

MAHII: Machine And Human Interactive Interface

G. Sahoo

*Department of Computer Science & Engineering
Birla Institute of Technology, Mesra,
Ranchi: 835 215, India*

drgsahoo@yahoo.com

Bhupesh Kumar Singh

*Department of Computer Science & Engineering
Lingaya's Institute of Mgt. & Technology,
Faridabad: 121002, India*

kumar.bhupesh04@gmail.com

Abstract

Free-style sketching is more natural than drawing a sketch using mouse and palette based tool. A number of sketching systems have been developed but not much attention is paid towards friendliness of the system. We present here an efficient Sketching system MAHII for sketching the strokes from multiple domains. The proposed paper demonstrates how a sketching system can be made user friendly and user interactive too. We also present the architecture of our system that solves many flaws encountered in the previously built systems. The interface should be such that it minimizes the gap between a human being and a machine. Our system provides a robust architecture that proves to be efficient. It also allows the user to sketch the diagram freely. And at the same time it edits the sketches as well as recognizes the sketches efficiently. This paper represents more robust and natural environment to the user. The MAHII provides the user an interface that is built in accordance with a novice user. MAHII provides free style sketching, recognizes the sketch produced by the user with accurate results.

Keywords: MAHII, sketch, interface, stroke, multi-domain, Heuristics.

1. INTRODUCTION

Sketching and drawing are being used in various aspects of life. In the recent years, many sketching systems have been developed, but there is always a gap between how a user draws a sketch and how a machine interprets the sketch. A little has been thought about the development of interactive human-machine interface. However, for some specific purpose many pen-based sketching systems and sketch recognition systems have been developed in the literature. But these systems suffer from the basic problem of naturalness of the sketching environment [2]. The sketching interface must be such that it allows the user to draw sketches freely and naturally. It must know when the system should take input from the user, edit the sketch as well as when it should recognize the sketch.

For sketching and recognizing the user drawn sketches, we propose here a system known as Machine And Human Interactive Interface (MAHII). The proposed system gives a solution to the problems that have been encountered in previously built sketching interfaces. Further, our

system works on multi-domain still maintaining the efficiency and naturalness of the system. We present the interface architecture of our system whose framework has already been explained in our earlier work [1]. This system MAHII is built so that it proves to be useful to a computer-expert and even to a novice person. It contains some of the known commands of computer as well as normal behavioral commands that a layman can use in sketching the diagrams. The System MAHII also gives the user more liberty to draw sketches freely and naturally. Moreover, the user need not to learn every capability of the system but still he can draw the sketch and get accurate results of recognition.

The paper is organized as follows. In the next section we discuss the design issues of the sketching systems followed by a brief overview of the proposed system. Section 4 and 5 take care of the presenting the abstract view and layered architecture of the proposed system. Section 6 describes the interface commands followed by experiments and user feedback in the section 7. The comparison study has been discussed in section 8 followed by the conclusion and future work in the following sections.

2. DESIGN ISSUES

Various design issues have been predicted from previously sketch-based systems. We arrive at the following design issues that can help in overcoming these issues successfully.

- i) The system should display recognition result only when the user completes sketching.
- ii) The system should have some editing layers so that an editing layer can perform editing after every sketch input
- iii) The recognition should take place continuously and simultaneously but the result can be displayed only when the waiting time of the user is more than average waiting time between strokes.
- iv) The recognition process should be restricted to a single domain until automatic domain detection becomes feasible.

This can be achieved by the use of some interfacing mechanism, which is our main concern in this work. The primary goal of the interfacing mechanism is to connect the domain class description to the recognition engine only when its few corresponding elements are recognized by the recognition engine through Heuristic engine using a database containing shapes. With or without giving more stress to the recognition efficiency, one should go for describing a system that is more natural having user-friendly environment. Further, the system should possess a powerful and robust architecture and can be more useful for the user.

