of the swarm based model is concerned with partitions which do not need to be considered as special cases. On the contrary, when partitions merge, there is a sudden increase in the number of IP address conflicts and the system has to make a large effort to respond to the new environmental change, [11], [23].

IV. AUTOBEECONF

Auto-BeeConf is the new auto-configuration algorithm for efficient ad-hoc network administration presented in this paper. The algorithm is inspired by the foraging principles of honey bees and it is supposed to share the services of the IP layer; more precisely, *Auto-BeeConf* is supposed to relay on the services of the *BeeAdHoc* routing protocol placed in the TCP/IP suite of any network node. The main features of *Auto-BeeConf* are two: first, the acquirement of the controlled multicast, and second, the intelligent division of the labor force which is done proportionally to the available food resources. The controlled multicast allows to limit the information to propagate only through a node subset in such a way that the network is poorly flooded inducing a noticeable overhead and energy saving. The labor division allows nodes to manage a number of addresses proportionally to their battery charges in such a way that the address losses are reduced when nodes leave the network because of a battery discharge. *Auto-BeeConf* is a hybrid algorithm that works through two phases. In the first one, a node that wants to join a MANET tries to get an IP address by means of state-full policy that allows it to look for an address among its neighbor nodes; in case of an iterated number of failures with respect to its request, it assumes that none of its neighbors has free addresses and starts trying with a stateless policy. In such a way the incoming node has the chance to look for conflicts as well as for a valid available address. The two phases strongly balance themselves inducing a promising improvement in performances as compared to existing state-of-the-art auto-configuration algorithms due to the reduced use of control packets.

A. Protocol Operation

A node that wants to join a MANET senses its neighbors by means of *Hello Messages* and sends an *IP Assignment Request Bee Agent* to the best of them soon after the initializations of two variables, *my_back-off* and *my_patience*. A node is better than another when its battery charge is larger; the incoming node gets such information from the *Hello Messages*.

Phase 1: The neighbor nodes receiving the request look for a free IP address in their tables. If they do not have free addresses, they only discard the request; otherwise, they divide their tables proportionally to the requiring device life-time and type , send a part to it by using the just received bee agent, and start waiting for an acknowledgment (ACK). In the case where the ACK does not return within a certain amount of time, neighbor nodes restore their original IP address table. The requiring node might receive various address blocks depending on the number of neighbors which captured its bee; it will only retain the first one acknowledging the owner in such a way that the other blocks it received may be released. However, the requiring node might also not receive any response from its neighbors. Thus it is necessary to enlarge the search radius. The *IP Assignment Request Bee Agent* must be now flooded to all neighbors in range using a number of iteration (*my_patience*) and a back-off time (*my_back-off*). Phase 1 will be iterated until the maximum value of *my_patience* will be reached. If the requiring node has not still be configured, it must enter the Phase 2 of the algorithm.

Phase 2: The node generates an IP address that is coherent with the address class, the network mask and the *my_patience* value using a MAC address based function. In order to verify the uniqueness of such an address, the node auto-assigns it to itself, resets the my*_patience* value, and generates a *BeeARP* according to the specification of the Address Resolution Protocol (ARP) in the TCP/IP suite, and the setting of a TTL. The bee-agent is sent to its best neighbors, and, each time it reaches a node, it verifies whether or not it is the destination node. In the former case, it asks the destination for a free IP address that, if available, quickly is brought back to the requiring node. In case a free IP address is not available, the bee-agent starts its journey back toward the source trying to get a free IP address from each intermediate node. In the latter case, when the bee-agent has reached a node that is not the destination, it is tried to be forwarded to the destination by means of the *BeeAdHoc* algorithm until its TTL expires. Thus, from the requiring node point of view, a *BeeARP* might come back or not. If it does, the next step is to verify whether or not it has certified the absence of an address conflict since in one case the node can begin its network activities whereas in the other it is still in lack of an address. The requiring node might also consider itself configured when the *BeeARP* TTL expires before returning home. Phase 2 is allowed to be iterated a *my_patience* maximum value number of times.

In case of failure, a *max_try* value bounds possible iteration of both phase1 and phase 2; after that the access to the MANET is forbidden to the requiring node since it is reasonable to think that IP addresses are all over, or the node is in a hotspot or dead-zone.

Network partitions do not affect the protocol operation. Network merging might create conflicts. In this case, as soon as the merging is detected by some node via the ID network, a bee swarm might be quickly broadcasted through the network with the task to resolve conflicts according to phase 2.

V. SIMULATION FRAMEWORK

The performance of *AutoBeeConf* has been evaluated as compared to *AntConf*, and *BuddyConf* using a MASON (Multi-Agent Simulator Of Neighborhoods. ….or Networks...or something), [24], [25], based simulator. Even though MASON "*is a fast discrete-event multiagent simulation library core in Java, designed to be the foundation for large custom-purpose Java simulations, and also to provide more than enough functionality for many lightweight simulation needs",* it does not allow to vary among different routing protocols. Nevertheless, MASON is Java based so that this has made it possible to design a suitable environment for the necessary scenarios.

Simulations were carried out for the set of parameters reported in TABLE I. Node and link failures were considered during burst intervals. Every node was given a set of neighbor nodes to which it can directly communicate in a duplex manner.

TABLE I. SIMULATION PARAMETERS

Parameters	***Values***
Simulation Area	35 m x 35 m to 200 m x 200 m
Mobile Node Number	50 to 1600
Mobility Pattern	Random Walk 2d Mobility Model
Node Range or Coverage	30 m
Simulation Number	288

Comparisons have been made both in discharging and not discharging modalities with a binary exponential increment of the node number step by step as shown in TABLES II and III, where each result is the average of 8 simulations grouped by number of nodes.

As TABLES II and III show, *AutoBeeConf* performances appear promising with respect to *AntConf* and *BuddyConf* , both for the number of connected nodes and the requested time to converge as the network size increases. The ant-based algorithm holds good with respect to the execution time suffering yet for the number of configured nodes. *BuddyConf* behaves well with respect to configured nodes suffering yet for the execution time as compared with both the swarm-like algorithms. *AutoBeeConf* takes advantages from the cooperation of the two phases it uses: when the number of devices that want to join the network increases, and thus the probability of the address space depletion increases, the second phase of the algorithm allows to quickly recovering all the lost addresses. TABLES IV and V simply synthesize results of TABLES II and III.

VI. CONCLUSIONS AND FUTURE WORKS

A new auto-configuration algorithm for wireless ad-hoc networks, *AutoBeeConf*, has been presented. Its simulation showed that ideas inspired from natural systems provide a sufficient motivation for designing and developing algorithms for scheduling and routing problems as well as for auto-configuration. According to the literature a reverse engineering approach has been followed that has allowed mapping concepts from a bee colony to an auto-configuration problem. The algorithm has been evaluated in a simulation environment; however, the simulation model was developed in such a way that the constraints of a real network would be taken into account. Extensive testing and evaluation under varying environmental parameters that represent a real network conditions have been done. The results from all experiments reveal that the performance of AutoBeeConf is of the order of the best auto-configuration algorithms, even though it is achieved at a much less energy expenditure.

Future works could consider extension of the protocol to deal with:

- network merging,
- global connectivity with Internet,
- security issue,
- TCP congestion,
- exploration of the honey bee colony behavior for its reengineering in other problem framework,
- Exploration of the different swarm intelligence forms.

A last consideration about the amount of things that nature has still to teach to everybody is due. It has very recently been discovered by two Stanford researchers that Pogonomyrmex *barbatus* colonies, a species of harvester ants, determine how many foragers to send out of the nest in much the same way that TCP discovers how much bandwidth is available for the transfer of data in Internet in order to avoid or recover from network congestion. The researchers are calling them the *anternet*. According to Prabhakar it is worthwhile to conclude by saying "Ants have discovered an algorithm that we know well, and they've been doing it for millions of years", [26].

REFERENCES

[1] C. Siva Ram Murthy and B. S. Manoj, "Ad-hoc wireless networks: architectures and protocols", Prentice Hall, 2004

[2] E. Royer and C.K. Toh, "A review of current routing protocols for ad-hoc mobile wireless networks", IEEE Personal Communications, vol. **6,** 1999, pp. 46-55.

[3] C. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance vector routing (DSDV) for mobile computers", Proceedings of SIGCOMM, 1994, pp. 234–244.

[4] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad-hoc wireless networks", Mobile Computing, 1996, pp. 153–181.

[5] P. Sinha, R.Sivakumar and V. Bharghavan, "CEDAR: a core extraction distributed ad-hoc routing protocol", Proceedings IEEE INFOCOM, 1999, pp. 202-209.

[6] C. Perkins, J. T. Malinen, R. Wakikawa, E. M. Belding-Royer and Sun Y, "IP address auto-configuration for ad-hoc networks", IETF Draft, 2001.

[7] E. Bonabeau, M. Dorigo and G. Theraulaz, "Swarm intelligence. From natural to artificial systems, 1999, Oxford University Press, 0-19-513159-2.

[8] D. Chrysostomou and A. Gasteratos, "Optimum multi-camera arrangement using a bee colony algorithm", Proceedings of IEEE Int. Conf. on Imaging Systems and Techniques, 2012, pp. 387 – 392.

[9] D. Chrysostomou, A. Gasteratos, L. Nalpantidis and G. Ch. Sirakoulis, "Multi-view 3D scene reconstruction using ant colony optimization techniques", Measurement Science and Technology, 2012, vol. 23, 11.

[10] M. Mohsin and R: Prakash, "IP address assignment in a mobile ad-hoc network, Proceedings IEEE MILCOM, 2002.

[11] S. Ring, V. Kumar and M. E. Cole, "Ant colony optimization based model for network zero-configuration", Proceedings SPCOM, 2004, 423-427.

[12] M. Dorigo and T. Stützle, "Ant colony optimization", MIT Press, 2004, 0-262-04219-3.

[13] M. Farooq, "Intelligent network traffic engineering through bee-inspired natural protocol engineering", Natural Computing Series, Springer, 2006.

[14] M. Dorigo, V. Maniezzo and A. Colorni, " Ant system: optimization by a colony of cooperating agents", IEEE Transactions on Systems, Man, and Cybernetics-Part B, 1996, 26 (1), pp. 29-41.

[15] G.A Di Caro, "Ant Colony Optimization and its application to adaptive routing in telecommunication networks", PhD thesis, Facultè des Sciences Appliquèes, Universitè Libre de Bruxelles, 2004.

[16] G.A Di Caro, F. Ducatelle and L.M. Gambardella, "Anthocnet: an ant-based hybrid routing algorithm for mobile ad-hoc networks. Proceedings PPSNVIII, 2004 ,LNCS , (3242), pp. 461–470.

