

**Towards sketch-based user interface:
domain-independent online sketch recognition system**

by

Abdelrahman Hassan Ali Fahmy Abdalla

A Thesis Submitted to the
Faculty of Engineering at Cairo University
In Partial Fulfillment of the Requirements for the Degree of

MASTER OF SCIENCE
in
COMPUTER ENGINEERING

**FACULTY OF ENGINEERING, CAIRO UNIVERSITY
GIZA, EGYPT**

2015

Towards sketch-based user interface: domain-independent online sketch recognition system

by

Abdelrahman Hassan Ali Fahmy Abdalla

A Thesis Submitted to the
Faculty of Engineering at Cairo University
In Partial Fulfillment of the Requirements for the Degree of

MASTER OF SCIENCE
in
COMPUTER ENGINEERING

Under the Supervision of

Prof. Dr. Amir F. Atiya

Computer Engineering Department
Cairo University

**FACULTY OF ENGINEERING, CAIRO UNIVERSITY
GIZA, EGYPT**

2015

Towards sketch-based user interface: domain-independent online sketch recognition system

by

Abdelrahman Hassan Ali Fahmy Abdalla

A Thesis Submitted to the

Faculty of Engineering at Cairo University

In Partial Fulfillment of the Requirements for the Degree of

MASTER OF SCIENCE

in

COMPUTER ENGINEERING

Approved by the examining committee:

Prof. Dr. Amir F. Atiya,
Faculty of Engineering - Cairo University

Thesis Main Advisor

Prof. Dr. Magda B. Fayek,
Faculty of Engineering - Cairo University

Internal Examiner

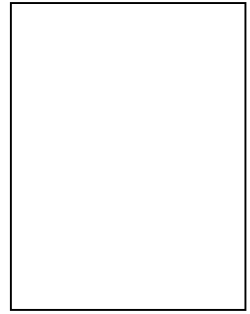
Prof. Dr. Mohamed Waleed Fakhr,
College of Computing and Information Technology - AAST

External Examiner

**FACULTY OF ENGINEERING, CAIRO UNIVERSITY
GIZA, EGYPT**

2015

Engineer: Abdelrahman Hassan Ali Fahmy Abdalla
Date of Birth: 29 / 04 / 1985
Nationality: Egyptian
E-mail: abdelrahman.hasan@gmail.com
Phone. : 01095187636
Address: 33 Rawda square, el-manial street, Masr el Kadema.
Registration Date: 01 / 10 / 2009
Awarding Date: / /
Degree: Masters of Science
Department: Computer Engineering



Supervisors: Prof. Dr. Amir F. Atiya

Examiners: Prof. Dr. Amir F. Atiya (Cairo University)
Prof. Dr. Magda B. Fayek (Cairo University)
Prof. Dr. Mohamed Waleed Fakhri (Arab Academy for Science, Technology, & Maritime Transport)

Title of Thesis:

Towards sketch-based user interface: domain-independent online sketch recognition system

Keywords:

Sketch Recognition, Machine Learning, Human-Computer Interaction

Summary:

With the recent advances in tablet devices industry, sketch recognition has become a potential replacement for existing systems' traditional user interfaces. Structured diagrams (flow charts, Markov chains, module dependency diagrams, state diagrams, block diagrams, UML, graphs, etc.) are very common in many science fields. Usually, such diagrams are created using structured graphics editors like Microsoft Visio. Structured graphics editors are extremely powerful and expressive, but they can be cumbersome to use. This thesis presents a domain-independent sketch recognition system that converts user's sketch into structured shapes in usable electronic format with minimal effort. It supports a wide set of primitives shapes that are common between many structured diagrams. It will allow users to sketch block diagrams, flow charts, etc., using their tablet devices in a very user-friendly and constraints-free manner.

Acknowledgments

All thanks and praise go to ALLAH who gave me the strength that enabled me finish this work. I am much grateful to my supervisor, Dr. Amir Atiya, who did not spare any effort to help me. He supported me all the way. He was patient and supportive, and provided many invaluable pieces of advice. I am also much grateful to Dr. Shafeka El-Sayed, Amira Barakat, Wael Nabil, Alaa Nabil, Ahmed Emam, Doaa Hassan, Heba Mostafa, and Mai Ali for their help in creating training set and test set used to evaluate the system. Finally, this work is dedicated to my family for their complete support and prayers.