3. THE PROPOSED SYSTEM

In this section we give the overview of our proposed system called Machine And Human Interactive Interface (MAHII). By defining stroke as a set of points from pen-up, we categorize strokes that are drawn or sketched as:

1. Single stroke input
2. Multi strokes input

And then we say that

- i) The system MAHII recognizes both single stroke input and multi-strokes input as well.
- ii) Further, it provides a new style of interface that takes care of human-machine interaction as well as free-style sketch recognition.
- iii) Even by considering a single window for both input and output in our designing process we still maintain a low complexity regarding the architecture of the system. For this we provide a three-layered architecture that we will explain in subsequent sections.

In our construction, we consider two modes – sketch mode and edit mode. MAHII provides the user more friendly environment because it does not require the user to select modes –sketch

mode or editing mode. When the user draws sketches, the system shows him the edited images simultaneously based on which he/she can modify the respective sketch. So, he/she does not need to switch between sketch mode and edit mode [4], [13]. User can draw whenever he wishes to draw. MAHII can work in single domain as well as multi-domains. Even the user is not required to select the domain which he wants to draw in. For this, we provide domain class descriptions for each domain and an interface mechanism through which the system links to the particular domain automatically. MAHII automates all the functionalities. MAHII does not use specialized tools like palette based tools etc. and provides features for novice as well as expert users. So, it fulfils the demands of all types of users.

Again it can be pointed out here that some work has been done on Human machine interaction [5] but in solitude; similarly many pen-based sketching systems have been developed but without paying much attention towards human machine interaction. However, the proposed system explores the user aspects of a recognition based sketching system that efficiently recognizes the naturally drawn sketches and displays the result on the same window. MAHII reduces the complexity and increases the efficiency of the sketching system and gives the result with accuracy. It changes itself according to the user environment.

4. ABSTRACT VIEW OF THE SYSTEM FRAMEWORK

The system framework (Figure 1) that has been discussed in [1] gives the information about the working of the system. Now, this framework is implemented as the architecture of the system and the corresponding interface is named as MAHII. The architecture is explained below.

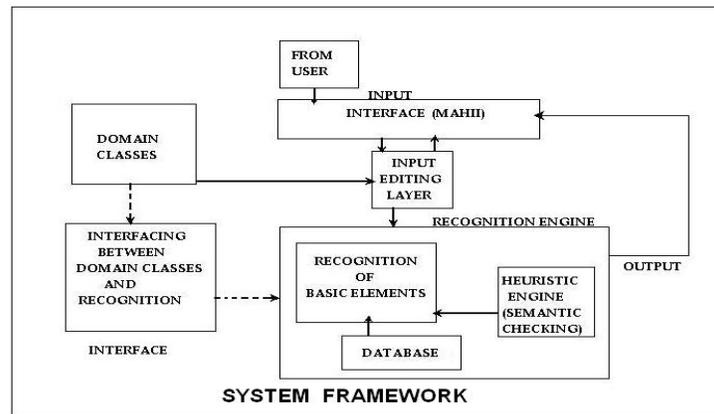


FIGURE 1: System framework

User provides input to the system by drawing sketches on the MAHII. This interface gets the sketch input from the user and sends it to the input refining block where input is refined by removing noise from the sketch input and then it is sent to the recognition engine as well as to the MAHII window. The edited or refined sketch is sent back to MAHII window because it displays the user the edited version of the sketch drawn at previous time instant. If it is not according to the user's requirement, user can change it by using simple commands that are of common behavior. We explain such commands in a later section. When the edited input is sent to MAHII, it is also sent to the recognition engine also. After recognition, the output of the recognition is sent to the MAHII window. The flexibility and ease use of this system reflects here that user needs not to know when he should change the sketch mode, edit mode or recognition mode. The system takes care of these things by itself.

If a user draws a stroke say in 2 seconds and next stroke after 4 seconds then he draws the third stroke after 5 seconds and so on, the system takes the average of the first few time gaps between the consecutive strokes. Thus, it calculates the average of time difference then sets the

systems' recognition time greater than the calculated average waiting time. The system does this for every user. So, if a user is faster in sketching, the system provides him results in less time and for a novice user, it displays the result according to the time taken by the user to draw next stroke. Thus, the system changes itself according to the user. It reduces the problem when the system should or should not recognize sketches.