[17] G.A Di Caro, F. Ducatelle and L.M. Gambardella, "Anthocnet: an adaptive nature-inspired algorithm for routing in mobile ad-hoc

networks", European Transactions on Telecommunications, 2005, 16 (5), pp. 443–455.

[18] H.F. Wedde and M. Farooq, "Beehive: new ideas for developing routing algorithms inspired honey bee behavior", Handbook of Bioinspired Algorithms and Applications, 2005, 21, pp. 321–339.

[19] H.F. Wedde and M. Farooq, "The wisdome of the hive applied to mobile ad-hoc networks", Proceedings of IEEE Swarm Intelligence Symposium, 2005, pp.341–348.

[20] H.F. Wedde and M. Farooq, "A performance evaluation framework for nature insired routing algorithms", Applications of Evolutionary Computing, 2005, LNCS (3449), pp.136–146.

[21] H.F. Wedde and M. Farooq, "Beeadhoc: an energy-aware scheduling and routing framework, Technical report-pg439, LSIII, School of Computer Science, University of Dortmund., 2004.

[22] J. Jeong, J. Park, H. Kim, H. Jeong and D. Kim, "Ad-hoc IP address auto-configuration", IETF draft, 2005.

[23] V. Kumar and E. Cole, "An ant colony optimization model for wireless ad-hoc auto-configuration", Proceedings of IEEE Int. Conf. on Systems, Man and Cybernetics, 2005, vol. I, pp. 103-108.

[24] http://cs.gmu.edu/~eclab/projects/mason/

[25] S. Luke, C. Cioffi-Revilla, L. Panait, K. Sullivan and G. Balan, "MASON: A Multiagent Simulation Environment", Simulation 2005, (81), 517-527.

[26] B. Prabhakar, K. N. Dektar and M. Gordon, "The regulation of ant colony foraging activity without spatial information, PLOS Computational Biology, 2012 , vol. VIII, 8, e1002670

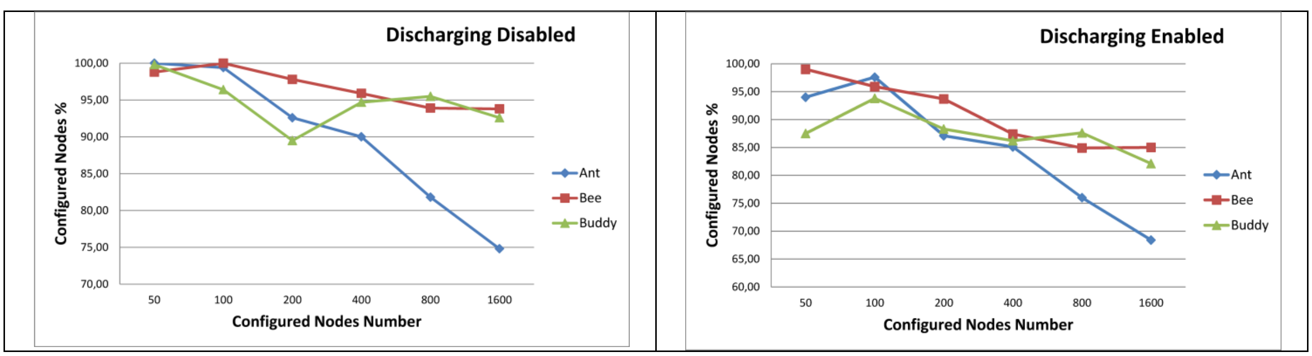
TABLE II. NUMBER OF CONNECTED NODES BY AUTOBEECONF, ANTCONF AND BUDDYCONF

AutoBeeConf			*AntConf*			*BuddyConf*		
Connected nodes			*Connected nodes*			*Connected nodes*		
Nodes	**No Discharging**	**Discharging**	**Nodes**	**No Discharging**	**Discharging**	**Nodes**	**No Discharging**	**Discharging**
50	98,80	99,00	50	100,00	94,00	50	99,80	87,50
100	100,00	95,90	100	99,40	97,60	100	96,40	93,80
200	97,80	93,70	200	92,60	87,10	200	89,50	88,30
400	95,90	87,40	400	90,00	85,10	400	94,70	86,20
800	93,90	84,90	800	81,80	76,00	800	95,50	87,60
1600	93,80	85,00	1600	74,80	68,40	1600	92,60	82,10

TABLE III. CONNECTION TIMES FOR AUTOBEECONF, ANTCONF AND BUDDYCONF

AutoBeeConf			*AntConf*			*BuddyConf*		
Time			*Time*			*Time*		
Nodes	**No Discharging**	**Discharging**	**Nodes**	**No Discharging**	**Discharging**	**Nodes**	**No Discharging**	**Discharging**
50	59,40	61,00	50	57,00	59,40	50	119,90	93,60
100	109,10	111,30	100	109,30	108,00	100	209,10	199,60
200	224,00	223,90	200	215,50	216,60	200	399,00	450,00
400	461,00	429,00	400	433,60	428,40	400	798,10	717,10
800	1142,50	931,70	800	1217,40	1061,20	800	1994,40	1935,30
1600	2442,50	2159,20	1600	3652,20	3183,40	1600	5958,10	5930,80

TABLE IV.



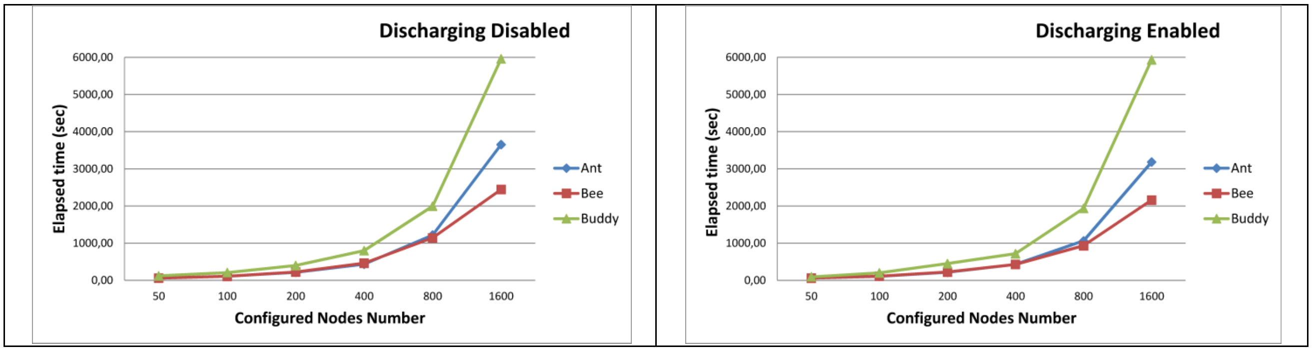
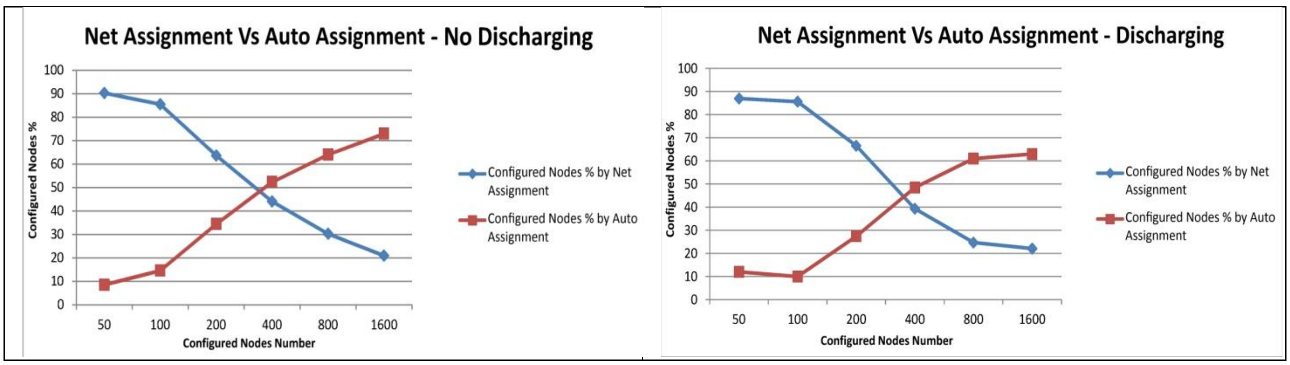

Discharging Disabled
Elapsed time (sec)
Configured Nodes Number
Ant
Bee
Buddy
Discharging Enabled
Elapsed time (sec)
Configured Nodes Number
Ant
Bee
Buddy

TABLE V.


Net Assignment Vs Auto Assignment - No Discharging
Configured Nodes %
Configured Nodes Number
Configured Nodes % by Net Assignment
Configured Nodes % by Auto Assignment
Net Assignment Vs Auto Assignment - Discharging
Configured Nodes %
Configured Nodes Number
Configured Nodes % by Net Assignment
Configured Nodes % by Auto Assignment

Improved Scatter Search Using Cuckoo Search

Ahmed T. Sadiq Al-Obaidi
Computer Science Department University of Technology Baghdad, Iraq

Abstract— **The Scatter Search (SS) is a deterministic strategy that has been applied successfully to some combinatorial and continuous optimization problems. Cuckoo Search (CS) is heuristic search algorithm which is inspired by the reproduction strategy of cuckoos. This paper presents enhanced scatter search algorithm using CS algorithm. The improvement provides Scatter Search with random exploration for search space of problem and more of diversity and intensification for promising solutions. The original and improved Scatter Search has been tested on Traveling Salesman Problem. A computational experiment with benchmark instances is reported. The results demonstrate that the improved Scatter Search algorithms produce better performance than original Scatter Search algorithm. The improvement in the value of average fitness is 23.2% comparing with original SS. The developed algorithm has been compared with other algorithms for the same problem, and the result was competitive with some algorithm and insufficient with another.**

Keywords-component; Metaheuristic; Scatter Search; Cuckoo Search; Combinatorial Problems; Traveling Salesman Problem

I. INTRODUCTION

There are several heuristic and metaheuristic algorithms have been used to solve a wide range of NP-hard problems. A large number of real-life optimization problems in science, engineering, economics, and business are complex and difficult to solve. They can't be solved in an exact manner within a reasonable amount of time [1]. Real-life optimization problems have two main characteristics, which make them difficult: they are usually large, and they are not pure, i.e.; they involve a heterogeneous set of side constraints [2]. Metaheuristic techniques are the basic alternative solution for this class of problems. Recently, many researchers have focused their attention on a metaheuristics. A metaheuristic is a set of algorithmic concepts that can be used to define heuristic methods applicable to a wide set of different problems. The use of metaheuristics has significantly increased the ability of finding solutions practically relevant combinatorial optimization problems in a reasonable time [3]. Prominent examples of metaheuristics are Evolutionary Algorithms, Simulated Annealing, Tabu Search, Scatter Search, Variable Neighborhood Search, Memetic Algorithms, Ant Colony Optimization, Cuckoo Search, and others. Which successfully solved problems include scheduling, timetabling, network design, transportation and distribution problems, vehicle routing, the traveling salesman problem and others [4].

