Recognition vs Reverse Engineering in Boolean Concepts Learning

Gabriel Shafat, Prof. Ilya Levin

School of Education, Tel Aviv University

CELDA 2012 Cognition and Exploratory Learning in Digital Age

Outline

- Boolean Concepts
- Cognitive complexity of Boolean Concepts
- Recognition vs. Reverse Engineering
- Method and Experiment
- Conclusions
- Further research

Boolean Concepts

- Logical thinking is the key to a wide variety of complex problem solving and decision making processes and therefore Boolean Concepts are essential

- An important aspect of concept learning theory is the ability to predict the level of difficulty in learning different types of concepts

- Statements can be formally represented using Boolean variables X_i that have either a value of “1” or “0”

- Boolean concepts can be defined by a:

  1. Boolean expression composed of basic logic operations:
     - ✓ negation - “not”
     - ✓ disjunction - “or”
     - ✓ conjunction - “and”

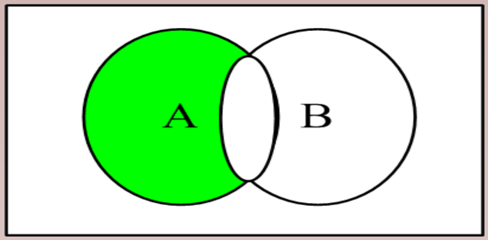
  2. Models :
     - ✓ Karnaugh map
     - ✓ Truth table
     - ✓ Venn diagram
     - ✓ Hasse diagram
     - ✓ Graphical representation
     - ✓ Binary Decision Diagram (BDD)
     - ✓ Switches & Logic Gates

Negation – "not"

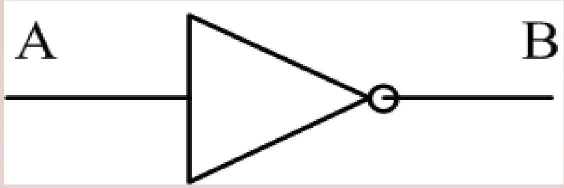
$$B = \overline{A}$$

expression

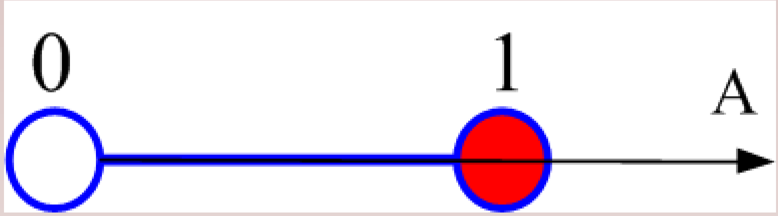
A	B
0	1
1	0



Venn diagram



logic gate



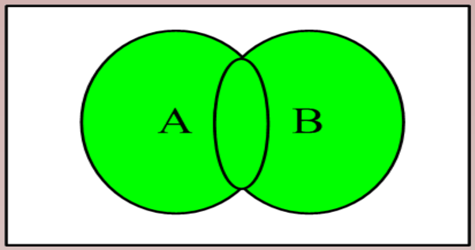
graphical representation

Disjunction - "or"

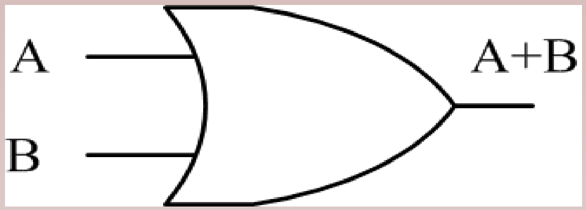
$$A + B = A \vee B$$

expression

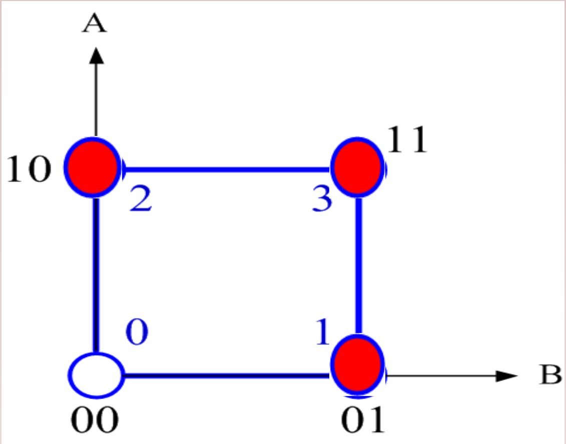
A	B	A+B
0	0	0
0	1	1
1	0	1
1	1	1



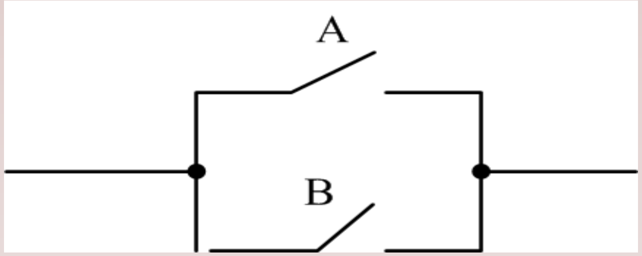
Venn diagram



logic gate



graphical representation

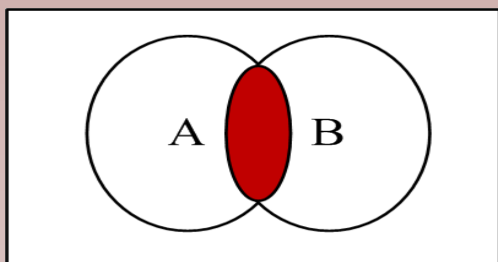


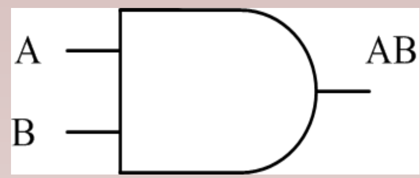
switches

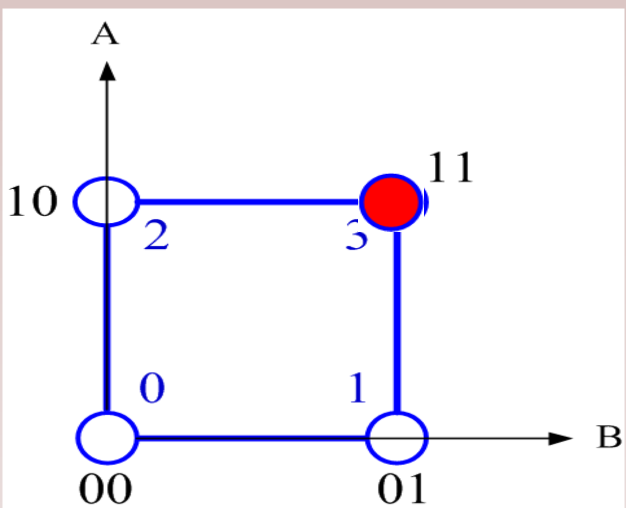
Conjunction -“and”