The interface, which we represent in the framework that is between domain class description and recognition engine, helps the system to work in a multi-domain environment while it works actually over a single domain. In comparison to the earlier multi-domain sketch systems [4], [12], [13] where the user had to select explicitly the domain that he wants to use, MAHII does not impose any such requirement. The recognition engine actually requires the domain descriptions. Moreover, the database contains knowledge about recognized shapes and their particular domain. So, when the system needs to select a particular domain, it identifies few basic shapes or elements that are parts of a particular domain followed by the class description with the recognition engine. By using this modular approach we not only increase the system efficiency, naturalness of the interface but also reduce the complexity of the system and enable the system to work in a multi-domain environment. For an example, let us consider three domains as one domain for mechanical engineering, one domain for electronic circuit designing and one domain for dancing choreography. Initially, the user draws the input in a particular domain and gets the edited and refined sketch at the next instant of time. If he draws 3-4 basic shapes, the system relates them to the particular domain and gets to know that it belongs to, for instance, choreography domain. It gets this basic information, of course, from the corresponding database. Then, it interfaces the choreography domain with the recognition engine.

5. LAYERED ARCHITECTURE OF MAHII

The MAHII interface uses modular approach. Basically the architecture of MAHII consists of three layers namely

- i) Input layer or sketching layer or presentation layer
- ii) Editing layer or middle layer or refining layer
- iii) Core layer or main processing layer or recognition layer

and are depicted in Figure 2.

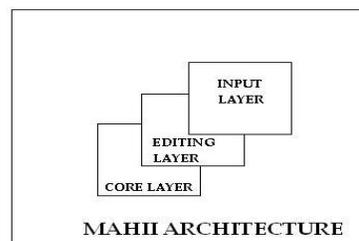


FIGURE 2: MAHII architecture

5.1 Input layer

The input layer is the window that is visible to the user for drawing the sketches. The user can draw frequently the sketches and these are used as input to the system. The user even may not have the knowledge of the functionality of the computer. As the user draws the symbols or sketches, he gets the edited image as well. The user need not wait for the system as and when to draw a sketch on a pen-based system.

This window after taking the input from the user not only displays the smooth sketch but also displays the recognition layer's result. We consider many issues and tackle various problems encountered in the earlier sketch interfaces.

The sketch interface MAHII consists of a single window. However, the editing and recognition takes place in the backend. The window represents simple, congestion-free screen to the user.

Though it contains menus for the computer known user still it contains many commands that a novice user can use.

We now describe some of the advantages of MAHII over existing system like SkRUI. The system SkRUI which was developed earlier contains two separate windows [4]. User draws in one window and gets the result in the other window. So, user switches explicitly between the two. MAHII makes this task easier. It uses a single window for all purposes and it does not have any synchronization problem. Further, many interfaces provide menu based tools like edit, undo, redo, cut, copy, paste. We make these tools available in our system in addition to some extra natural commands that can easily be used by novice.

In SkRUI, user cannot draw new items when it is in edit mode whereas MAHII does editing by itself and allows the user to draw the new sketches as well as to modify the previously drawn items. For this purpose, it uses buffers for temporary storage of previously drawn sketches and whenever the user wants to copy a sketch, he just needs to redraw the sketch over the existing one. The MAHII interface shows large buttons (menu based) so that user can easily identify them and the button is labeled with the pictures of what it does.

5.2 Editing Layer

This layer performs the editing of the sketch input. This layer includes the functioning of image filtering, image segmentation and image editing. It provides input to the recognition layer. It gets the input from the input layer and refines it by removing the noise from the sketch and increases the contrast of faded image/sketch. Any unclosed or unconnected Figures or basic shapes are connected by the system itself [6]. For example, if user wants to draw a circle, but incomplete, the editing layer does this for the user (Figure 3).