II. BACKGROUND

There is several literature surveys applied to improve or hybridization of Scatter Search algorithm. Ali M. *et al* [5] presented improved SS using Bees Algorithm. The improvement provides SS with random exploration for search space of problem and more of intensification for promising solutions. The experimental results prove that the improved SS algorithm is better than original SS algorithm in reaching to nearest optimal solutions. Juan José *et al* [6] presented development for multiple object visual trackers based on the Scatter Search Particle Filter (SSPF) algorithm. It has been effectively applied to real-time hands and face tracking. Jose A. *et al* [7] presented the SSKm algorithm proposed methodology for global optimization of computationally expensive problems. Saber *et al* [8] presented hybrid genetic Scatter Search algorithm that replaced two steps in Scatter Search (combination and improvement) with two steps in genetic (crossover and mutation). This algorithm leads to increase the efficiency and exploration of the solution process. T. Sari *et al* [9] evaluate Scatter Search and genetic algorithm. Resource constrained project scheduling problem which is an NP-hard problem is solved with two algorithms. They conclude that genetic algorithm outperformed Scatter Search. Tao Zhang *et al* [10] presented development of new Scatter Search approach for the stochastic travel- time vehicle routing problem with simultaneous pick-ups and deliveries by incorporating a new chance-constrained programming method. A generic genetic algorithm approach is also developed and used as a reference for performance comparison. The evaluation shows the performance characteristics and computational results of the SS solutions are superior to the GA solutions. Oscar Ibáñez *et al* [11] parented a new skull-face overlay method based on the Scatter Search algorithm. This approach achieves faster and more robust solutions. The performance compared to the current best performing approach in the field of automatic skull-face overlay. The presented approach has shown an accurate and robust performance when solving the latter six face-skull overlay problem instances. Ying Xu and Rong Qu [12] presented a hybrid Scatter Search meta-heuristic to solve delay-constrained multicast routing problems, this approach intensify the search using tabu and variable neighborhood search then is efficient in solving the problem in comparison with other algorithms which is descent the search. Jue Wang *et al* [13] proposed novel approach to feature selection based on rough set using Scatter Search to improve cash flow and credit collections. The conditional entropy is regarded as the heuristic to search the optimal solutions. The experimental result has a superior performance in saving the computational costs and improving classification accuracy compared with the base classification methods.

Regarding the previous works discussed above, This paper presents new improvement to the Scatter Search algorithm using CS which is one of the several swarm intelligence methods that was proposed to solve Combinatorial Optimization problems.

The contribution is that the improved Scatter Search with CS reaching to the nearest optimal solutions than original Scatter Search.

The Scatter Search algorithm is proven successful in travelling salesman problem [14]. The Traveling Salesman Problem (TSP) is a classical NP-hard combinatorial problem. Let given a graph G = (N, E), where N = {1, ..., n} is the set of nodes and E = {1, ..., m} is the set of edges of G, which represent the costs. The c_{ij}, associated with each edge linking vertices, i and j. The problem consists in finding the minimal total length Hamiltonian cycle of G. The length is calculated by the summation of the costs of the edges in a cycle. If for all pairs of nodes {i,j}, the cost's c_{ij} and c_{ji} are equal, then the problem is said to be symmetric, otherwise it is said to be asymmetric. It represents an important test ground for many evolution algorithms [1].

The rest of the paper is organized as follows. Scatter Search Technique is described in Section 3. Section 5 presents brief description for CS Algorithm. The first enhanced SS is proposed in Section 6. Section 7 includes the second enhanced SS. In section 8. The experimental results are presented. Finally, some concluding remarks are presented in Section 9.

III. SCATTER SEARCH TECHNIQUE

Scatter Search (SS) algorithm is one of the population-based Metaheuristics. It works on a population of solutions, which are stored as a set of solutions called the Reference Set. The solutions to this set are combined in order to obtain new ones, trying to generate each time better solutions. According to quality and diversity criteria, Fig. 1 illustrates the basic SS algorithm [1, 15].

The design of a SS algorithm is generally based on the following five steps [15, 16]:

- *A Diversification Generation Method* to generate a population (Pop) of diverse trial solutions within the search space.
- *An Improvement Method* to transform a trial solution into one or more enhanced trial solutions.
- *A Reference Set Update Method* to build and maintain a Reference Set. The objective is to ensure diversity while keeping high-quality solutions. For instance, one can select $RefSet_1$ solutions with the best objective function and then adding $RefSet_2$ solutions with the optimal diversity solutions ($RefSet = RefSet_1 + RefSet_2$).
- *A Subset Generation Method* to operate on the reference set, to produce several subsets of its solutions as a basis for creating combined solutions.
- *A Solution Combination Method* to transform a given subset of solutions produced by the Subset Generation Method into one or more combined solution vectors.

After generating the new solutions which are generated from Solution Combination Method, these solutions will be improved by Improvement Method, and this solution will become a member of the reference set if one of the following rules is satisfied [15]:

```
Scatter Search Algorithm
Input: Population of the problem.
Output: The best of solutions
  Initialize the population Pop using a
  Diversification Generation Method.
  Apply the Improvement Method to the
  population.
  Reference Set Update Method (Good solutions
  for RefSet1 and Diversity solutions for
  RefSet2).
  While (itr < MaxItr) do
       While (Reference set is changed) do
          Subset Generation Method
            While (subset-counter < > 0) do
                 Solution Combination Method.
                 Improvement Method.
                 Reference Set Update Method;
            End while
       End while
  End while
  Return the best of solutions
```

Fig. 1. Basic Scatter Search Algorithm

1) *The new solution has a better objective function value than the solution with the worst objective value in $RefSet_1$.*

2) *The new solution has a better diversity value than the solution with the worst diversity value in RefSet2.*

The search is continued while RefSet is changed. If no change in RefSet, the algorithm will check if the number of iteration (*itr*) reach the max iteration (*MaxItr*) that detected by the user, then the algorithm will display the good solution(s) reached, else, the new population will be generated, and RefSet1 will be added to the start of this population.

IV. CUCKOO SEARCH ALGORITHM

CS is a heuristic search algorithm which has been proposed recently by Yang and Deb [17]. The algorithm is inspired by the reproduction strategy of cuckoos. At the most basic level, cuckoos lay their eggs in the nests of other host birds, which may be of different species. The host bird may discover that the eggs are not its own and either destroy the egg or abandon the nest all together. This has resulted in the evolution of cuckoo eggs which mimic the eggs of local host birds. To apply this as an optimization tool, Yang and Deb used three ideal rules [17, 18]:

1) *Each cuckoo lays one egg, which represents a set of solution co-ordinates, at a time and dumps it in a random nest;*

2) *A fraction of the nests containing the best eggs, or solutions, will carry over to the next generation;*

3) The number of nests is fixed and there is a probability that a host can discover an alien egg. If this happens, the host can either discard the egg or the nest and this result in building a new nest in a new location. Based on these three rules, the basic steps of the Cuckoo Search (CS) can be summarized as the pseudo code shown as in Fig. 2.

```
Cuckoo Search via Levy Flight Algorithm
Input: Population of the problem;
Output: The best of solutions;
   Objective function f(x), x = (x1, x2, ...xd)^T
   Generate initial population of n host nests xi
                                        (i = 1, 2, ..., n)
   While (t <Max Generation) or (stop criterion)
         Get a cuckoo randomly by Levy flight
         Evaluate its quality/fitness Fi
         Choose a nest among n(say,j)randomly
         If (Fi > Fj) replace j by the new solution;
         A fraction(pa) of worse nests are
           abandoned and new ones are built;
         Keep the best solutions (or nests with
           quality solutions);
         Rank the solutions and find the current
           best;
         Pass the current best solutions to the next
           generation;
   End While
```

Fig. 2. Basic Cuckoo Search Algorithm

When generating new solution $x^{(t+1)}$ for, say cuckoo *i*, a

Levy flight is performed

$$x^{(t+1)}_i = x(t)_i + \alpha \oplus Levy(\beta) \ \ldots\ldots \ (1)$$

where $\alpha > 0$ is the step size which should be related to the scales of the problem of interests. In most cases, we can use $\alpha = 1$. The product $\oplus$ means entry-wise walk while multiplications. Levy flights essentially provide a random walk while their random steps are drawn from a Levy Distribution for large steps

$$Levy \sim u = t^{-1-\beta} \qquad (0 < \beta \leq 2) \ \ldots\ldots\ldots \ (2)$$

this has an infinite variance with an infinite mean. Here the consecutive jumps/steps of a cuckoo essentially form a random walk process which obeys a power-law step-length distribution with a heavy tail. In addition, a fraction *pa* of the worst nests can be abandoned so that new nests can be built at new locations by random walks and mixing. The mixing of the eggs/solutions can be performed by random permutation according to the similarity/difference to the host eggs.

V. The Proposed Scatter Cuckoo Search

The improvement to SS algorithm was accomplished by using nature inspired swarm intelligent algorithm, which is Cuckoo Search. Cuckoo search algorithm has proven its ability in solving some combinatorial problems and finding the nearest global optimum solution in reasonable time and good performance. Because the SS algorithm is composed of several steps, there will be several places to improve the SS algorithm. However, by the applied experiments, Subset Generation Method, Improvement Method and Reference Set Update Method are the most effective steps in improving the SS algorithm.

When we try to improve the SS algorithm, the time is the big problem that is found in the Improvement Method. Where the Improvement Method is applying on all populations rather than to each new solution produced from Combination Method, so this will take a large amount of time, this will affect the SS algorithm as one of the metaheuristic algorithms that the main goal of it in solving the problems is to find the optimal solution in reasonable time.

However, when trying to improve the SS algorithm in Reference Set Update Method in SS algorithm, the results were good and in reasonable time. The steps of CS will take its solutions from steps in SS, which is Reference Set Update Method and explore more of solutions and retrieve the best solutions reached to complete SS steps. See Fig. 3, which is show the improved SS algorithm using CS.