$$AB = A \wedge B$$

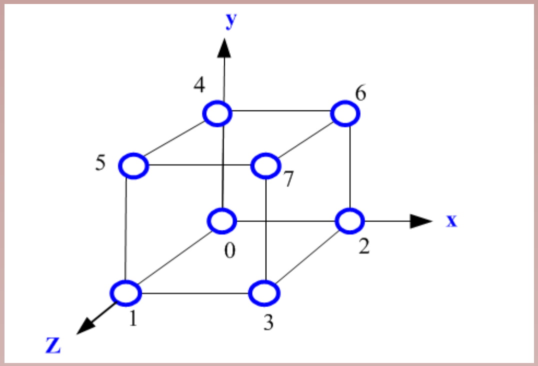
A	B	AB
0	0	0
0	1	0
1	0	0
1	1	1

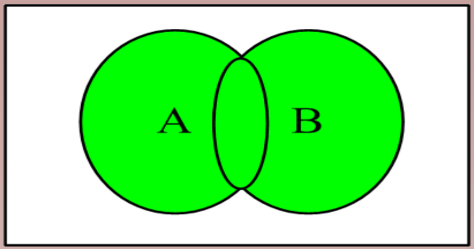


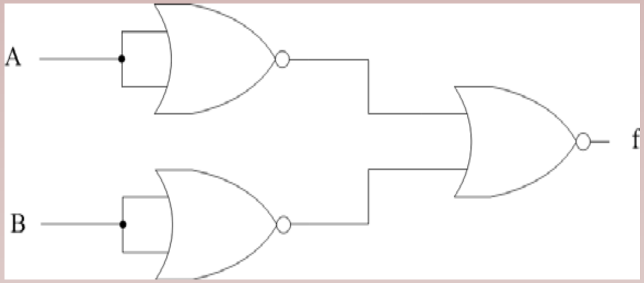


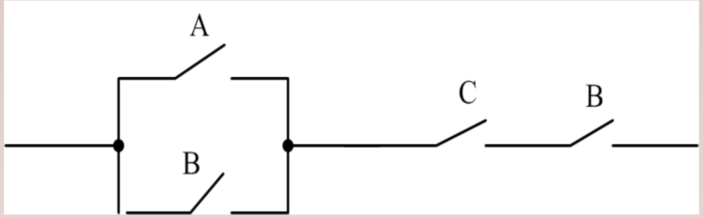


$$f = \bar{x}\,\bar{z} + \bar{y}\,\bar{z} + x\,y\,z$$

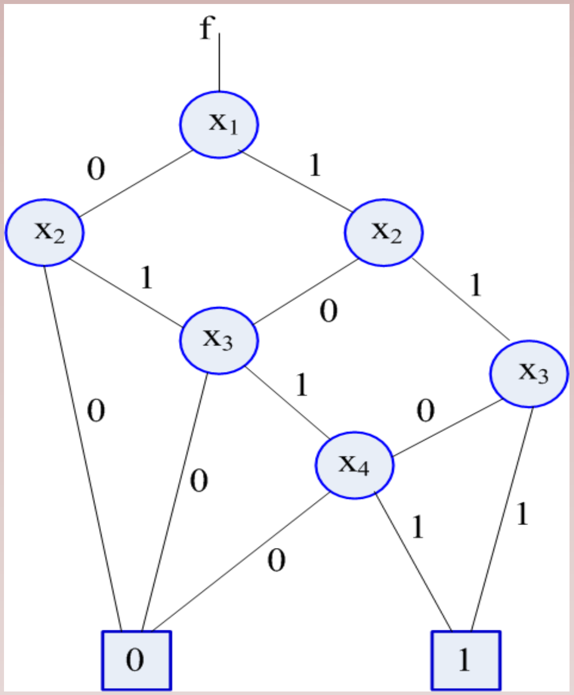








	x	y	z	
0	0	0	0	$\bar{x}\,\bar{y}\,\bar{z}$
1	0	0	1	$\bar{x}\,\bar{y}\,z$
2	0	1	0	$\bar{x}\,y\,\bar{z}$
3	0	1	1	$\bar{x}\,y\,z$
4	1	0	0	$x\,\bar{y}\,\bar{z}$
5	1	0	1	$x\,\bar{y}\,z$
6	1	1	0	$x\,y\,\bar{z}$
7	1	1	1	$x\,y\,z$



Boolean concepts with n binary variables

- In general: for n Boolean variables there are 2^n different combinations.

- For 3 variables there are 8 different combinations.

- Assuming that the Boolean concepts are represented by x, y, z, we can represent all the combinations using the expression:

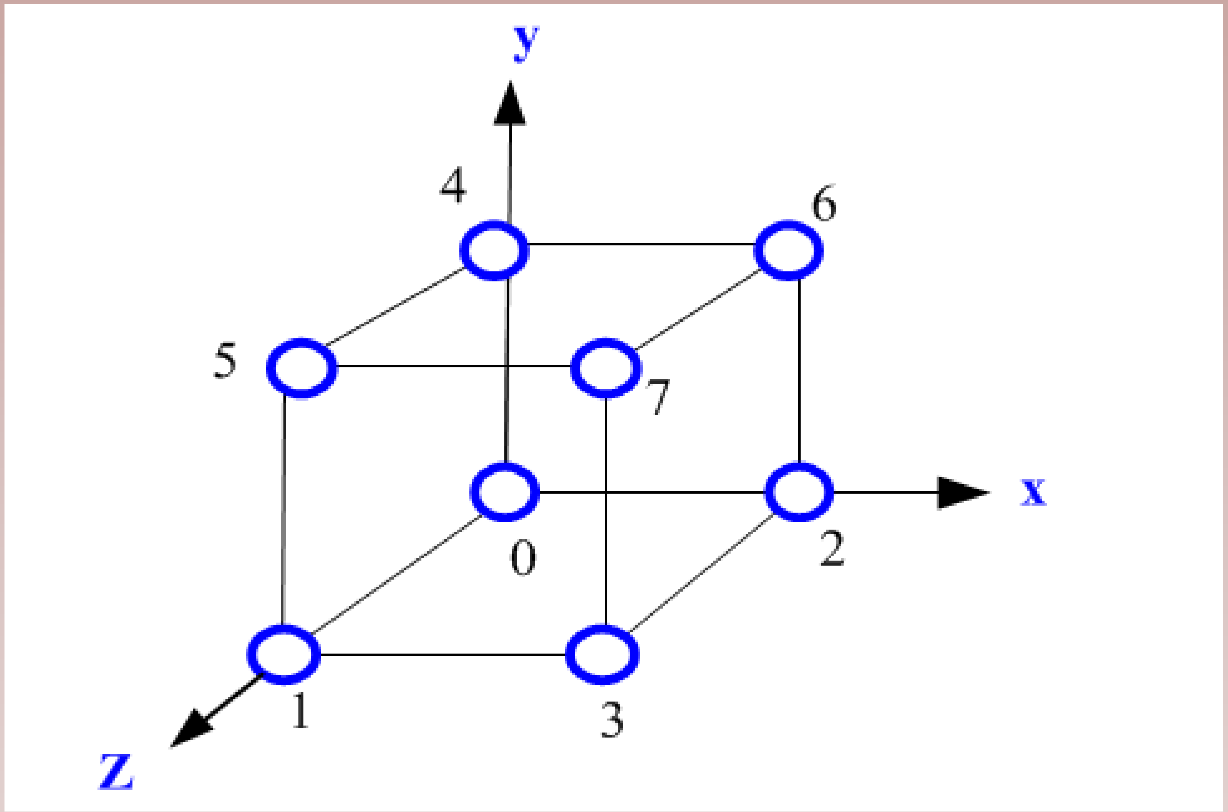
$$\bar{x}\bar{y}\bar{z}+\bar{x}\bar{y}z+\bar{x}y\bar{z}+\bar{x}yz+x\bar{y}\bar{z}+x\bar{y}z+xy\bar{z}+xyz$$

- ✓ Or concurrently we can represent all the combinations as follows, using a truth table