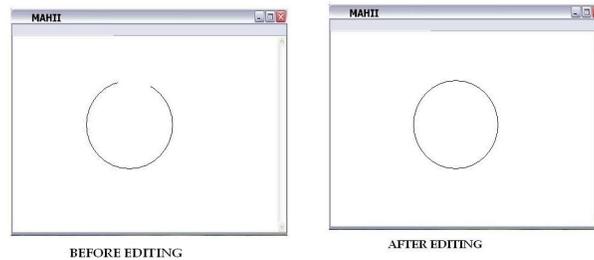


FIGURE 3: Working principle of editing layer

This layer takes input strokes one by one from the input layer as soon as the user draws the strokes on the MAHII interface. If the user draws a stroke at t time instant, the editing layer gets the input at say $(t+\Delta t)$ time instant and edits during $(t+2\Delta t)$ and it returns the result at $(t+3\Delta t)$ time instant to both the input layer and recognition layer (Figure 4). The recognition layer does its job during $2\Delta t$ units of time. The final output of recognition layer gets displayed at $(t+5\Delta t)$. Here Δt represents a very small delay of the order of microseconds.

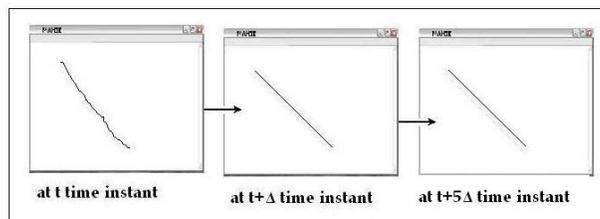


FIGURE 4: Process of editing at different time instants

Previously developed systems have a drawback that system should wait until the user finish sketching to display editing or recognition results. But since both of these activities take place simultaneously at the backend so, neither user nor the system should wait for sketching or responding respectively.

Here, an alternation that we want to provide to the user to know about the mode is that one can use pen-based editing, that is, when the cursor is in pen-mode, then the user can draw sketches and when it is in cursor mode it means the editing mode. Thus, sketching and editing use distinct pen motions.

5.3 Recognition layer

This layer is the core of our proposed system. This is the only layer that is given the power of recognition (the output) and named as recognition layer. It gets the input from the editing layer as mentioned above and uses heuristic engine to perform semantic checking, and searches for the relevant details from the system database. Finally, it shows the result to the MAHII interface or input window.

For processing, the system uses the domain class descriptions through an interface. The heuristic engine is being used here for structure matching and semantic checking. We can say that the recognition layer introduces the concept of domain independence. It recognizes the basic elements belonging to a particular domain using the Heuristic engine and database of samples of shapes and then identifies to which domain those elements belong to and then selects a particular domain class description out of many and connects it to the recognition engine. Thus, it makes the architecture as a disconnected type of architecture. When a new domain is referred to, the previous domain interface gets destroyed. It also helps the domain descriptor to define separate domain class descriptions and different domains never mingle with each other. This also reduces the complexity of domain description plus interprets the results with accuracy.

The recognition layer recognizes the output and displays the result to the input-output layer of the interface (MAHII). The recognition layer provides the result in real time because, it calculates the average waiting time for each user and then responds accordingly. It does not wait for editing or recognition by user, it starts working as soon as it gets the input.

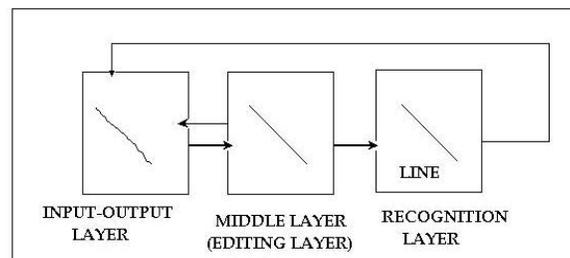


FIGURE 5: Abstract layered view

Based on the input-output, we represent a simple view of layered architecture of MAHII interface. The Figure 5 reflects the simple view of a three layered architecture through which we can estimate the real time responses. We use a single window for getting input from the user as well as for showing the output of editing and recognition of the sketch. This is an efficient approach since the user uses a single interface there is no need to switch between the sketch mode and the edit mode.