In Reference Set Update Method, $RefSet_1$ of b_1 of the best solutions and $RefSet_2$ of b_2 of diversity of solutions will be chosen. $RefSet_1$ will enter to the new steps that added from CS to SS. The new steps provide a more diversity to the $RefSet_1$ which is benefit from the neighborhood search in the cuckoo search steps. Also the updated RefSet will contain more enhanced solutions than the old because the substitution operator forms the cuckoo solutions.

Improved Scatter Cuckoo Search Algorithm
Input: Population of the problem.
Output: The best of solutions
Initialize the population Pop using Diversification Generation Method.
Apply the Improvement Method to the population.
Reference Set Update Method (Good solutions for $RefSet_1$ and Diversity solutions for $RefSet_2$).
While (*itr* < *MaxItr*) do
While (Reference set is changed) do
Get a cuckoo randomly by Levy flight (from RefSet)
Evaluate its quality/fitness F_i
Choose a nest among n (say j) randomly
If ($F_i > F_j$) replace j by the new solution;
A fraction(pa) of worse nests are abandoned and new ones are built;
Keep the best solutions (or nests with quality solutions to substitute the RefSet);
Subset Generation Method
While (subset-counter < > 0) do
Solution Combination Method.
Improvement Method.
Reference Set Update Method;
End while
End while
End while
Output: The best of solutions

Fig. 3. Improved Scatter Search Algorithm Using Cuckoo Search

VI. Experimental Results

TSP is one of the main combinatorial problems that used as test ground for most search techniques. We apply original SS and enhanced SS algorithms to symmetric TSP as a tool to measure the performance of the proposed enhanced SS.

SS and its improvement algorithms were implemented in Microsoft Visual C# 2005 Express Edition and run on a computer whose processor is Intel Core2 Duo T657064 2.0 GHz, with 2 GB main memory, 200 GB hard disk. The algorithms were applied to symmetric instances of the benchmark TSPLIB [20] with sizes ranging over from 26 to 1379. The stop criteria are chosen as follows:

1. *If no change in Reference Set.*
2. *To reach a maximum number of iterations = 20.*

The following parameters are chosen:

- Initial population *P* =100,
- The size of | $RefSet_1$| $=b_1=10$, the size of | $RefSet_2$| $=b_2=10$ and the size of reference set | RefSet| = | $RefSet_1$|+| $RefSet_2$|=20.
- The fraction (pa) of CS is 0.25.

A first experiment compared SS with it improvements. Twenty five independent runs of each algorithm were performed. The results are shown in Table I.

TABLE I. Comparison of SS and Proposed SS-CS for Average Optimality

Instances	*Averages Of SS*	*Average of Proposed SS-CS*
Fri26	1600	**1205**
Dantzig42	1990	**1597**
Att48	100995	**83544**
Eil51	1133	**907**
Eil101	2616	**2133**
KroA100	127667	**109542**
KroB100	124799	**104989**
KroC100	126565	**106912**
KroD100	123197	**101391**
KroE100	129005	**109802**
KroB200	269085	**240008**
Lin105	91707	**72879**
Lin318	513090	**401003**
Pr76	432145	**289019**
Pr124	537678	**359097**
Pr299	646297	**505992**
Pr439	1692199	**1041839**
Pr1002	6050966	**4700199**
Nrw1379	1344099	**1001899**
Berlin52	20811	**14768**
Bier127	520107	**371892**
A280	29046	**18901**

To see clearly the difference between SS and its improvement see Fig. 4.

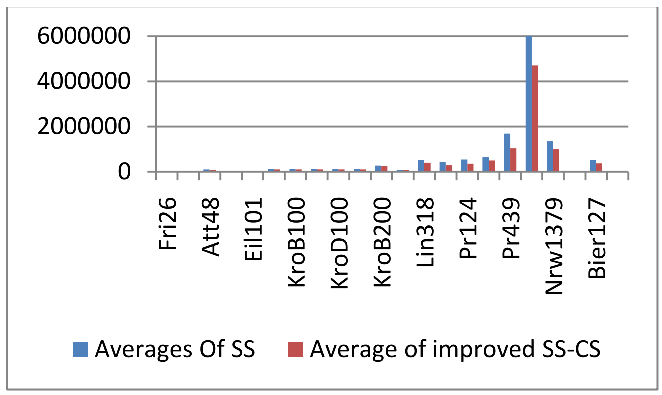


Fig. 4. Difference between SS and proposed SS-CS

Computational experiments illustrate the differences between SS algorithm, and the improved SS algorithm. The Nearest Optimal Solution (*NOPT*) for improved SS has been indicated in Table II with bold font. The difference is increased whenever the size of instance is increased.

Averages of fitness *f(x)* required to reach the nearest optimal solutions that output from original SS, and its improvement have been computed. In all instances, the 2 Improved SS obtained better results than original SS with little difference in time, averages of elapsed time and difference of the ratio between the averages of time required to reach optimal solution in improved SS and SS is ≅0.33 second.

The ratio of difference was computed as follows (Averages of Elapsed Time (sec) for improved SS - Averages of Elapsed Time (sec) for SS).

TABLE II. COMPARISON OF SS AND PROPOSED SS-CS FOR *NOPT*

Instances	***NOPT in SS***	***NOPT in Proposed SS-CS***
Fri26	1379	**1207**
Dantzig42	1810	**1195**
Att48	86890	**81899**
Eil51	1003	**858**
Eil101	2423	**1189**
KroA100	113253	**101702**
KroB100	111239	**101099**
KroC100	113539	**102081**
KroD100	113245	**102004**
KroE100	120552	**97099**
KroB200	251029	**230165**
Lin105	82838	**73997**
Lin318	494126	**441786**
Pr76	401947	**332900**
Pr124	500592	**402909**
Pr299	618178	**503128**
Pr439	1611932	**1470674**
Pr1002	5889830	**5070901**
Nrw1379	1301255	**1149099**
Berlin52	17931	**14811**
Bier127	501161	**417903**
A280	27789	**21089**

Table III shows the averages of elapsed time for SS and Improved SS algorithms for the instances in Table I.

TABLE III. AVERAGE OF ELAPSED TIME FOR SS AND PROPOSED SS-CS

Instances	***Average of elapsed time for SS (Sec)***	***Average elapsed time for Proposed SS-CS (Sec)***
Fri26	0.48	0.51
Dantzig42	0.63	0.70
Att48	0.74	0.80
Eil51	0.61	0.69
Eil101	1.11	1.20
KroA100	1.07	1.27
KroB100	1.08	1.21
KroC100	1.07	1.29
KroD100	1.09	1.31
KroE100	1.08	1.39
KroB200	2.29	2.47
Lin105	1.19	1.41
Lin318	3.91	4.21
Pr76	0.88	1.72
Pr124	1.31	1.51
Pr299	3.64	4.29
Pr439	5.51	5.91
Pr1002	15.87	16.91
Nrw1379	23.56	24.12
Berlin52	0.64	0.78
Bier127	1.55	1.75
A280	3.38	4.49

The results of improved SS will be the best because the added steps from CS in different steps of SS provided a good diversity & intensification for the new and ratio of getting *NOPT* solutions will be increased. The ratio of getting *NOPT* solution will be increased respectively with increasing the size of $RefSet_1$.

In the second computational experiment we use the same parameters in first computational experiments except for the $|RefSet_1| = b_1 = 20$ where $|RefSet| = |RefSet_1| + |RefSet_2| = 30$.

We compute the averages of fitness and elapsed time with ten runs for the same instances in Table I. The results of the second experiments are illustrated in Table IV. When we increase the value of $RefSet_1$ to 20, we found the results for SS and improved SS are better than the results in Table I.

TABLE IV. COMPARISON OF SS AND PROPOSED SS-CS FOR AVERAGE OPTIMALITY WITH $REFSET_1$=20

Instances	***Averages Of fitness for SS***	***Average of fitness Proposed SS-CS***
Fri26	1461	**1012**
Dantzig42	1751	**1129**
Att48	92156	**73987**
Eil51	1005	**973**
Eil101	2312	**1797**
KroA100	118654	**97341**
KroB100	115987	**99762**
KroC100	114982	**98967**
KroD100	111707	**97521**
KroE100	117233	**97939**
KroB200	251087	**113998**
Lin105	84590	**71017**
Lin318	475691	**347531**
Pr76	379328	**258023**
Pr124	500807	**401812**
Pr299	601011	**491763**
Pr439	1562181	**1170739**
Pr1002	5761184	**4348761**
Nrw1379	1104810	**914361**
Berlin52	17981	**12451**
Bier127	478521	**317659**
A280	27234	**19963**

To see clearly the difference between SS and its improvement with $RefSet_1$=20 see Fig. 5.

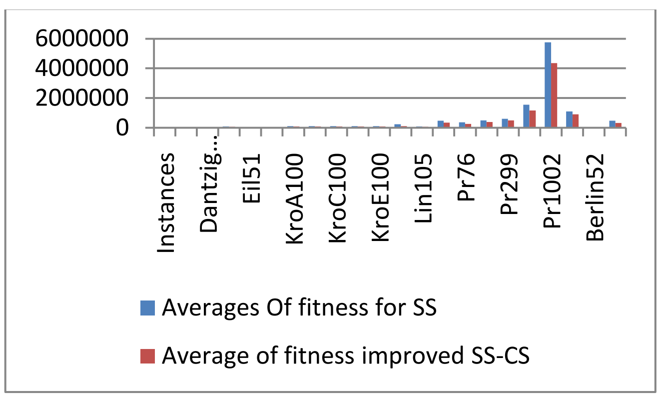


Fig. 4. Difference between SS and proposed SS-CS with RefSet1=20

In spite of the results are better with RefSet1=20, there is a still difference in time. This difference is caused by the new size of $RefSet_1$which increase the exploration and intensification for new solutions. Table V shows the NOPT results of SS, SS-CS with $RefSet_1$=20. Table VI shows the elapsed time for SS and improved SS with $RefSet_1$=20. The increased time where $RefSet_1$=20 is ≅1.2 second for SS-CS.

TABLE V. COMPARISON OF SS AND THE PROPOSED SS-CS FOR ***NOPT*** WITH $REFSET_1$=20

Instances	***NOPT in SS***	***NOPT in Proposed SS-CS***
Fri26	1364	**1108**
Dantzig42	1689	**1195**
Att48	84041	**79865**
Eil51	909	**801**
Eil101	2091	**1009**
KroA100	112772	**98124**
KroB100	114654	**98221**
KroC100	113124	**98722**
KroD100	110012	**96912**
KroE100	114789	**93582**
KroB200	231314	**211133**
Lin105	82139	**70997**
Lin318	467549	**400755**
Pr76	373254	**300991**
Pr124	498982	**400001**
Pr299	608723	**490074**
Pr439	1631578	**1410633**
Pr1002	5902741	**5000190**
Nrw1379	1298711	**1079812**
Berlin52	17172	**12018**
Bier127	480941	**400901**
A280	26576	**19754**

In the second experiments, for instances with large size such as Lin318, Pr299, Pr439, Pr1002 and A280 we noticed that the average of elapsed time with improved SS is larger than original SS with approximately 1 second only . This case can lead us to the fact that improved SS with large instances can reach to the best *NOPT* solution with a very reasonable time than original SS.