Contents

Acknowledgments	i
Contents	ii
List of Tables	v
List of Figures	vi
List of Algorithms	viii
List of Abbreviations	ix
Abstract	x
1 Introduction	1
2 Sketch recognition problem	3
2.1 Introduction	3
2.2 Low-level sketch recognition	3
2.3 High-level sketch recognition	3
2.4 Recognition of multi-stroke primitives	4
2.5 Sketch recognition system interface	5
2.5.1 Online	5
2.5.2 Offline	5
2.6 Conclusion	5
3 Literature survey	6
3.1 Offline Sketch recognition systems	6
3.1.1 Sketch recognition by corners removal	6
3.1.2 User-Directed Sketch Interpretation (UDSI)	7
3.1.3 Context-independent handwritten offline diagram recognition using support vector machines	9
3.2 Online sketch recognition systems	10
3.2.1 Sketched single-stroke primitives recognition	10
3.2.1.1 Sketch recognition based on incremental intention extraction	10
3.2.1.2 PaleoSketch: Accurate primitive sketch recognition and beautification	11
3.2.1.3 template-based sketch recognition	13

3.2.2	Sketched multi-stroke primitives recognition	14
3.2.2.1	Multi-stroke support based on spatial properties	14
3.2.2.2	Multi-stroke support based on timeout	15
3.2.2.3	Multi-stroke support based on segmentation	15
3.2.2.4	Multi-stroke support based on template matching	18
3.2.3	Sketch recognition using hypothesis-based concept and Bayesian network	18
3.2.4	Sketch recognition on intelligent surfaceless sketching user interfaces	20
3.3	Sketch Recognition of digital logic circuits	22
3.4	Context aware primitives recognition	22
3.5	Rule-based sketch recognition	23
3.6	HMM-based efficient sketch recognition	24
3.7	Other sketch recognition applications	24
4	Classifiers background	26
4.1	Introduction	26
4.2	Support Vector Machine (SVM)	26
4.2.1	Soft margins	28
4.3	k -Nearest Neighbors (k -NN)	29
4.4	Random Forests (RF)	30
5	The proposed system: domain-independent online sketch recognition system	32
5.1	Introduction	32
5.2	System architecture	32
5.2.1	Input stage	32
5.2.2	Grouping	33
5.2.2.1	Euclidean distance grouping	34
5.2.3	Classification preprocessing	36
5.2.4	Features extraction	37
5.2.5	Classification	39
5.2.6	System output	39
5.2.7	Squares classification	41
5.3	Domain-dependent sketch recognition systems	42
5.3.1	System input	42
5.3.2	Grouping	43
5.3.3	Recognition	43
5.3.4	System output	43
5.3.4.1	Beautification	43
5.3.4.2	Output format	44
6	Experimental Results	45
6.1	Discussion	50
6.2	Grouping output samples	51
6.3	Sample test cases	53

7	Conclusion and Future Work	56
7.1	Conclusion	56
7.2	Contributions	56
7.3	Future work	57
	References	58
A	Geometric terms background	63
B	Utilities algorithms	65
B.1	Sort input points in circular order	65
B.2	Strokes grouping algorithm	65
B.3	Supported primitives drawing algorithms	66
B.3.1	Arrow drawing algorithm	67
B.3.2	Diamond drawing algorithm	68
C	Framework usage	69
C.1	Training phase	69
C.2	Running phase	72

List of Tables

3.1	Result's confusion matrix	23
6.1	Open shapes training set	45
6.2	Closed shapes training set	45
6.3	10-fold cross validation results	47
6.4	Open shapes test set	47
6.5	Closed shapes test set	47
6.6	testing results	48
6.7	SVM closed shapes classifier confusion matrix	48
6.8	SVM open shapes classifier confusion matrix	49
6.9	RF closed shapes classifier confusion matrix	49
6.10	RF open shapes classifier confusion matrix	49
6.11	K-NN closed shapes classifier confusion matrix	49
6.12	K-NN open shapes classifier confusion matrix	49
6.13	Recognition accuracy comparison of widely used sketch recognition sys- tems	50
6.14	Our system and PaleoSketch recognition accuracy comparison	51