6. INTERFACE COMMANDS

Our system promises to be a user-friendly interface by indulging itself according to the user's choice. It can respond the way the user behaves. For a computer-literate user, it provides menu-

based buttons and tools. At the same time, it provides normal or generally used commands [7] that a novice can use it as he/she draws the sketch on the paper with a pen.

If a user draws a sketch on a paper and if it is not drawn according to the user's choice then he/she simply cuts it using a X symbol (Figure 6(a)). In the similar way, this command deletes the figure. If a user wants to copy the same figure he/she needs to redraw it over the figure itself (Figure 6(b)). This creates a copy in the buffer of that window area wherever the user leaves the cursor or with a click the sketch can be pasted. For the same purpose, the user can use the command C, say, for copy and P for paste (Figure 6(c)).

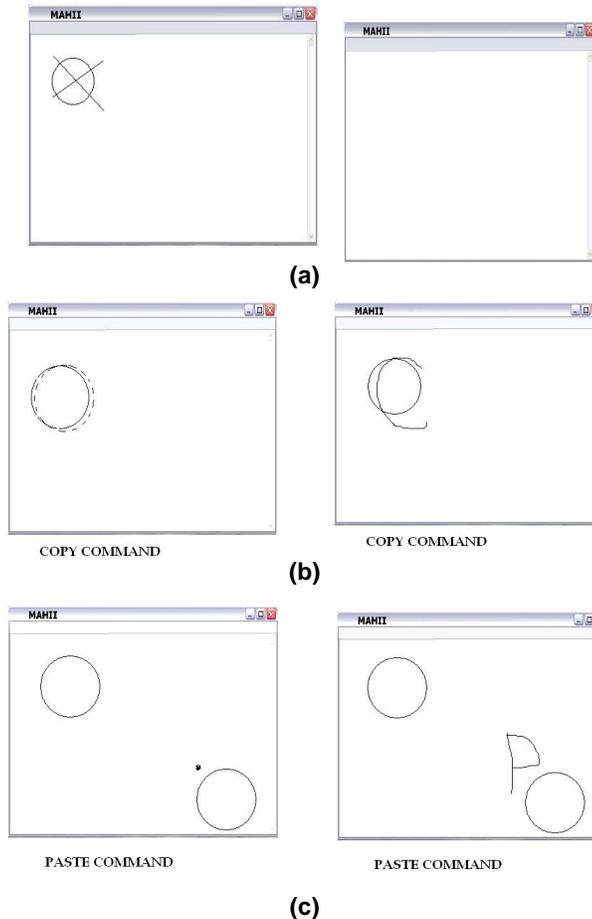


FIGURE 6: Demonstration of (a) Delete command, (b) Copy command, (c) Paste command

If user wants to cut a sketch and paste it somewhere else he needs to delete the figure after copy and paste. If user wants to redo, he can simply do it by using the command like R or r over the window area which will redo the last undo (Figure 7) part.



FIGURE 7: Redo command

To undo, the user needs to use the symbol U or u. This undoes the thing what the user has done just before (Figure 8).

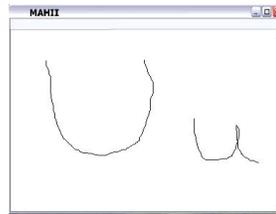


FIGURE 8: Undo command

To rub a portion of the sketch, a zigzag command can be used (Figure 9).

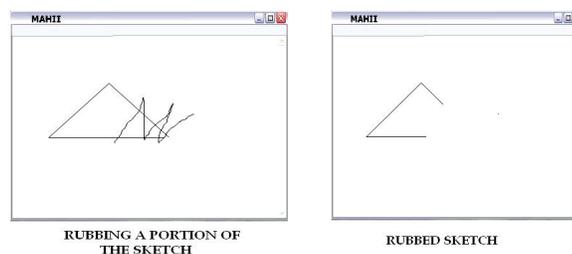


FIGURE 9: Rubbing a portion of a sketch

7. EXPERIMENTS AND USER FEEDBACK

We have provided the system to be used by any type of users. There are two groups of people to use the system. Some of them are good enough in computer skills and some have never any interaction with the computer before. Our basic aim is to get the idea of how the system MAHII is interactive and natural to the users. There are no instructions or demonstration given how MAHII basically works; instead it provides amazing results of much satisfactory and fulfilling the target of natural or user-friendly environment.