In general, comparing the time with the *NOPT* solutions isn't important for those who are looking for *NOPT* solutions, and they aren't cared about the time.

TABLE VI. AVERAGE OF ELAPSED TIME FOR SS AND PROPOSED SS-CS WITH $REFSET_1$=20

instances	***Average of elapsed time for SS (Sec)***	***Average elapsed time for Proposed SS-CS (Sec)***
Fri26	1.27	1.40
Dantzig42	1.67	1.92
Att48	1.96	2.20
Eil51	1.61	1.85
Eil101	2.93	3.32
KroA100	2.83	3.49
KroB100	2.86	3.31
KroC100	2.83	3.55
KroD100	2.89	3.60
KroE100	2.86	3.82
KroB200	6.06	6.79
Lin105	3.15	3.88
Lin318	10.36	11.15
Pr76	2.33	4.72
Pr124	3.47	4.15
Pr299	9.65	10.89
Pr439	14.60	15.85
Pr1002	42.05	43.34
Nrw1379	62.43	66.33
Berlin52	1.70	2.14
Bier127	4.11	4.82
A280	8.96	10.05

In third experiment we compare the *NOPTs* of improved SS in Table VII and VIII with results obtained by other algorithms. We compute the average deviation for the output solutions *SD = 100(NOPT – opt) / opt*, where *NOPT* is the Nearest Optimal Solution output from Improved SS and the *opt* is the optimal solution taken from TSPLIB [20].

TABLE VII. RESULTS OF IMPROVED SS ARE BETTER THAN SOME ALGORITHMS

Instances	**Optimal in TSPLIB in [20]**	***SD for NOPT* for Proposed SS-CS**	***SD* for optimal solutions in[19]**	***SD* for optimal solutions in[21]**
Fri26	937	28.81	-	34.47
Dantzig42	699	70.95	-	119.45
Att48	10628	670.59	-	573.96
Eil51	426	101.40	-	125.35
Eil101	629	89.03	-	259.61
KroA100	21282	377.87	808.51	378.78
KroB100	22141	356.61	-	347.35
KroC100	20749	391.98	854.24	389.84
KroD100	21294	379.02	-	350.37
KroE100	22068	339.99	-	345.15
KroB200	29437	681.89	828.21	662.59
Lin105	14379	414.61	835.15	393.62
Lin318	41345	968.53	880.41	962.99
Pr76	108159	207.78	744.56	216.44
Pr124	59030	582.54	801.44	599.80
Pr299	48191	944.02	894.60	991.79
Pr439	107217	1271.68	882.16	1209.28
Pr1002	259045	1857.53	927.95	1910.50
Nrw1379	56638	1928.84	891.17	2105.92
Berlin52	7542	96.38	-	127.45
Bier127	118282	253.31	724.70	259.06
A280	2579	717.72	872.48	900.34

TABLE VIII. RESULTS OF IMPROVED SS ARE FAR FROM RESULTS OF SOME OTHER ALGORITHMS

Instances	***SD for NOPT* for Proposed SS-CS**	***SD* for optimal solutions in[22]**	***SD* for optimal solutions in[23]**
Fri26	28.81	0	0
Dantzig42	70.95	0	0
Att48	670.59	0	0
Eil51	101.40	0	0
Eil101	89.03	0.107	0
KroA100	377.87	0	0
KroB100	356.61	0.036	0
KroC100	391.98	0	0
KroD100	379.02	0.019	0
KroE100	339.99	0.001	0
KroB200	681.89	0.509	0
Lin105	414.61	0	0
Lin318	968.53	0.769	0.29
Pr76	207.78	0	0
Pr124	582.54	0	0
Pr299	944.02	0.066	0.01
Pr439	1271.68	0.572	0.18
Pr1002	1857.53	-	-
Nrw1379	1928.84	-	-
Berlin52	96.38	0	0
Bier127	253.31	0.064	0
A280	717.72	0.305	0

Table VII shows how the results of improved SS-CS are better than some results such as in [19] and [21]. Also Table

VIII shows how the improved SS-CS results are far from other results of other algorithms such as [22] and [23].

VII. Conclusions

This paper presented improved SS algorithms. The improvement provides SS with random exploration for search space of problem and more of diversity and intensification for promising solutions based on the Cuckoo search algorithm. From experimental results, the average of fitness value for improved SS algorithms are better than original SS algorithm, the improvement in the value of average fitness is 23.2% comparing with original SS. From experimental results, the 2improved SS algorithms are better than original SS algorithm in reaching to nearest optimal solutions.

The elapsed time for the improved SS is larger than the elapsed time for original SS in a reasonable value. The difference in elapsed time to reach Nearest Optimal Solution isn't a problem for those whose look for optimal solutions, and they aren't cared about the time. In general, the ratio of difference isn't very large. Also, the optimal solution of the improved SS is better than some algorithms but is far away from some others.

For future work, the improved SS algorithm for TSP give an enhanced results comparing with the original SS but not good results comparing with most dependent algorithms, so it is reasonable to improve the SS & other improved SS with a mix techniques based on more than one improved steps to obtain the good results.

References

[1] El-Ghazali Talbi, “Metaheuristics from Design to Implementation,”John Wiley & Sons, 2009.

[2] F. Glover, Gary A. Kochenberger, “Handbook of Metaheuristics,” Kluwer Academic, 2003.

[3] Dorigo Mario, Stützle Thomas,“Ant Colony Optimization”. MIT Press, 2004.

[4] Eberhart, R., Y. Shi, and J. Kennedy, “Swarm Intelligence,” Morgan Kaufmann, San Francisco, 2001.

[5] Ali M. Sagheer, Ahmed T. Sadiq and Mohammed S. Ibrahim, “Improvement of Scatter Search Using Bees Algorithm”, Proceedings of the 6th International Conference on Signal Processing and Communication Systems ICSPCS2012, Gold Coast, Australia, 12 - 14 December, 2012.

[6] Juan José Pantrigo, Antonio S. Montemayor, Raúl Cabido, " Scatter Search Particle Filter for 2D Real-Time Hands and Face Tracking", Springer, 2005.

[7] Jose A. Egea, Emmanuel Vazquez, Julio R. Banga, Rafael Martí," Improved scatter search for the global optimization of computationally expensive dynamic models", Journal of Global Optimization, vol. 43, pp. 175 – 190, 2009.

[8] Saber M. El-Sayed, Waiel F. Abd EL-Wahed, Nabil A. Ismail, " A Hybrid Genetic Scatter Search Algorithm for Solving Optimization Problems", Faculty of Computers and Informatics, Operations Research Department, Egypt, 2008.

[9] T. Sari, V. Cakir, S. Kilic and E. Ece, Evaluation of Scatter Search and Genetic Algorithm at Resource Constrained Project Scheduling Problems, INES 2011 : 15th International Conference on Intelligent Engineering Systems, June 23–25, 2011, Poprad, Slovakia.

[10] Tao Zhang, W.A.Chaovalitwongse, YuejieZhang, Scatter search fort hestochastic travel-time vehicle routing problemwith simultaneous pick-ups and deliveries, Elsevier Ltd., Expert Systems with Applications 39 (2012) 6123–6128.

[11] Oscar Ibáñez, Oscar Cordón, Sergio Damas, José Santamaría, An advanced scatter search design for skull-face overlay in craniofacial superimposition, Elsevier Ltd., Expert Systems with Applications 39 (2012) 1459–1473.

[12] Ying Xu and Rong Qu ,A hybrid scatter search meta-heuristic for delay-constrained, multicast routing problems, Springer Science+Business Media, LLC 2010, Appl Intell (2012) 36:229–241.

[13] Jue Wang, Abdel-Rahman Hedar, Shouyang Wang, Jian Ma, Rough set andscatter search metaheuristic based feature selection for credit scoring, Elsevier Ltd., Expert Systems with Applications 39 (2012) 6123–6128.

[14] M. S. Geetha Devasena and M. L. Valarmathi, Meta Heuristic Search Technique for Dynamic Test Case Generation, International Journal of Computer Applications (0975 – 8887) Volume 39– No.12, February 2012.

[15] Laguna, M. and R. Martí, “Scatter Search: Methodology and Implementations in C,” Kluwer Academic Press, 2003.

[16] Fred Glove, Manuel Laguna, “Fundamentals of Scatter Search and Path Relinking,” Control and Cybernetics, Volume 29, Number 3, pp. 653-684, 2000.

[17] X. S. Yang and S. Deb, "Cuckoo search via Lévy flights". World Congress on Nature & Biologically Inspired Computing (NaBIC 2009). IEEE Publications. pp. 210–214, December, 2009.

[18] H. Zheng and Y. Zhou, “A Novel Cuckoo Search Optimization Algorithm Base on Gauss Distribution”, Journal of Computational Information Systems 8: 10, 4193–4200, 2012.

[19] Pham D.T., Ghanbarzadeh A., Koç E., Otri S., Rahim S., and M.Zaidi “The Bees Algorithm - A Novel Tool for Complex Optimisation Problems,” Proceedings of IPROMS 2006 Conference, pp.454-461.

[20] Reinelt, G.: TSPLIB, 1995. Available: http://www.iwr.uni heidelberg.de/iwr/comopt/ software/TSPLIB95/

[21] S. B. Liu, K. M. Ng, and H. L. Ong, “A New Heuristic Algorithm for the Classical Symmetric Traveling Salesman Problem,” World Academy of Science, Engineering and Technology, pp. 269-270, 2007.

[22] Jens Gottlieb and Günther R. Raidl, “Evolutionary Computation in Combinatorial Optimization,” Lecture Notes in Computer Science 3906, Springer-Verlag, pp. 44-46, 2006.

[23] Gutin, G., Punnen, A. P. (Ed.), “Traveling Salesman Problem and Its Variations,” Kluwer Academic Publishers (2002).