List of Figures

1.1	Examples of sketch usage	1
2.1	Distinction in labeling between low-level and high-level recognition . . .	4
2.2	Sketching multi-stroke circle	4
2.3	Different ways of drawing arrow, 3 out of 4 ways will use multi-stroke . .	4
3.1	Algorithm steps	8
3.2	UDSI recognition accuracy	9
3.3	Flow chart for incremental intention extraction algorithm	11
3.4	PaleoSketch system overview	11
3.5	User input	12
3.6	Rank of 4 (1 (line) + 3 (Arc) = 4)	12
3.7	Rank of 8 (1(line) + 8 (polyline made up of 7 lines) = 8)	12
3.8	\$1 supported gestures	13
3.9	Strokes grouping example	14
3.10	Dominant points	16
3.11	Segmentation	16
3.12	Algorithms accuracies comparison	17
3.13	HHrec symbol set	17
3.14	System overview	18
3.15	hypothesis-based concept example	19
3.16	Family tree patterns	20
3.17	Family tree use case	20
3.18	Types of surfaceless sketching, (a,b) mid-air sketching (c,d) above sur- face sketching	21
3.19	Their Preliminary recognition results for primitive geometric shapes cat- egorized as either polylines or curvatures	22
3.20	LADDER System Description	23
3.21	System Overview	25
3.22	System output example	25
4.1	Training set samples for two classes	27
4.2	non-linear classifier for the two classes	27
4.3	Mapping to higher dimensional space	28
4.4	Linearly separated two classes: 'x' and 'o'	28
4.5	decision boundary margin and support vectors	29
4.6	k-NN example	30
4.7	Random forests	31
5.1	System stages overview	33

5.2	Multi-stroke sketched rectangle	34
5.3	Multi-stroke sketched triangle	34
5.4	Euclidean grouping using graphs	35
5.5	Supported ways of drawing arrows	36
5.6	Ways of drawing arrows which may confuse grouping technique	36
5.7	Line editing example	37
5.8	Thinness feature	38
5.9	Straightness feature	38
5.10	Intersection with regression line	38
5.11	Using Casteljau's algorithm for arc drawing	40
5.12	Framework maintains sketched primitives geometric properties	41
5.13	Shapes orientation examples	41
5.14	Domain-dependent sketch recognition system generic architecture	42
5.15	Sample high-level shapes	43
5.16	Recognized high-level shapes beautification examples	44
6.1	Expected system out for each supported primitive	46
6.2	Comparison between the accuracy results of our framework (FahmySVM) with widely used sketch recognition systems	50
6.3	Test cases for triangles that are drawn using multiple strokes	51
6.4	Test case for a rectangle that is drawn over two strokes	52
6.5	Multi-stroke circle test case	52
6.6	Test case for line that was edited to diamond	52
6.7	Multi-stroke ellipse test case	52
6.8	Test case 1	53
6.9	Test case 2	54
6.10	Test case 3	55
6.11	Test case 4	55
A.1	Convex Hull	63
A.2	(a) Aligned minimum area enclosing rectangle (b) Non-aligned minimum area enclosing rectangle	64
A.3	Largest area quadrilateral/triangle	64
B.1	Arrow points	67
B.2	Diamond vertices	68
C.1	Console application screen shot	69
C.2	Sample image for training set pattern	70
C.3	Training set folder organization example	70
C.4	Default classifiers folders	70
C.5	Train classifier	71
C.6	Test classifier accuracy	72
C.7	Load models	73
C.8	Models folders structure	74

List of Algorithms

B.1	Sort points in circular order by their polar angel	65
B.2	Strokes grouping algorithm	66
B.3	Arrow drawing algorithm	67
B.4	Diamond drawing algorithm	68

List of Abbreviations

A	Area
API	Application Programming Interface
CGAL	Computational Geometry Algorithms Library
ch	Convex Hull
DPSO	Discrete Particle Swarm Algorithm
er	Enclosing Rectangle
H	Height
HMM	Hidden Markov Models
K-NN	K-Nearest Neighbors
lq	Largest Quadrilateral
lt	Largest Triangle
MMD	Minimum Mean Distance
NN	Neural Networks
P	Perimeter
PSO	Particle Swarm Optimization Algorithm
RF	Random Forest
SVG	Scalable Vector Graphics
SVM	Support Vector Machine
UML	Unified Modeling Language
W	Weight