- ✓ The table shows all binary combination and Boolean expression respectively

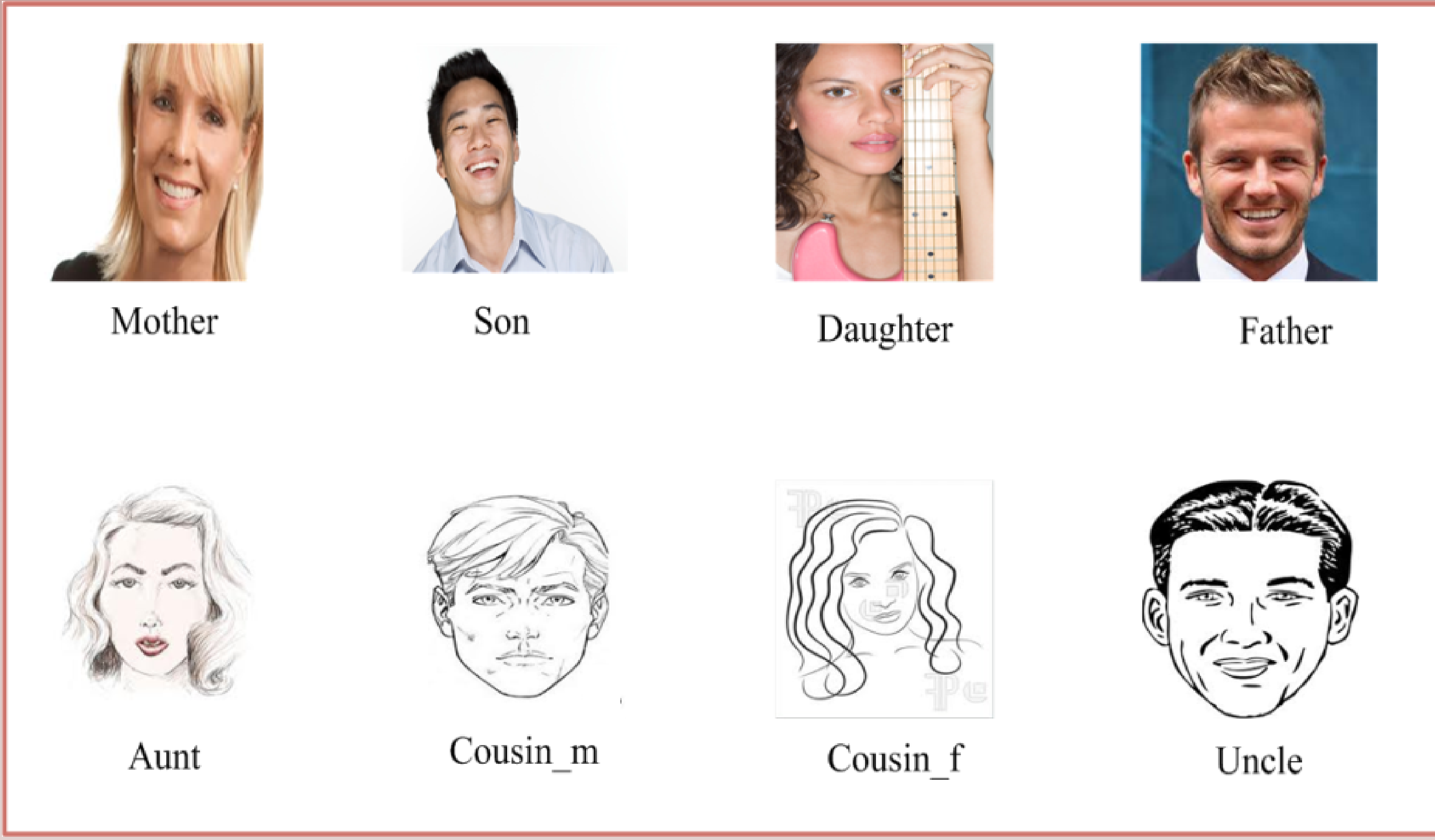
	x	y	z	
0	0	0	0	$\overline{x}\ \overline{y}\ \overline{z}$
1	0	0	1	$\overline{x}\ \overline{y}\ z$
2	0	1	0	$\overline{x}\ y\ \overline{z}$
3	0	1	1	$\overline{x}\ y\ z$
4	1	0	0	$x\ \overline{y}\ \overline{z}$
5	1	0	1	$x\ \overline{y}\ z$
6	1	1	0	$x\ y\ \overline{z}$
7	1	1	1	$x\ y\ z$

- ✓ Or graphically by a cube



- ✓ Each of the vertices in the cube represents one of the combinations respectively

Example 1:



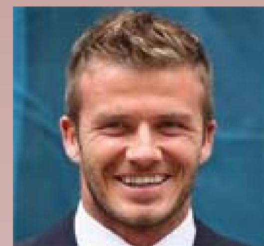
x-parents "0", children "1"
y-male "0", female "1"
z-family "0", kin "1"