Texture Based Image Retrieval Using Framelet Transform–Gray Level Co-occurrence Matrix(GLCM)

S.Sulochana[1]
Research Scholar
Institute of Remote Sensing, Anna University
Chennai, India

R.Vidhya
Associate Professor
Institute of Remote Sensing, Anna University
Chennai, India

***Abstract*— This paper presents a novel content based image retrieval (CBIR) system based on Framelet Transform combined with gray level co-occurrence matrix (GLCM).The proposed method is shift invariant which captured edge information more accurately than conventional transform domain methods as well as able to handle images of arbitrary size. Current system uses texture as a visual content for feature extraction. First Texture features are obtained by computing the energy, standard deviation and mean on each sub band of the Framelet transform decomposed image .Then a new method as a combination of the Framelet transform-Gray level co-occurrence matrix (GLCM) is applied. The results of the proposed methods are compared with conventional methods. We have done the comparison of results of these two methods for image retrieval. Euclidean distance, Canberra distance, city black distance is used as similarity measure in the proposed CBIR system.**

Keywords- *Content Based image Retrieval (CBIR); Discrete Wavelet transform (DWT); Framelet Transform; Gray level- co-occurrence matrix (GLCM)*

I. INTRODUCTION

Content based image retrieval is emerging as an important research area with application to digital libraries and multimedia data bases [1]. Content based mage retrieval is a technique, which uses visual contents to search images from large scale image databases according to user's interests. During the past decade remarkable progress has been made in both theoretical research and development. There remain many challenging research problems that continue to attract researchers from multiple disciplines. The main goal of the content based image retrieval is to find images which are similar to query image visually without using any textual descriptions for the image.

Feature extraction is the basis of content based image retrieval. Images are usually represented by the visual features such as color shape and texture. There are mainly twp approaches for feature extraction in content based image retrieval. (i) Feature extraction in spatial domain and feature extraction in transform domain. Feature extraction in spatial domain based on statistical calculation on the image. Many of the spatial domain methods suffer from insufficient number of features and also sensitive to noise. The transform domain include the use of Discrete Cosine transform (DCT), Multiresolution methods such as Gabor filters, Wavelet transform, curve let and Contourlet transform for feature representation. Most of the transform domain methods is that they do not capture edge information of an image efficiently. Finding better transform domain approaches, which can capture the edge information, is a challenging field in content based mage retrieval.

Content based image retrieval (CBIR) system perform two main tasks (i) Feature extraction, where in a set of features called feature vector is generated to represent the content of the image in the database. The second task is similarity measurement where a distance between the query image and each image in the database using their feature vectors is computed so that the closest images can be retrieved.

Most of the existing methods mainly focused on the efficient extraction of color, shape and texture features. Color is the basic feature, color histograms are commonly used for color feature extraction. The color histogram method requires simple calculation. However it is unsuitable for images in which there is a great color variation .But it does not include any spatial information.

Shapes are based on contour information in an image which includes edge detection and correlograms. Edge detection leads better results only clear contour information.

Texture is one of the important features due to its presence in most of the real and synthetic radar imagery which makes high attention in CBIR and also medical imaging, Remote sensing etc. Wavelet transform have been used most widely in many aspects of image processing such as noise removal, image compression, image super resolution and image retrieval. The texture feature of an image is extracted by mean and variance of the wavelet subbands .But wavelets [2-4] loses their universality in capturing the edge discontinuities in image which is important in texture representation.

Another mMultiresolution approach, Gabor filters [5-8] consists of group of wavelets each of which capturing energy at a specific resolution and orientation. Therefore Gabor filters are able to capture local energy of the entire image .But it suffer due to computational complexity, their non-invariance to rotation as well as non orthogonal property of the Gabor filters that implies redundancy in the filtered image.

The Dual tree complex wavelets transform (DT-CWT) [9-12] as introduced by Kingsbury has been used to found an important tool for image texture analysis and feature extraction which overcome the drawbacks of both Gabor and Discrete Wavelet transform(DWT).

Curvelet transform [13]-[15] was introduced by Donoho is another multiresolution transform which provides more edge information. But computationally is not efficient in in large images.

Contourlet transform[16-19] was proposed by Do& Vetterli is a multiscale and directional image representation that uses a wavelet like structure for edge detection and than a local directional transform for contour segment detection . This transform also shift sensitive due to up and down sampling of Laplacian filters.

Cunha and Zhou proposed a modified version [20-21] of Contourlet transform which was constructed by combing a non sub sampled Laplacian pyramid and non sub sampled directional filter banks known as Non sub sampled Contourlet transform. Though much advancement made in content based image retrieval system, finding an efficient retrieval method is a major challenge for researchers. In this paper we introduced a new texture feature based on Framelet transform is proposed .The technique makes use of Framelet transform which represents the latest research on multiresolution analysis of digital image processing. This method overcomes the weakness of conventional wavelets to obtain noise free edges of images with less computational complexity.

All standard paper components have been specified for three reasons: (1) ease of use when formatting individual papers, (2) automatic compliance to electronic requirements that facilitate the concurrent or later production of electronic products, and (3) conformity of style throughout the proceedings. Margins, column widths, line spacing, and type styles are built-in; examples of the type styles are provided throughout this document and are identified in italic type, within parentheses, following the example.

II. FRAMELET TRANSFORM

Framelet transform [22-25] which is similar to wavelets but has some differences. Framelets has two or more high frequency filter banks, which produces more subbands in Decomposition.

This can achieve better time frequency localization ability in image processing. There is redundancy between the Framelet sub bands, which means change in coefficients of one band can be compensated by other sub bands coefficients. After Framelet decomposition, the coefficient in one subband has correlation with coefficients in the other subband. This means that changes on one coefficient can be compensated by its related coefficient in reconstruction stage which produces less noise in the original image.

A. *Mathematical overview*

In contrast to wavelets, Framelets have one scaling function $\varphi(t)$ and two wavelet functions $\psi_1(t)$ and$\psi_2(t)$.

A set of functions $\{\psi_1, \psi_2, \dots\dots\dots\dots \psi_{N-1}\}$ in a square integrable space L^2 is called a frame if there exist *A>0, B<*∞ so that, for any function $f \in L^2$

$$A\|f\|^2 \leq \sum_{i=1}^{N-1} \sum_j \sum_k \left|\langle f, \psi^i(2^j - k)\rangle\right|^2 \leq B\|f\|^2 \qquad (1)$$

Where *A* and *B* are known as frame bounds. The special case of A = *B* is known as tight frame. In a tight frame we have, for all $f \in L^2$ .In order to derive fast wavelet frame, multiresolution analysis is generally used to derive tight wavelet frames from scaling functions

Now we obtain the following spaces,

$$V_j = span_k \{\varphi(2^j t - k\} \qquad (2)$$

$$W_j = span_k \{\psi^i(2^j t - k\} \quad i = 1.2, \dots\dots N-1 \qquad (3)$$

With $V_j = V_{j-1} \cup W_{1,j-1} \cup W_{2,j-1} \cup \dots\dots . \cup W_{N-1,j-1}$ (4)

The scaling function $\varphi(t)$ and the wavelets $\psi_1(t)$ and $\psi_2(t)$ are defined through these equations by the low pass filter $h_0(n)$ and the two high pass filters $h_1(n)$ and $h_2(n)$

Let$\varphi(t) = \sqrt{2} \sum_n h_0(n)\varphi(2t - n)$ (5)

$$\psi_i(t) = \sqrt{2} \sum_n h_i(n)\varphi(2t - n) \quad i = 1,2 \dots. \qquad (6)$$

B. *Perfect Reconstruction conditions and Symmetry Conditions*

The Perfect Reconstruction (PR) conditions for the three band filter bank can be obtained by the following two equations

$$\sum_{i=0}^{2} H_i(z)H_i(z^{-1}) = 2 \qquad (7)$$

$$\sum_{i=0}^{2} H_i(-z)H_i(z^{-1}) = 0 \qquad (8)$$

A wavelet tight frame with only two symmetric or anti symmetric wavelets is generally impossible to obtain with a compactly supported symmetric scaling function(t).Therefore if $h_0(n)$ is symmetric compactly supported. Then antisymmetric solution $h_1(n)$ and $h_2(n)$ exists if and only if all the roots of $2 - H_0(z)H_0(z^{-1}) + H_0(-z)H_0(-z^{-1})$ has even multiplicity.

case$H_2(z) = H_2(-z)$: The goal is to design a set of three filters that satisfy the PR conditions in which the low pass filter $h_0(n)$ is symmetric and the filters $h_1(n)$ and $h_2(n)$ are either symmetric or anti symmetric. There are two cases. Case I denotes the case where $h_1(n)$ is symmetric and $h_2(n)$ is anti symmetric. Case II denotes the case where $h_1(n)$ and $h_2(n)$ are both anti symmetric. The symmetric condition for $h_0(n)$ is

$$h_0(n) = h_0(N - 1 - n) \qquad (9)$$

Where N is the length of the filter $h_0(n)$ .We dealt with case I of even length filters. Solutions for Case I can be obtained from solutions where $h_2(n)$ time reversed version of is $h_1(n)$ and where neither filter is anti symmetric. To show this suppose that $h_0(n)$, $h_1(n)$ and $h_2(n)$ satisfy the PR conditions and that

$$h_2(n) = h_1(N - 1 - n) \qquad (10)$$

Then by defining

$$h_1^{new} = \frac{1}{\sqrt{2}}(h_1(n) + h_2(n - 2d)) \qquad (11)$$

$$h_2^{new} = \frac{1}{\sqrt{2}}\left(h_1(n) - h_2(n - 2d)\right) \qquad with\ d\epsilon z \qquad (12)$$

The filters h_0 h_1^{new} , h_2^{new} also satisfy the PR conditions, and h_1^{new} and h_2^{new} are symmetric and symmetric as follows

$$h_1^{new}\ (n) = \ h_1^{new}\,(N_2 - 1 - n) \qquad (13)$$
$$h_2^{new}\ (n) = \ -h_2^{new}\,(N_2 - 1 - n)(14) \qquad (14)$$

Where $N_2 = N + 2d$

The polyphase components of the filters $h_0(n)$, $h_1(n)$ and$h_2(n)$ are given in [25] with symmetries in Equ(9) And Equ (10) satisfies the PR conditions . The 2D extension of filter bank is illustrated on "Fig .1".