Abstract

With the recent advances in tablet devices industry, sketch recognition has become a potential replacement for existing systems' traditional user interfaces. Structured diagrams (flow charts, Markov chains, module dependency diagrams, state diagrams, block diagrams, UML, graphs, etc.) are very common in many science fields. Usually, such diagrams are created using structured graphics editors like Microsoft Visio. Structured graphics editors are extremely powerful and expressive, but they can be cumbersome to use. This thesis presents a domain-independent sketch recognition system that converts user's sketch into structured shapes in usable electronic format with minimal effort. We have presented a robust grouping technique which is based on spatial distance between strokes. We have also introduced the concept of classification preprocessing which is proved to minimize classifiers confusion by doing a precategorization to the shapes. We have compared the accuracies of three classifiers; Support vector machines (SVMs), Random Forest (RF), and K- Nearest neighbor (K-NN). We have also compared the proposed system to benchmark low-level recognizers, and the proposed system proved very competitive accuracy. The proposed system supports a wide set of primitives shapes that are common between many structured diagrams. It will allow users to sketch block diagrams, flow charts, etc., using their tablet devices in a very user-friendly and constraints-free manner. One of the main usages of the proposed system is to be integrated with other domain-dependent sketch recognition systems. The presented system will recognize generic primitives in the early stages and then it will be the task of the domain-dependent system to map these primitives to their corresponding high-level shapes in a specific domain. As an use case, this system will be integrated with a system presented in [48], to test it as a part of domain-dependent sketch recognition system with two applications: GUI design and circuits/electronics diagrams sketching applications.

Chapter 1

Introduction

Sketching is considered as one of the most intuitive languages for expressing human thoughts. One simple sketch can express what thousands of words can. We can use a sketch to draw a scene, show a flow, put a design, etc. Sketches are widely used in many fields of science as a unified language for explaining theories and designs. Thousands of examples for sketching usage can be mentioned. Chemists sketch chemical reaction diagrams and engineers sketch block diagrams, flow charts, class diagrams, circuits design, GUI design, etc. These diagrams are considered as a unified language to explain engineering designs. The majority of engineers prefer to use paper and pencil to draw the diagrams rather than computer software. Meanwhile, it would be more conducive to save these designs in electronic format as it will be much easier to edit and share.

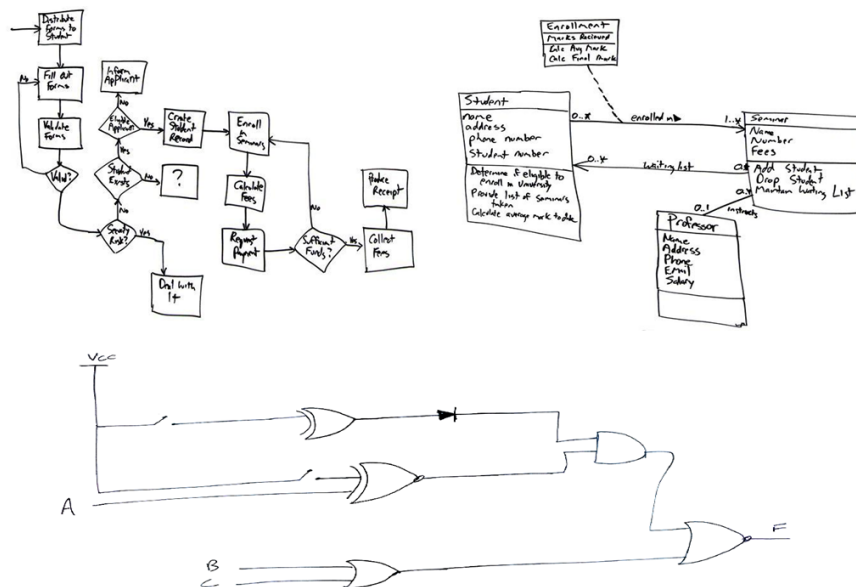


Figure 1.1: Examples of sketch usage

With the rise of tablet devices usage, developing a system that can convert hand-drawn sketches into structured (electronic) format became highly required. It will facilitate sketch creation and modification using Computer Aided Design (CAD) tools. The existing tools mainly use drag and drop feature for diagrams creation. The user constructs the diagram by incrementally selecting from the list of supported shapes. Sketching will provide a more intuitive replacement for this process. In this work, we propose an online