The experiment was done as follows. There were 12 users who were asked to draw sketches of various domains. There were 6 novice users and 6 were computer known people. Novice users found our system simple and easy to operate. They draw sketches of electronic circuits and shortest-path graphs without explicitly switching between the domains. Our system was successful in finding the specific domain itself. This provided the novice users enough flexibility. Even MAHII provided self-editing feature that the users appreciated. Since people did not know any command that MAHII provides, so they used normal behavioral commands e.g. to delete a figure they just crossed over it. So, MAHII proves to be successful as natural, simple and user friendly.

Then we tried to find out how it encourages the computer known people. They were also asked to draw sketches from various domains. Our system proved successful for them as well. They sketched strokes and they found it easy to use the menu based commands since the menu based commands were based on buttons that were labeled with the corresponding images rather than the command name. MAHII was successful for expert users as well in all respects.

Table 1 shows our efficiency measurement in terms of ease of use.

USER	Efficiency in Terms of Ease of Use
Novice	93%
Expert	94%

Table 1: Efficiency of MAHII for various users

8. RELATED WORK AND COMPARISON STUDY

In this section we compare the efficiency and effectiveness of the proposed system with other existing systems defined for sketching. In Table 2 we have given the recognition rate of MAHII for different inputs, and domains.

Comparison with previous work shows that MAHII improves the sketching, editing and recognition. This shows its capability better than the earlier systems.

Input Type	Static	97%
	Dynamic	95%
Domain	Single Domain	96%
	Multi Domain	87%

Table 2: Recognition rate

Various sketch systems like discussed as in paper [11] have shown the efficiency (6.5 / 7) of their system to 92.8%. CALI [10] gives recognition efficiency up to 95.3% for an overall recognition rate. The recognition rate for static inputs has been improved to 95.1%, but for dynamic input, it is limited up to 70% only. Bayesian approach [9] that gives the accuracy rate till 94% is based on the probability. Whereas using the Fuzzy Logic approach the efficiency is up to 92% for a single domain system and not much work has been done to increase the efficiency of multi-domain sketching system [8]. However, MAHII gives 87% efficiency for multi-domain sketching.

Further, for the purpose of various comparison aspects we consider here the existing systems CAD, SkRUI, SketchRead, FIRST and depict the corresponding graphs in Figures 10 and 11.

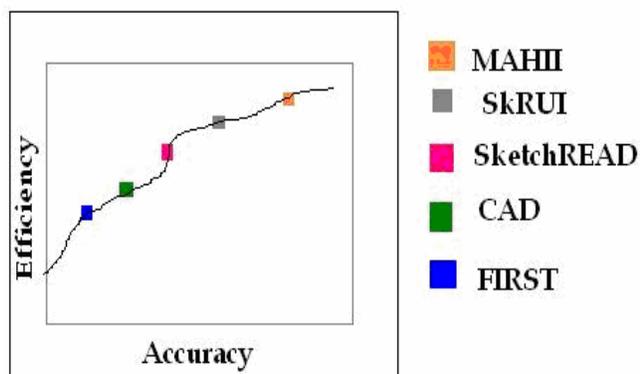
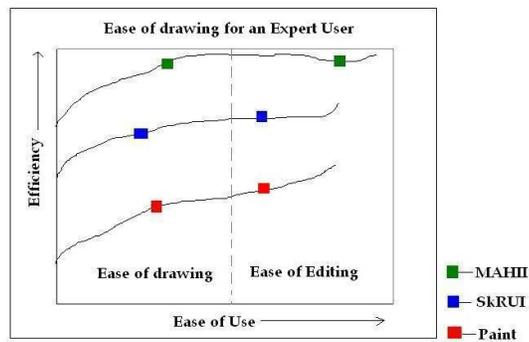
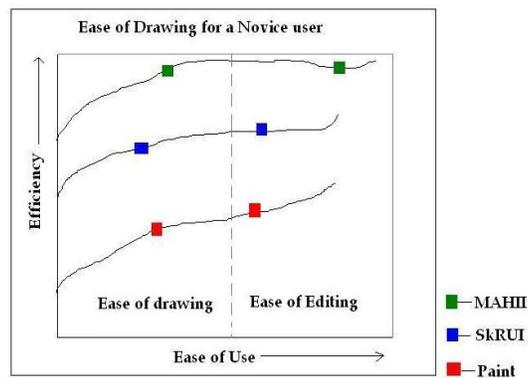