000

001

010

011

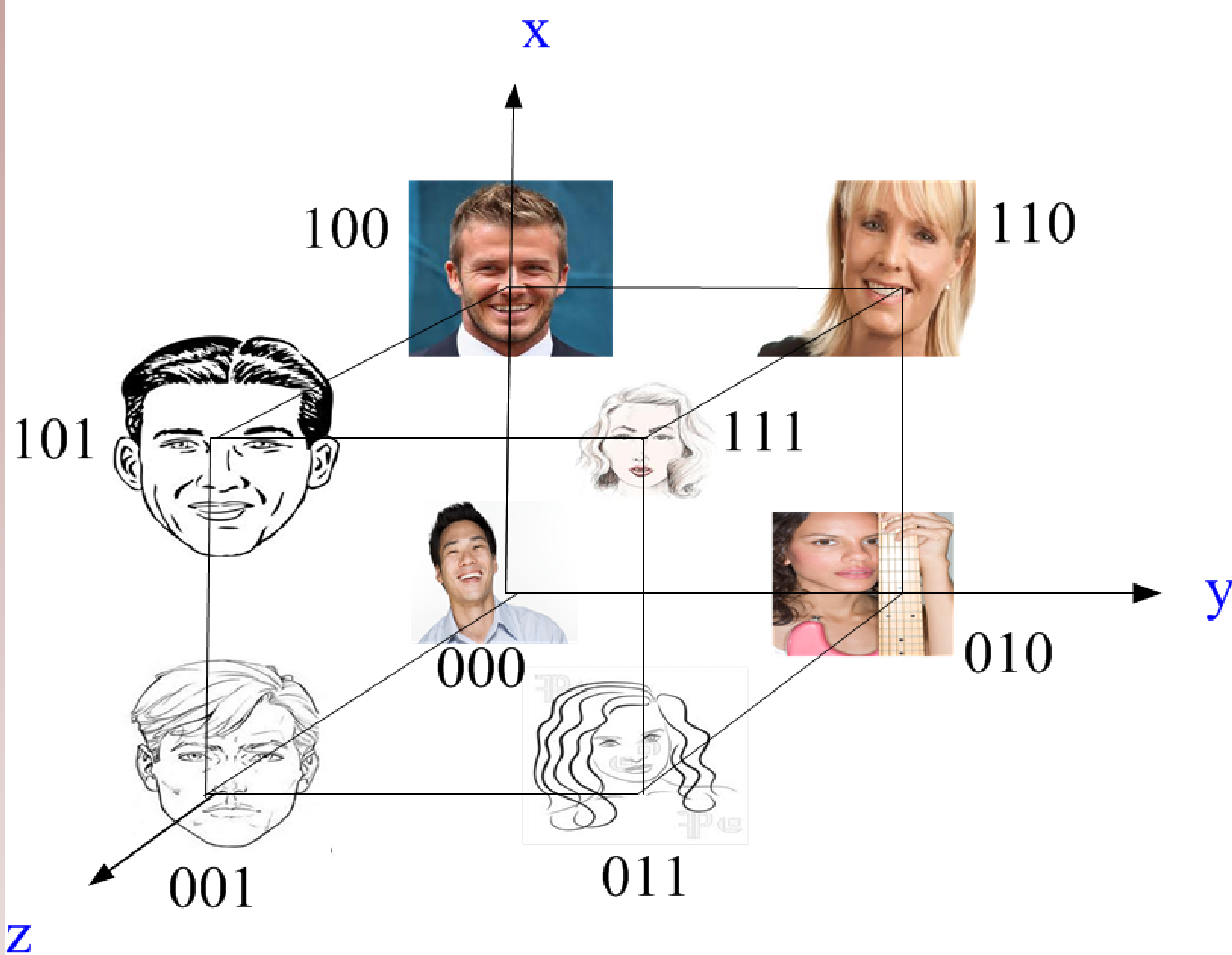


100

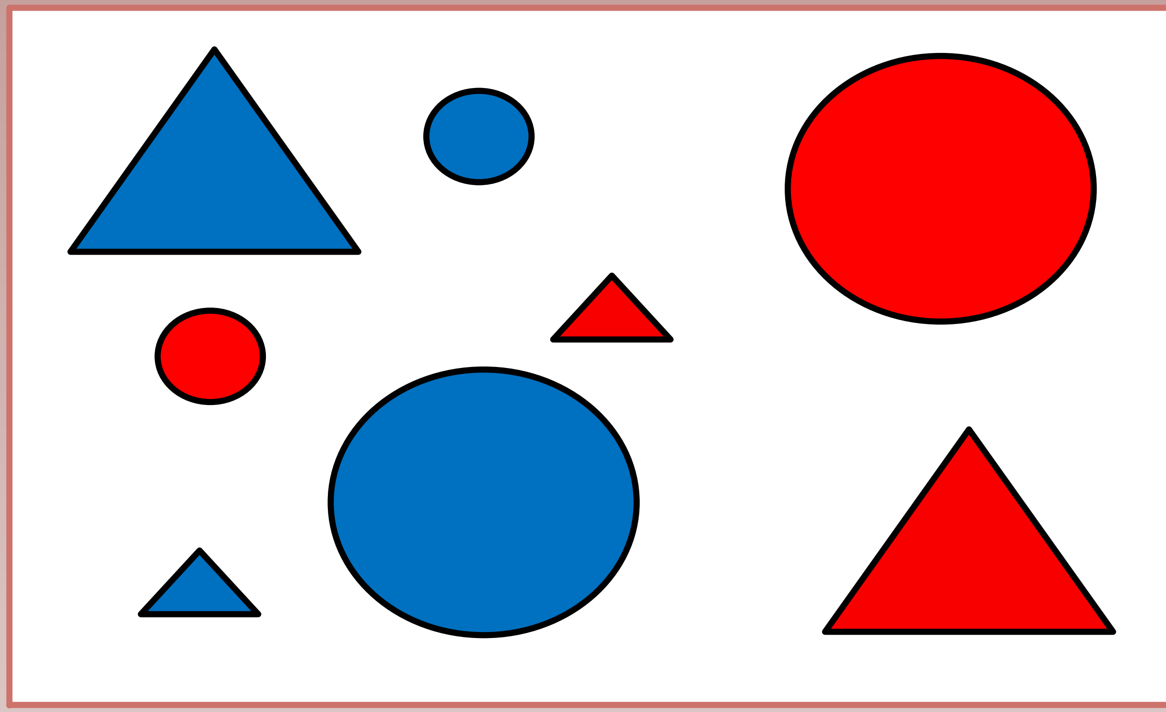
101

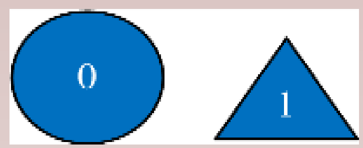
110

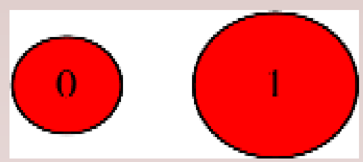
111

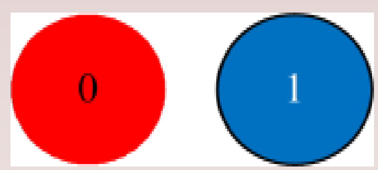

x
100
110
101
111
y
000
010
001
011
z

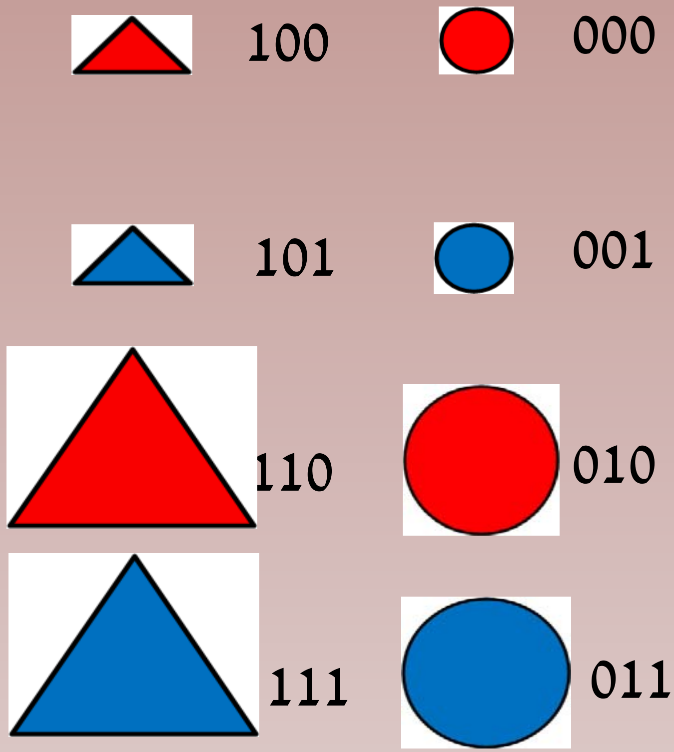
Example 2:



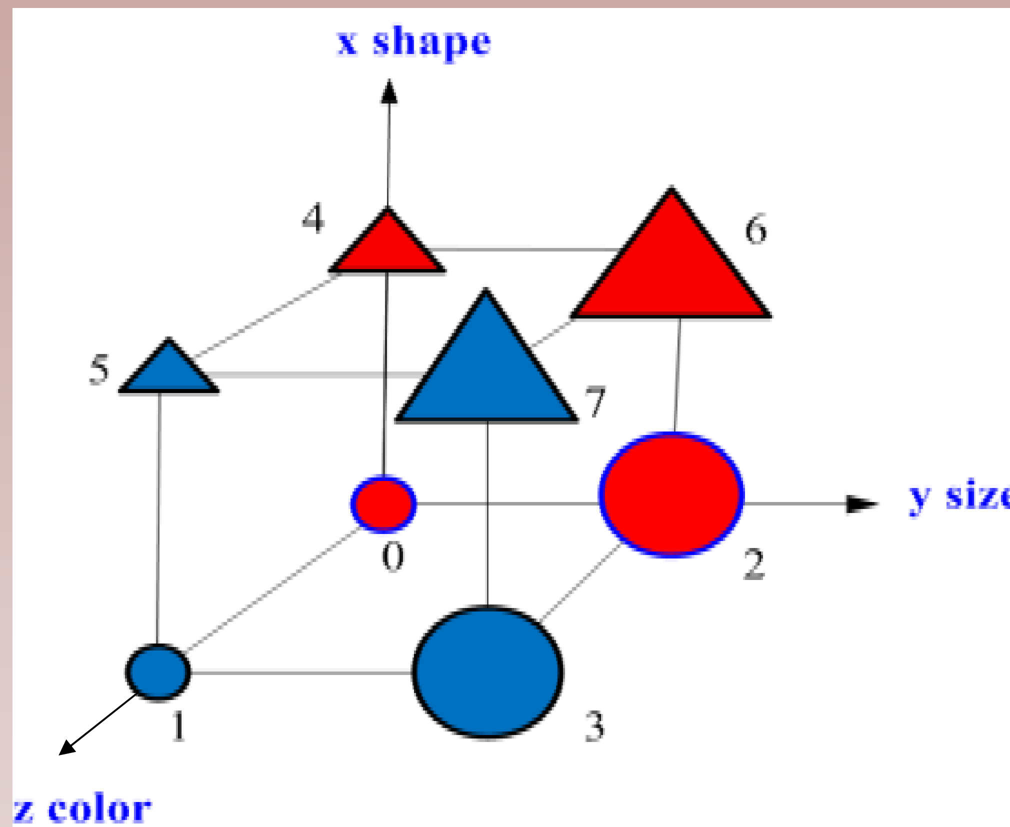
x-shape 

y-size 

z-color 



8 binary combinations



Graphical representation

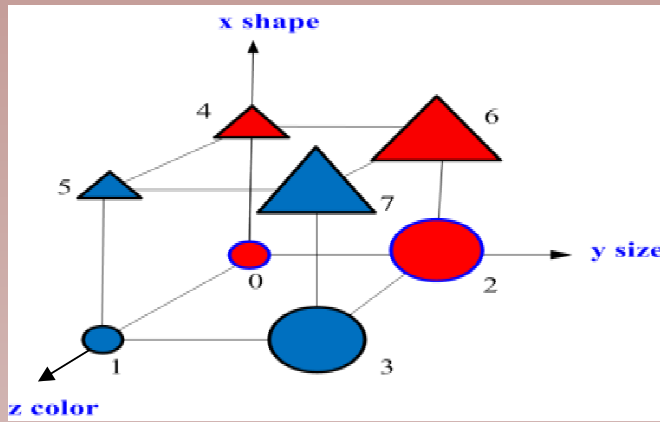
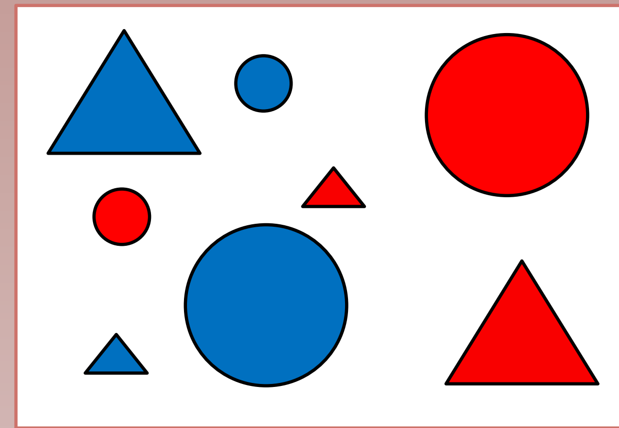
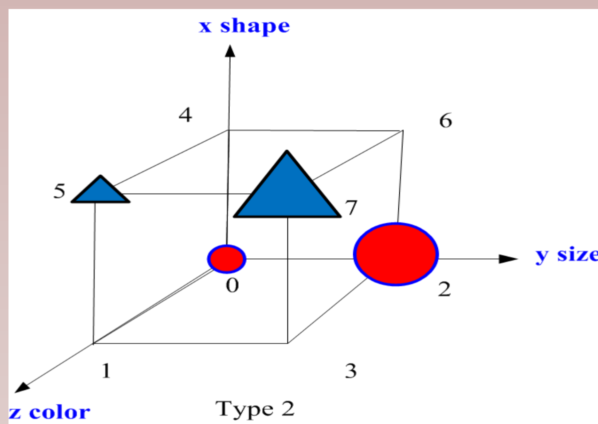
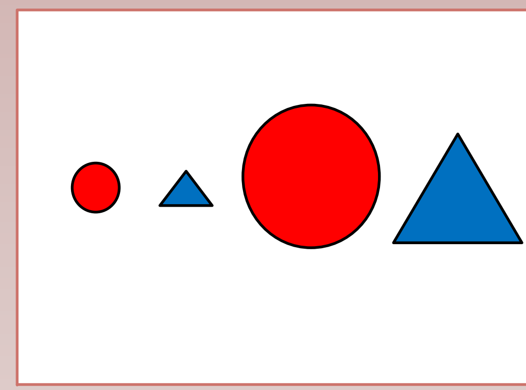
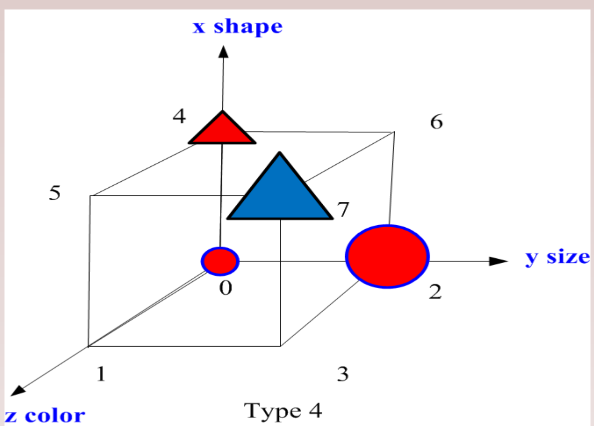
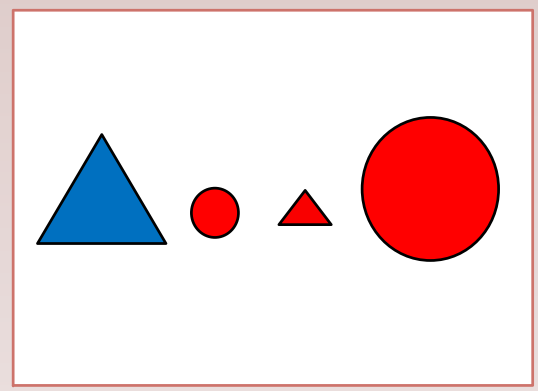
Cognitive complexity of Boolean Concepts

- These types of Boolean concepts have been studied extensively by **Shepard, Hovland, and Jenkins-SHJ** (1961), (Nosofsky, Gluck Palmeri, McKinley, and Glauthier 1994).