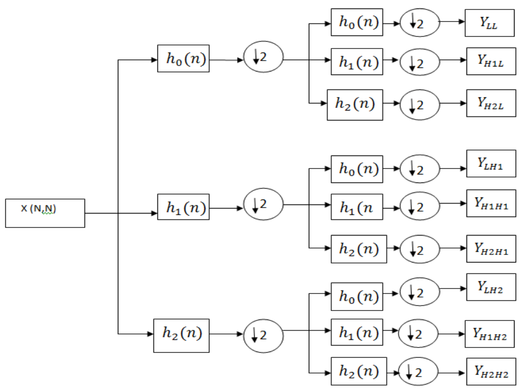


Fig. 1. An Over Sampled Filter Bank For 2D Image

III. Gray Level Co-Occurrence Matrix(GLCM)

Haralick first introduced the use of co-occurrence probabilities using GLCM [26-28] for extracting various texture features. GLCM is also called as Gray level Dependency Matrix. It is defined as "A two dimensional histogram of gray levels for a pair of pixels, which are separated by a fixed spatial relationship." GLCM of an image is computed using a displacement vector d, defined by its radius δ and orientation θ. δ values ranging from 1, 2 to 10 and every pixel has eight neighbouring pixels allowing eight choices for θ, which are 0°, 45°, 90°, 135°, 180°, 225°, 270° or 315°.

Gray level co-occurrence matrix (GLCM) is generated by counting the No. of times a pixel with i is adjacent to pixel with value j and then dividing the entire matrix by the total No. of such comparisons made. Each entry is therefore considered to be the probability that a pixel with value i will be found adjacent to a pixel of value j.

$$P_r(i,j) = \{C(i,j)|(\delta,\theta)\}$$
$$C(i,j) = \frac{P_d(i,j)}{\sum_{i=1}^{G}\sum_{j=1}^{G} P_d(i,j)}$$

Where $C(i,j)$ is the co-occurrence probability between gray levels i and j. i and j = Within the given image window, given a certain (δ, θ) Pair .G is the quantized number of gray levels.

The sum in the denominator thus represents the total number of gray level pairs (i, j) within the window. Graycomatrix computes the GLCM from a full version of the image. By default, if I a binary image, graycomatrix scales the image to two gray-levels. If I is an intensity image, graycomatrix scales the image to eight gray-levels. In order to use information contained in the GLCM, Haralick defined 14 statistical measures to extract textual characteristics. In this paper we used 4 features that can successfully characterize the statistical behaviour Let us consider P is the normalized GLCM of the input texture image.

Energy

Energy gives the sum of squared pixel values of GLCM

Energy = $\sum_{i,j} P(i,j)^2$

Contrast

Contrast measures the local variation in the gray level of GLCM

Contrast = $\sum_{i,j} |i-j|^2 P(i,j)^2$

Homogeneity = $\sum_{i,j} \frac{P(i,j)}{1+|i-j|}$

Correlation = $\sum_{i,j} \frac{(i-\mu_j)(j-\mu_j)P(i,j)}{\sigma_{i\sigma_j}}$

The choice of the displacement vector d is an important parameter of the Gray level co occurrence matrix. In general GLCM is computed for several values of d and the one which maximizes a statistical measure computed from P(i, j) is used.

IV. The Proposed Algorithm Using Framelet Transform

The basic steps involved in the proposed CBIR system as follows.

1) Feature vector (f) Decompose each image in Framelet Transfrom Domain.

2) Calculate the Energy, mean and standard deviation of the Framelet transform Decomposed image.

$$Energy = \frac{1}{M\times N}\ \sum_{1=1}^{M}\sum_{j=1}^{N}|W_k(i,j)|$$

StandardDeviation

$$(\sigma_k) = \sqrt{\frac{1}{(M\times N)}\sum_{i=1}^{M}\sum_{j=1}^{N}(W_k(i,j)-\mu_k)^2}$$

μ_k - *Mean value of the k^{th} Framelet transform subband.*

W_k Coefficient of k^{th} Framelet transform subband.

$M \times N$ is the size of the decomposed subband.

3) The resulting

$\bar{f} = [\sigma_1, \sigma_2, \ldots., \sigma_n, E_1, E_2 \ldots\ldots E_n\ ]$ is used to create the feature database.

4) Apply the query image and calculate the feature vector as given in step (2) & (3).

5) Calculate the similarity measure using Euclidean distance, Canberra distance, Manhattan distance.

6) Retrieve all relevant images to query image based on minimum Euclidean distance, Canberra distance, and Manhattan distance. The flow of algorithm is shown in "Fig.2".

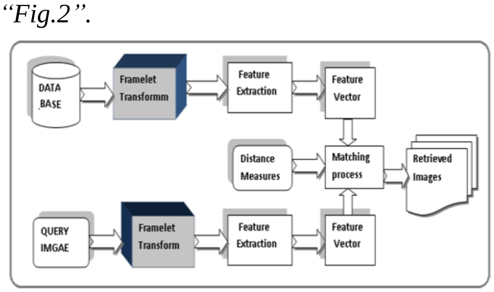


Fig. 2. Flow Of Algorithm Using Framelet Transform

V. The Proposed Algorithm using Framelet Transform-Gray level co-occurrence matrix (GLCM)

1) The images are decomposed using Framelet Transform.

2) GLCM of the Decomposed subbands are calculated with orientation and distance

3) Finally feature vectors such as contrast, Energy, Correlation, Homogeneity were extracted from GLCMs of subbands.

4) The resulting feature vector (f) is given by

$\bar{f} = [\,CONTRAST_n, ENERGy_n, CORRELATION_n, HOMOGENEITY_n\,]$] is used to create the feature database.

5) Apply the query image and calculate the feature vector as given in step (2) & (3).

6) Calculate the similarity measure using Euclidean distance, Canberra distance, Manhattan distance.

7) *Retrieve all relevant images to query image based on minimum Euclidean distance, Canberra distance, and Manhattan distance*.

The feature vectors are stored to be used in the similarity measurement. For creation of feature database above procedure is repeated for the entire image and these feature vectors stored in feature database. Flow of the algorithm is shown in Fig.3

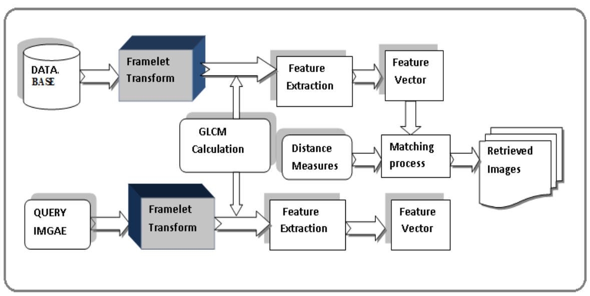


Fig. 3. Flow Of The Algorithm Using Framelet Transform+ GLCM

VI. Similarity Measurement

A query image is any one of the images from image database. This query image is processed to compute the feature vector. Distance metrics are calculated between the query image and every image in the database. This process is repeated until all the images in the database have been compared with the query image. After completing the distance algorithm, an array of distances is obtained and which is then sorted. In the presented work three types of similarity distance metric are used as given below:

C. Euclidean Distance

Euclidean distance is not always the best metric. The fact that the distances in each dimension are squared before summation, places great emphasis on those features for which the dissimilarity is large. Hence it is necessary to normalize the individual feature components before finding the distance between two images.

$$D(q,g) = \left(\sum_i |(f_i(q) - f_j(g))|^2 \right)^{1/2}$$

D. Manhattan or City block Distance

Manhattan or City block Distance is computationally less expensive than Euclidean distance because only the absolute differences in each feature are considered. City block distance is given by

$$D(q,g) = \sum_i |f_i(q) - f_j(g)|$$

E. Canberra Distance

In this distance, numerator signifies the difference and denominator normalizes the difference. Thus distance values will never exceed one, being equal to one whenever either of the attributes is zero. Thus it would seem to be a good expression to use, which avoids scaling effect. It is obvious that the distance of an image from itself is zero.

$$D(q,g) = \sum_{i=1}^{L_f} \frac{|f_{g,i} - f_{q,i}|}{|f_{g,i} + f_{q,i}|}$$

Where q is the query image,L_f is feature vector length is image in database. $f_{g.i}$ is i^{th} feature of image in the database. $f_{q.i}$ is i^{th}feature of image in the query image q.

VII. Performance Evaluation

To evaluate the retrieval efficiency of the proposed system, we use the performance measure, Recall and Precision. Recall measures the ability of the system to retrieve all the models that are relevant, while precision measures the ability of the system to retrieve only the models that are relevant.

$$Precision = \frac{\text{Number of relevant images retrieved}}{\text{Total number of images retrieved}}$$

$$Recall = \frac{\text{Number of relevant images retrieved}}{\text{Total number of relevant images}}$$

Precision Results are tabulated in Table .1 and Table .2.

VIII. Experimental Results

The algorithm is implemented in MATLAB platform. Database of 400 images of 4 different classes is used to check the performance of the algorithms developed. Same representative sample images which are used as query images as shown in “Fig. 4”.

Fig. 4. Sample Query Images From Wang Database

TABLE I. PRECISION VALUES FOR FRAMELET TRANSFORM AND DISCRETE WAVELET TRANSFORM

Feature	Distance	DISCRETE WAVELET TRANSFORM				FRAMELET TRANSFORM			
		Classes				Classes			
		Cars (%)	Flowers (%)	Horse (%)	Buildings (%)	Cars (%)	Flowers (%)	Horse (%)	Buildings (%)
Standard Deviation	Euclidean	40.32	42.25	39.68	39.86	42.32	44.37	43.67	41.68
	Manhattan	40.00	41.56	40.57	42.91	44.35	43.14	42.56	44.43
	Conberra	43.45	42.79	41.45	40.24	45.17	43.40	44.48	45.90
Energy	Euclidean	46.00	45.26	44.49	45.43	47.31	48.95	46.78	45.25
	Manhattan	46.23	43.69	42.33	44.12	46.95	45.67	47.39	47.02
	Conberra	47.56	46.10	45.32	43.78	48.90	47.65	44.73	45.62
Energy + Standard Deviation	Euclidean	48.63	49.92	47.86	46.82	49.64	50.37	51.39	50.38
	Manhattan	49.45	43.45	49.10	48,43	50.24	49.10	52.56	51.29
	Conberra	47.53	48.17	48.60	46.30	56.78	55.64	53.62	54.75

TABLE I. .PRECISION VALUES FOR FRAMELET TRANSFORM + FOR FRAMELET TRANSFORM+GLCM AND DWT+GLCM

Distance	DISCRETE WAVELET TRANSFORM+GLCM				FRAMELET TRANSFORM + GLCM			
	Cars (%)	Flowers (%)	Horses (%)	Buildings (%)	Cars (%)	Flowers (%)	Horses (%)	Buildings (%)
Euclidean	55.56	57.42	53.61	54.42	60.45	62.10	62.45	61.41
Manhattan	56.78	52.34	56.78	55.56	61.42	63.56	63.29	64.78
Canberra	59.67	55.53	57.69	56.79	65.89	66.95	64.67	66.60

To evaluate the algorithms such as Framelet transform and Framelet co-occurrence features in image retrieval, two relative algorithms DWT and DWT combined with Gray level co-occurrence are used to compare with it. We have used three different similarity measures, Euclidean distance, Manhattan distance and Canberra distance.