FIGURE 10: The comparison graph respect to sketching, editing and recognition



(a)



(b)

FIGURE 11: Efficiency curves respect to easy of drawing for (a) Expert user, (b) Novice user

From the above Table and Figures we can say that the system MAHII has a capability in sketching, editing and recognition over the existing systems. It provides better results with a higher rate of usefulness: its automatic editing feature makes it a perfect sketching system.

9. CONCLUSION AND FUTURE WORK

MAHII interface presented here is a new system that exhibits the powerful yet more natural environment for the user. It takes care about the every kind of user and the people of various domains. Users need not to learn computer to use this system. It reduces the complexity still showing more freedom to the user in drawing the sketches.

MAHII satisfies the requirements of novice user and it can further be enhanced to provide more real time responding and friendly environment. It may use large amount of buffer area, but satisfies the demand of multi domain recognition without the knowledge of user. This work can be carried out later.

10. REFERENCES:

1. G. Sahoo, Bhupesh Kumar Singh, "A New Approach to Sketch Recognition using Heuristic", International Journal of Computer Science and Network Security, Vol. 8 No. 2, 2008., PP. 102-108.,
2. Mankoff, J. Hudson, S.E., and Abowed. G. D., "Providing integrated toolkit-level support for ambiguity in recognition-based interfaces". In proceedings of CHI 2000 conference on Human Factors in Computing Systems, 368-375.

3. Newman. M.W., Lin. J., Hong. J. I., and Landay. J. A., "DENIM: An Informal web site design tool inspired by observations of practice." *Human-Computer Interaction* 18(3): 259-324
4. Alvarado C., "Sketch Recognition And User Interfaces: Guidelines for Design and Development", AAI Fall Symposium on Pen-Based Interaction, 2004.
5. Landay. J.A., and Myres, "Interactive Sketching for the early stages of user interface design". In proceedings of CHI 95, 43-50. B.A.1995
6. Mahoney. J. V., and Fromherz. M. P. J., "Three main concerns in Sketch recognition and an approach to addressing them". AAI Spring Symposium on Sketch Understanding, 2002.
7. Rubine D. "Specifying gesture by example". SIGGRAPH 1991, Las Vegas, pp.329-327
8. Fonseca. M., Jorge. J., "Using Fuzzy Logic to recognize geometric shapes interactively". Proceedings of the 9th IEEE conference on Fuzzy systems, pp. 291-296, 2000.
9. Shilman. M., Pasula. H., Russell. S., and Newton. R. "Statistical visual language models for ink parsing." Proceedings of AAI spring Symposium on Sketch Understanding, Stanford University, pp. 126-132, March 2002.
10. Fonseca. M.' Pimetal. C., "CALL: an online scribble recognizer for calligraphic interface". Proceedings of AAI Spring Symposium on sketch Understanding, Stanford University, PP.51-58, March 2002.
11. Wolin. A., Smith. D., and Alvarado. C., "A Pen—based Tool for efficient labeling of 2D sketching", EUROGRAPHICS workshop on sketch based Interfaces and Modeling (2007).
12. Alvarado. C. and Oltmans. M., Davis. R.' "A Framework for multi-domain sketch recognition". Proceedings of AAI Spring Symposium on sketch Understandings, Stanford University, pp.1-8, March 2002.
13. Alvarado. C.' "Multi-domain sketch understanding" PhD thesis, Department of Electrical Engineering and computer science, Massachusetts Institute of Technology, September 2004.