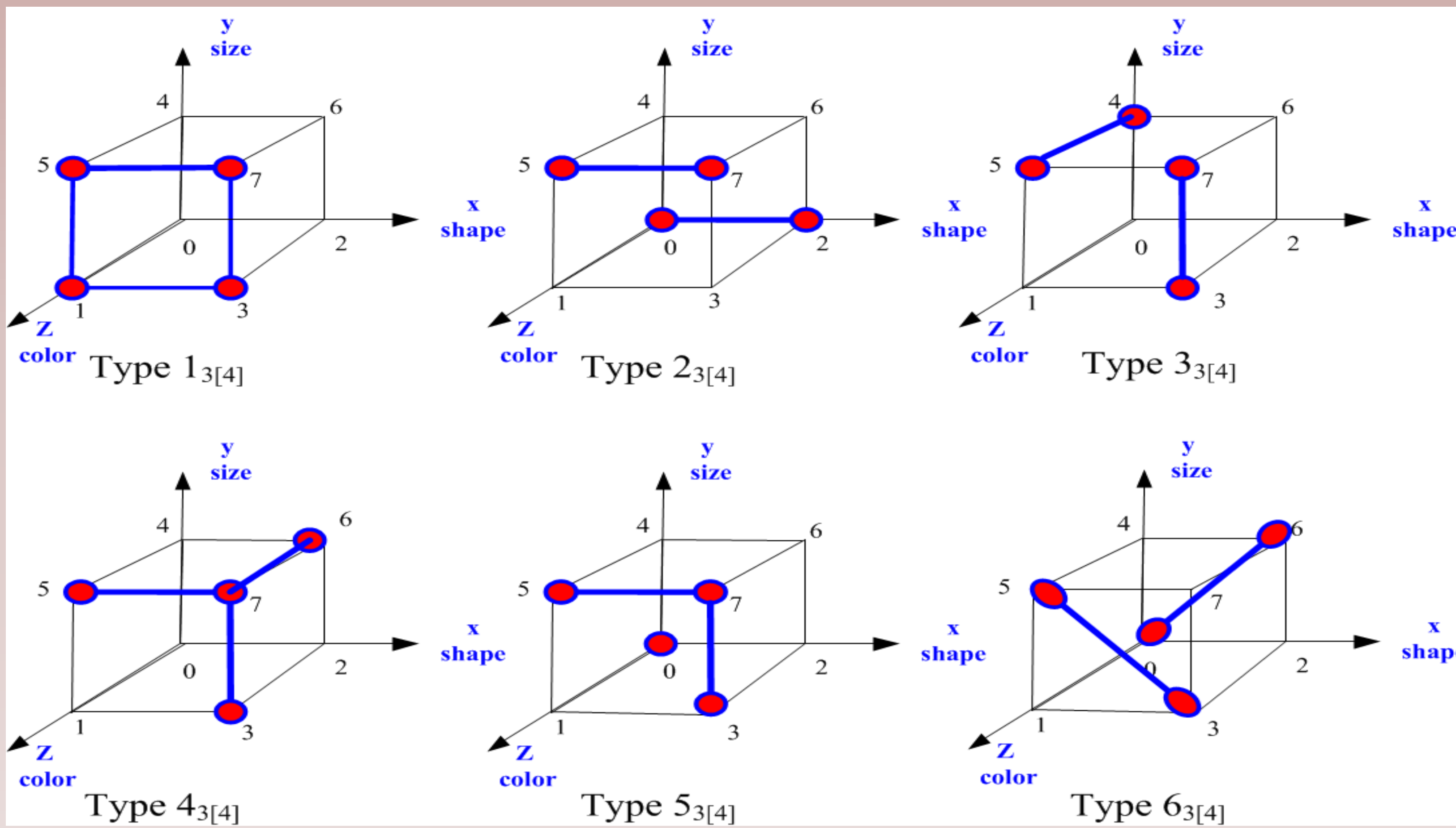
- These studies focused on Boolean concepts with three binary variables, where the concept receives “1” for 4 out of 8 possible combinations and “0” for the remaining 4 combinations.

- There are 70 possible concepts

$$C_4^8 = \frac{8!}{4!(8-4)!} = 70$$


x shape
y size
z color
0
1
2
3
4
5
6
7


x shape
y size
z color
0
1
2
3
4
5
6
7
Type 2


x shape
y size
z color
0
1
2
3
4
5
6
7
Type 4


- Since some of the 70 possible Boolean concepts are congruent, they can be categorized as the same type into six subcategories.
- The six subcategories with structural equivalence can be described in a geometrical representation using cubes



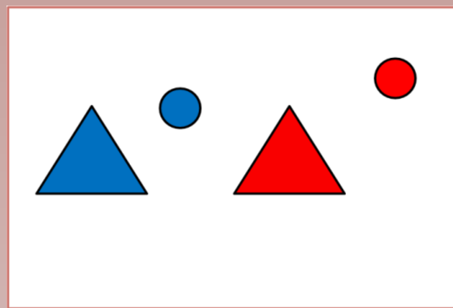
SHJ types graphed in Boolean space

- Type1 are the simplest to learn and the subgroup of concepts belonging to Type6 are the most difficult, according to the following order:
  Type 1<Type2<(Type3=Type4=Type5)<Type6

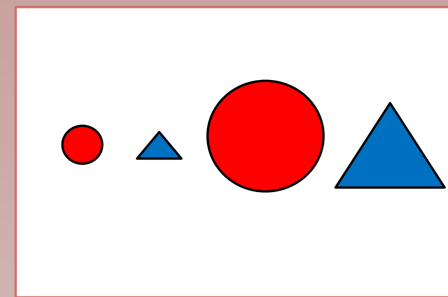
- The result of this study are highly influential since **Shepard at all** proposed two informal hypotheses

  - ✓ *The number of literals in the minimal expression predicts the concept's level of difficulty*

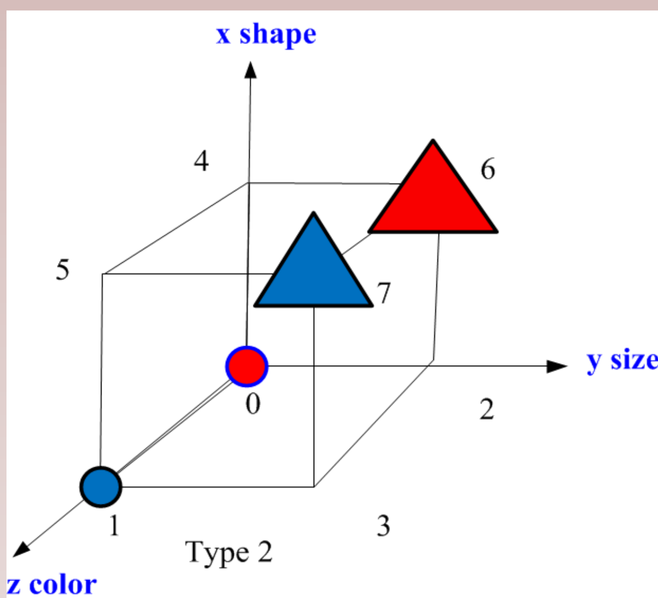
  - ✓ *Ranking the difficulty among the concepts in each type depends on the number of binary variables in the concept*