First algorithm is based on extracting features from coefficient in the subbands of Framelet Transform. In this Energy, standard deviation and Energy + Standard deviation is used to create the feature vectors. Three distance measures are used to match 4 sample query images with the Database includes 400 images of four different classes and the tabulated results were shown in Table 1. The proposed method gives high precision compared to discrete wavelet transform (DWT) based image retrieval.

In second algorithm Framelet-Gray level co-occurrence matrix (GLCM) method is particularly better than the method based on Discrete Wavelet transform- co-occurrence matrix (GLCM).

Proposed method we have used four GLCM statistical measures namely Energy, Contrast, Homogeneity and correlation .Gray level co-occurrence of coefficients of Framelet transform subbands angle(0^0) and distance (d=1) were used and the precision results shown in Table.2. Which shows proposed method is efficient than DWT-GLCM methods. Canberra distance measures gives better retrieval results in both the proposed methods. Average Precision is calculated and the graph is drawn between Average precision with distance measures as shown in “Fig.5”

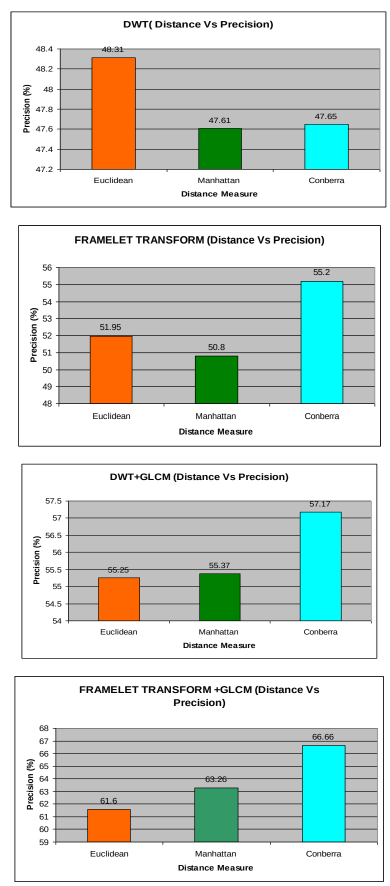


Fig. 5. Average Precision Vs Distance Measure

IX. CONCLUSION AND FEATURE SCOPE

The search for the relevant information in the large database has become more challenging. More précised retrieval techniques are needed in such cases. In this paper a new algorithm for content based image retrieval was presented. (i) Framelet Transform [Energy+ Standard Deviation] (ii) Framelet transform and GLCM were combined to build a feature vectors. Euclidean distance, Manhattan distance were used to match the four sample query image with four classes of [Cars, Flowers, Horse, Buildings] 400 images from WANG image database which includes 1000 image of ten different classes. The proposed method gives better retrieval results and higher precision. We have used only four GLCM statistical features with angle (0^0) and distance (d=1).Feature work of this study are Framelet-co-occurrence with different angle and various distances. To extract the feature vectors in images and design the CBIR system based on Framelet-co-occurrence features.

REFERENCE

[1] M. Henning, M. Nicolas, and B. David, et al, " A review of content based image retrieval systems in medical applications-clinical benefits and future directions," International Journal of Medical Informatics, vol. 73, no. 1 , pp. 1-21, 2004.

[2] [Asad Ali, Saeed Murtaza, Aamir Saeed Malik," Content Based Image Retrieval Using Daubechies Wavelet Transform". Proceedings of the 2nd National Workshop on Trends in Information Technology (NWTIT), Page 110-115, 2003

[3] K.Rajakumar, Dr.S.Muttan,"Medical Image Retrieval using Energy Efficient Wavelet Transform". 2010 Second International conference on Computing, Communication and Networking Technologies. Pages 1-5

[4] Balamurugan, V. AnandhaKumar, P,"An integrated color and texture feature based framework for content based image retrieval using 2D Wavelet Transform". Computing, Communication and Networking, 2008. ICCCn 2008. Pages 1-16.

[5] Youssef, Sherin M.; Mesbah, Saleh; Mahmoud, Yasmine M. "An efficient content-based image retrieval system integrating wavelet-based image sub-blocks with dominant colors and texture analysis Information Science and Digital Content Technology (ICIDT), 2012**, Vol. 3 Pages 518 – 523**

[6] Ju Han, Kaikuang Ma, "Rotation-invariant and scale-invariant Gabor features for texture image retrieval, "Image and Vision Computing, Vol. 25, Issue 9, 1 Sep.2007, pp. 1474-1481

[7] L. Chen, G. Lu, and D. S. Zhang, "Effects of Different Gabor Filter Parameters on Image Retrieval by Texture", In Proc. of IEEE 10th International Conference on Multi-Media Modelling, pp.273-278, Australia, 2004

[8] Mrs.Smita Jawale,"Gabor Wavelet Transform and Co-occurrence Matrix Based Texture Features For CBIR". International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 National Conference on Emerging Trends in Engineering & Technology (VNCET-30 Mar'12)

[9] Bayram, I.; Selesnick, I.W," A Dual-Tree Rational-Dilation Complex Wavelet Transform". Volume 59, issue 12, Pages 6251-6256

[10] Hua-Zhang Wang, Xiao-Hai He; Wen Jiao Zai," Texture image retrieval using dual-tree complex wavelet transform". Wavelet Analysis and Pattern Recognition, 2007. ICWAPR '07. Vol.1.Pages 230-234.

[11] Adeel Mumtaz , S.A.M. Gilani, Tahir Jameel," A Novel Texture Image Retrieval System Based On Dual Tree Complex Wavelet Transform and Support Vector Machines". IEEE--ICET 2006,2nd International Conference on Emerging Technologies Peshawar, Pakistan, 13-14 November 2006. Page163-168.

[12] Manesh Kokare, P. K. Biswas, , B. N. Chatterji," Rotation-Invariant Texture Image Retrieval Using Rotated Complex Wavelet Filters". IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics, Vol. 36, NO. 6, December 2006.

[13] M. H. Ould Mohamed Dyla, H.Tairi, A.Aarab," Content Based Image Retrieval Based on BEMD: use of Curvelet Transform coefficients distribution and Gabor wavelets". Multimedia Computing and Systems (ICMCS), 2011, page 1-5

[14] P.Manipoonchelvi, K.Muneeswaran," Significant Region Based Image Retrieval Using Curvelet Transform," 2011 International Conference on Recent Advancements in Electrical, Electronics and Control Engineering. Pages 291-294.

[15] Yungang Zhang, Lijing Gao, Wei Gao, Jun Liu," Combining Color Quantization with Curvelet Transform for Image Retrieval". 2010 International Conference on Artificial Intelligence and Computational inelligence.Vol.1 Pages 475-479

[16] Hoang Ng-Duc, Thuong Le-Tien, Tuan Do-Honl, Cao Bui-Thu, Ty Ng-Xuan," Image Retrieval using Contourlet based interest Points".10th International Conference on Information Science, Signal Processing and their Applications (ISSPA 2010). Pages 93-96.

[17] B. Syam, Y. Srinivasa Rao," Integrating Contourlet Features with Texture, Color and Spatial Features for Effective Image Retrieval".Information Management and Engineering(ICIME) 2010.pages 289-293.

[18] Jian-Zhong Ma, Xin-Wu Chen," Contourlet-S and Generalized Gaussian Model Texture Image Retrieval System". 2010 International Symposium on Intelligence Information Processing and Trusted Computing. Pages 422-425

[19] Ali Mosleh, Farzad Zargari, Reza Azizi," Texture Image Retrieval Using Contourlet Transform ".Signals, Circuits and Systems, 2009. ISSCS 2009.Pages 1-4

[20] Jian-Zhong Ma, Xin-Wu Chen , Li-Juan Zhong," Texture Image Retrieval System Based on Non-Subsampled Contourlet and Generalized Gaussian Density Model". International Conference on Artificial Intelligence and Computational Intelligence 2010.Vol. 2.Pages 181-185

[21] Ma Yuehui, Geng Peng, Liu JianHua , Jiao Yan," Image Retrieval Based on NSCT from the Science and Technology Resource Image Database". 2009 Second International Conference on Future Information Technology and Management Engineering. Pages 57-60

[22] Hadeel N.Al-Taai, "A Novel Fast Computing Method for Framelet Coefficients," American Journal of Applied Sciences 1522-1527, 2008.

[23] Ivan W. Selesnick,"A Higher Density Discrete Wavelet Transform". IEEE Transactions on signal procesing, Vol. 54, NO. 8, August 2006.

[24] A. Farras Abdelnour, Ivan W. Selesnick,"Symmetric Nearly Shift-Invariant Tight Frame Wavelets". IEEE Transactions on signal processing, Vol. 53, No. 1, January 2005

[25] I.W.Selesnick and A.F.Abdelnour, "Symmetric Wavelet Tight Frame with two generators", Applied and Computational Harmonic Analysis, Vol.17, pp.211-225, 2004.

[26] Miroslav Benco, Robert Hudec, "Novel Method for Color Textures Features Extraction Based on GLCM" Radio engineering, Vol. 16, NO. 4, December 2007.

[27] Ruziana Mohamad Rasli, T Zalizam T Muda, Yuhanis Yusof, Juhaida Abu Bakar," Comparative Analysis of Content Based Image Retrieval Technique using Color Histogram. A Case Study of GLCM and K-Means Clustering". Intelligent Systems, Modelling and Simulation (ISMS), 2012 .Pages 283-286.

[28] Bingbing Liu, Soo Chin Liew," Texture retrieval using grey-level co-occurrence matrix for Ikonos panchromatic images of earthquake in Java 2006".Geoscience and Remote Sensing Symposium, 2007, Pages 286-289

[29] D.Venkata Rao, N.Sudhakar, B.Ravindra Babu, L.Pratap Reddy," An Image Quality Assessment Technique Based on Visual Regions of Interest Weighted Structural Similarity", GVIP Journal, Volume 6, Issue 2, September, 2006

[30] Shanshan Li, Richang Hong, Xiuqing Wu, A Novel Similarity Based Quality Metric for Image Fusion", 2008 IEEE ICALIP2008

[31] Zhou Wang, Alan C. Bovik, "A Universal Image Quality Index" IEEE, Signal Processing Letters, Vol. 9, NO. 3, March 2002

[32] Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE* Transactionson Image Processing, vol. 13, no. 4, pp. 600-612, Apr. 2004.