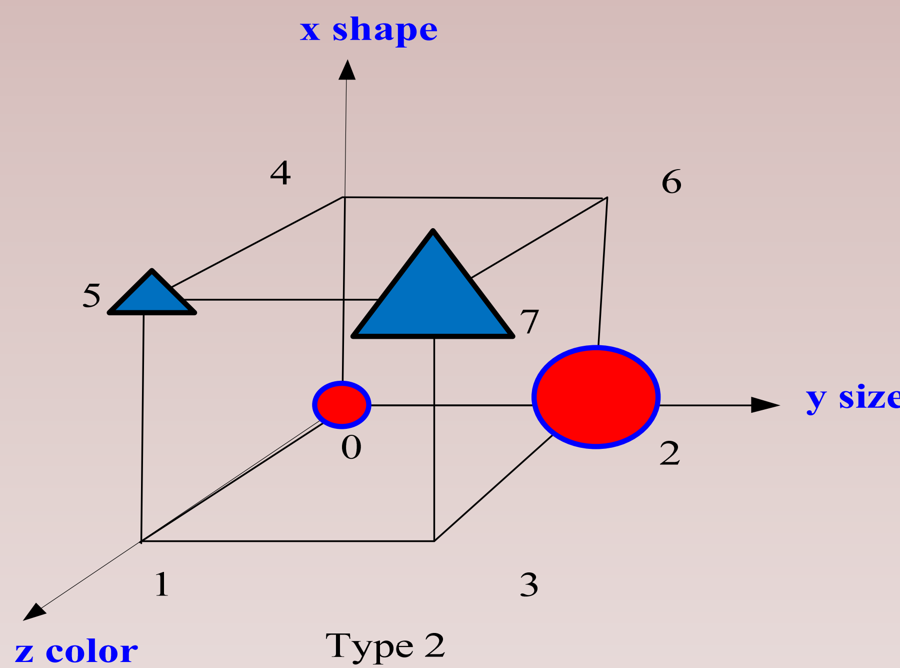


$$\bar{x}\,\bar{y} + xy$$

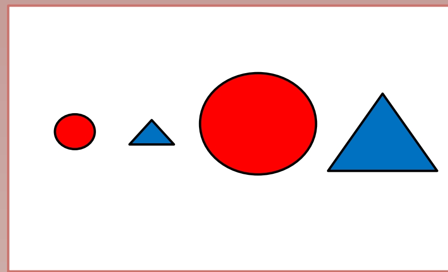


$$\bar{x}\,\bar{z} + x\,z$$

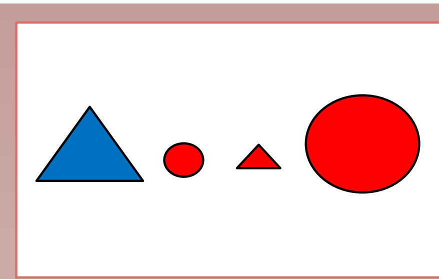




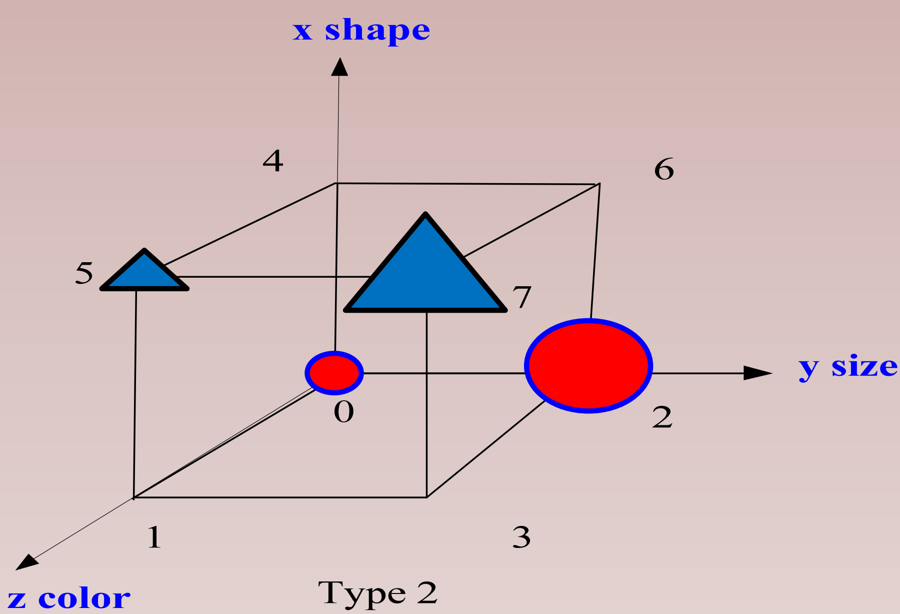
- Feldman (2000) based on the conclusions from the SHJ study, defined a quantitative relationship between the level of difficulty of learning Boolean concepts and the concept's Boolean complexity (Feldman 2000)
- The complexity measure of a Boolean concept as defined by Feldman is the *number of literals in the most minimal expression* that represents the concept's complete *SOP*
- Vigo (2009), developed an alternative theory for calculating the complexity measure of a Boolean concept, defined as *structural complexity*
- The theory is based on a Boolean derivative



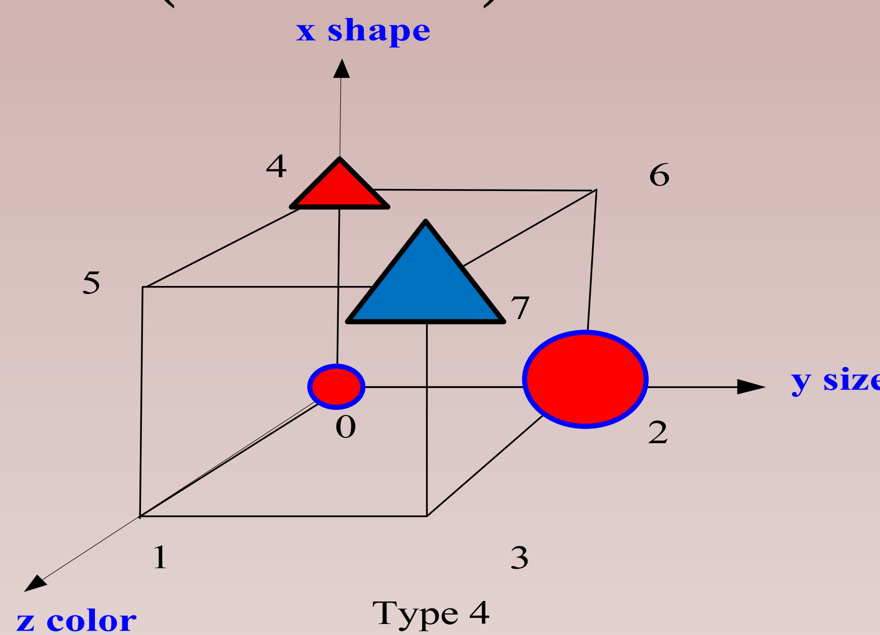
$$\bar{x}\,\bar{z} + x\,z$$



$$\bar{z}\left(\bar{x} + \bar{y}\right) + x\,y\,z$$



Type 2



Type 4

What is more complex for humans?

Feldman's Complexity Issues

- Problem of Boolean minimization (Vigo, 2003)

Feldman's heuristics failed to find the correct minimal descriptions for certain concepts. People cannot really minimize

- Problems of basis

To justify the set of primitive connectives that are allowed in formulating descriptions. A standard defense of not, and, and or, is that they have been conventional since the work of Neisser and Weene (1962). But, why should we exclude XOR?

A	B	A xor B
0	0	0
0	1	1
1	0	1
1	1	0

- Problems of functional properties

Individuals can carry out the task of categorizing instances of a concept without attempting to formulate its minimal description. Simple example: symmetric functions

- Boolean Approximation problems

People usually approximate their solutions

- Representation problems

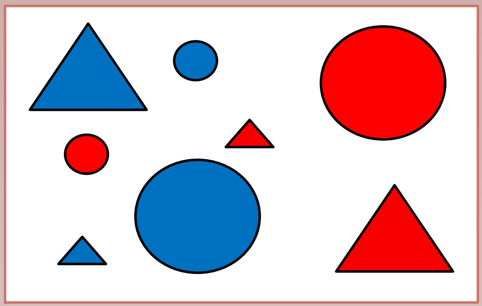
Cubes, sets, diagrammes, BDDs etc.

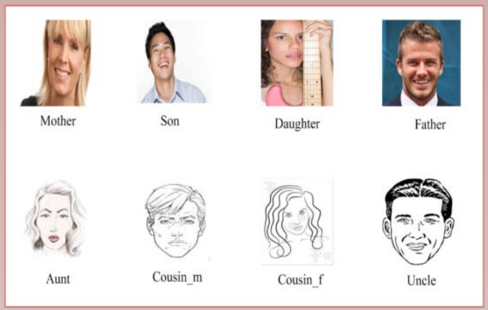
- Different tasks problems

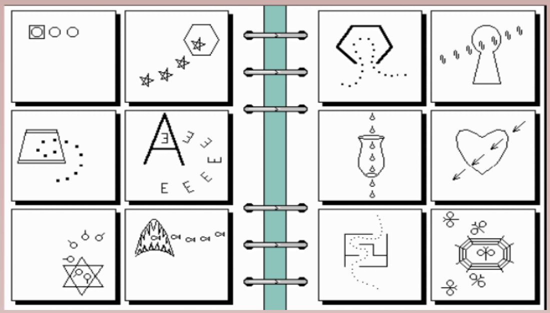
Whether the approach works for different types of tasks. Recognition vs. Reverse Engineering.

Recognition vs. Reverse Engineering

Recognition problems

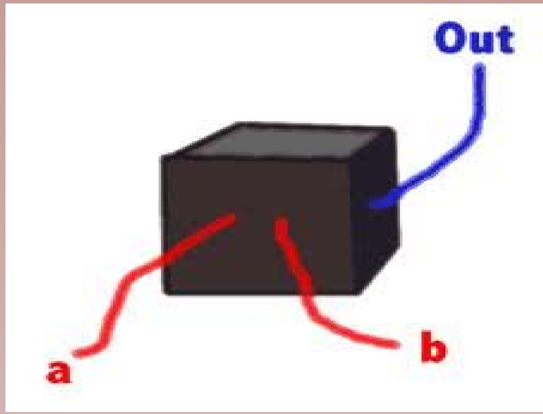






- The recognition problems are modeled in the form of visual representation of various objects in a common pattern, with composition of thus represented objects in the pattern
- Solving of the recognition problem thus may be understood as recognizing a Boolean concept represented visually, with further formulation of the concept
- The recognition problems can be perceived as a parallel process

Reverse Engineering



- The process of finding and reconstructing operating mechanisms in a given functional system of a digital electronic apparatus is defined as Reverse Engineering (Chikofsky & Cross, 1990)
- RE is applied in a wide variety of fields: competition in manufacturing new products, from electronic components to cars, among competing companies without infringing upon the competing company's copyrights
- A reverse engineering problem means reconstructing a Boolean function implemented within a given "black box". Since such a reconstruction is typically performed sequentially, step-by-step, this type of problem can be considered a sequential type

In studies conducted to date on Boolean concepts:

- ✓ The research population had no Boolean background (they did not study any formal course on the topic)
- ✓ The studies focused on relatively simple concepts
- ✓ The examinees were tested either on Recognition problems and Reverse Engineering problems
- ✓ The effects of Boolean function characteristics such as symmetry, linearity, etc. were not tested for complexity
- ✓ The results were examined based on a correct or incorrect answer, no partial solutions were examine

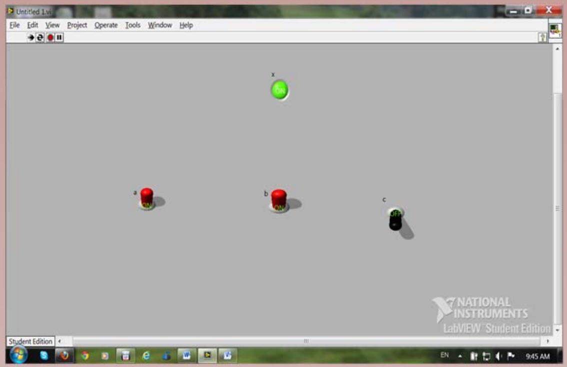
Our study

- ✓ The research population is Bachelor of Engineering students that have taken a Logic Systems course
- ✓ More complex concepts were examined in the study
- ✓ The examinees were also tested on Recognition problems and Reverse Engineering problems given the same Boolean concepts
- ✓ The effect of symmetric functions on Boolean complexity was examined

Method and Experiment

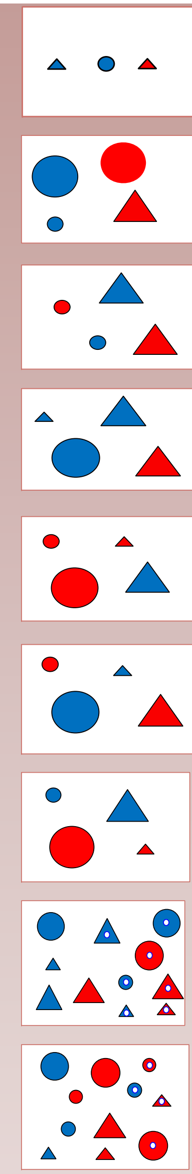
- The research population included thirty 1st year students studying for a Bachelor of Engineering degree at a college
- All students studied the Logic Design course in the same study group and with the same lecturer
- All students completed the course successfully with the final exam average grade of 75
- The experiment was conducted in two stages for 9 concepts, where each concept was described by means of a Boolean expression

The first stage: Reverse Engineering



- An experimental environment using Lab View was developed on a computer monitor for “black box” RE problems
- The state of the switches could be changed by clicking on the appropriate key with the mouse
- According to the switch’s state, the light bulb is either on or off
- The participants were required to try the different switch combinations that light the bulb and describe the combinations that light the bulb for each of the 9 concepts using a Boolean expression
- The tasks were presented as a simulation on a computer monitor and the time taken to complete each task was measured

The second stage: Recognition



- Recognition problems were examined using a questionnaire with nine patterns, where each pattern represents one of the nine concepts examined, respectively
- The participants were asked to describe each of the patterns with as few literals as possible
- A maximum of 45 minutes was allocated to completing the questionnaire
- The two stages of the experiment were conducted two days apart

Results and Conclusion

Boolean Concept	Minimal Description (Feldman)	Structural Complexity (Vigo)	RE.Prob %	Rec.Prob %
$\bar{b}(a+c)$	3	1.54	100	100
$\bar{a}\, c+b\, \bar{c}$	4	2.14	90	70
$\bar{a}\, \bar{b}+ab$	4	2.14	100	80
$a(b+c)+bc$	5	2.14	100	90
$(\bar{a}+\bar{b})\bar{c}+a\, b\, c$	6	2.34	60	40
$a(\bar{b}\, c+\bar{c}\, b)+\bar{a}(\bar{b}\, \bar{c}+b\, c)$	10	4.00	40	30
$a(b\, c+\bar{c}\, \bar{b})+\bar{a}(\bar{b}\, c+b\, \bar{c})$	10	4.00	60	30
$a(b+c+d)+b(d+c)+cd$	9	4.48	90	80
$\bar{a}(\bar{b}+\bar{c}+\bar{d})+\bar{b}(\bar{d}+\bar{c})+\bar{c}\, \bar{d}$	9	4.48	90	70

- Participants are more successful in solving RE problems than solving recognition problems
- Not a single participant that did not succeed in solving RE problems managed to solve recognition problems for the same concept
- However, not all the participants that managed to solve RE problems were also successful in solving the recognition problems
- Recognition problems are more difficult to learn than problems where the information is sequentially obtained, in this case RE problems
- The difficulty not only depends on the concept's complexity but also on the complexity of the manner in which the problem is presented

- The majority of participants did not recognize the “xor” operator in both types of problems
- Participants that grasped the “xor” concept as an operator to the same degree as the “or” and “and” concepts were more successful in solving the problem
- Apparently, the MD and SC measures are not sufficiently reliable in predicting the level of difficulty in solving the problems for symmetrical functions

Further research

- ✓ The same research population presented here that was tested for Recognition and Reverse Engineering problems will be tested for discovering failures and Reverse Engineering for sequentially logic systems and more complex combinations

- ✓ The results will be examined not only based on a correct or incorrect answer but also on the way it was solved and the strategy employed to solve it

- ✓ Additional characteristics of Boolean functions will be examined, such as linearity and threshold functions

"*Logic takes care of itself; all we have to do is to look and see how it does it*". Ludwig Wittgenstein

Thank You