

# 人工智能基础-七巧板

## 项目报告

姓名： 陈华玉

学号： 2017011518

班级： 自76

## 目录

1	系统简述.....	1
2	问题分析.....	1
2.1	先建图，后搜索思路.....	1
2.2	启发式搜索函数.....	2
2.3	对比试验.....	3
3	十三巧板.....	3
4	统计.....	4

## 1 系统简述

本次大作业，基于python算法后端与pyqt5图形化库，我们设计了七巧板解空间搜索的工程项目。该项目主要完成以下功能：

- 必做 设计实现七巧板搜索算法并设计相关图形化界面
- 必做 允许选手自定义输入图片
- 选做 设计13巧板（9块）相关解决办法

本项目相对与同类项目最大特点是运行速度快，在7巧板项目的核心处理时间少于1s，13巧板处理时间少于2s。

## 2 问题分析

我们将七巧板搜索问题建模为图搜索问题，并给出我们对于七巧板搜索的一般性解决步骤。

*Step 1:* 对输入图像(Picture)进行处理，找出所有的节点（包括外部显式和内部隐藏节点）并确定节点连接关系。

*Step 2:* 将节点与边的相关信息抽检建立成为拓扑图。并在拓扑图(Graph)中搜索所有可能的解空间(Status Space)。

*Step 3:* 对解空间进行启发式搜索（偏向于DFS变体），获得最终解。

以上每一个核心步骤被抽象为一个对象

我认为 *Step1* 属于基本自然图像处理范畴，不再予以详细阐述。

### 2.1 先建图，后搜索思路

解决图搜索问题的一个需要解决的部分即确定一个状态可到达的后继状态。在七巧板问题中，也就是确定下一块图像“可以”放在哪里。这里便有两个思路，一是在搜索过程共在像素级别进行试判断，这种方法相对更为自然，但是由于大量重复计算处理速度很慢。

- 递归搜索的策略

我们将构成七巧板的7个部分称为**图像**（block），观察发现，每个图像其实都可以被分割称为一些相同的小三角形，我将这种小三角形称为元素（element）。

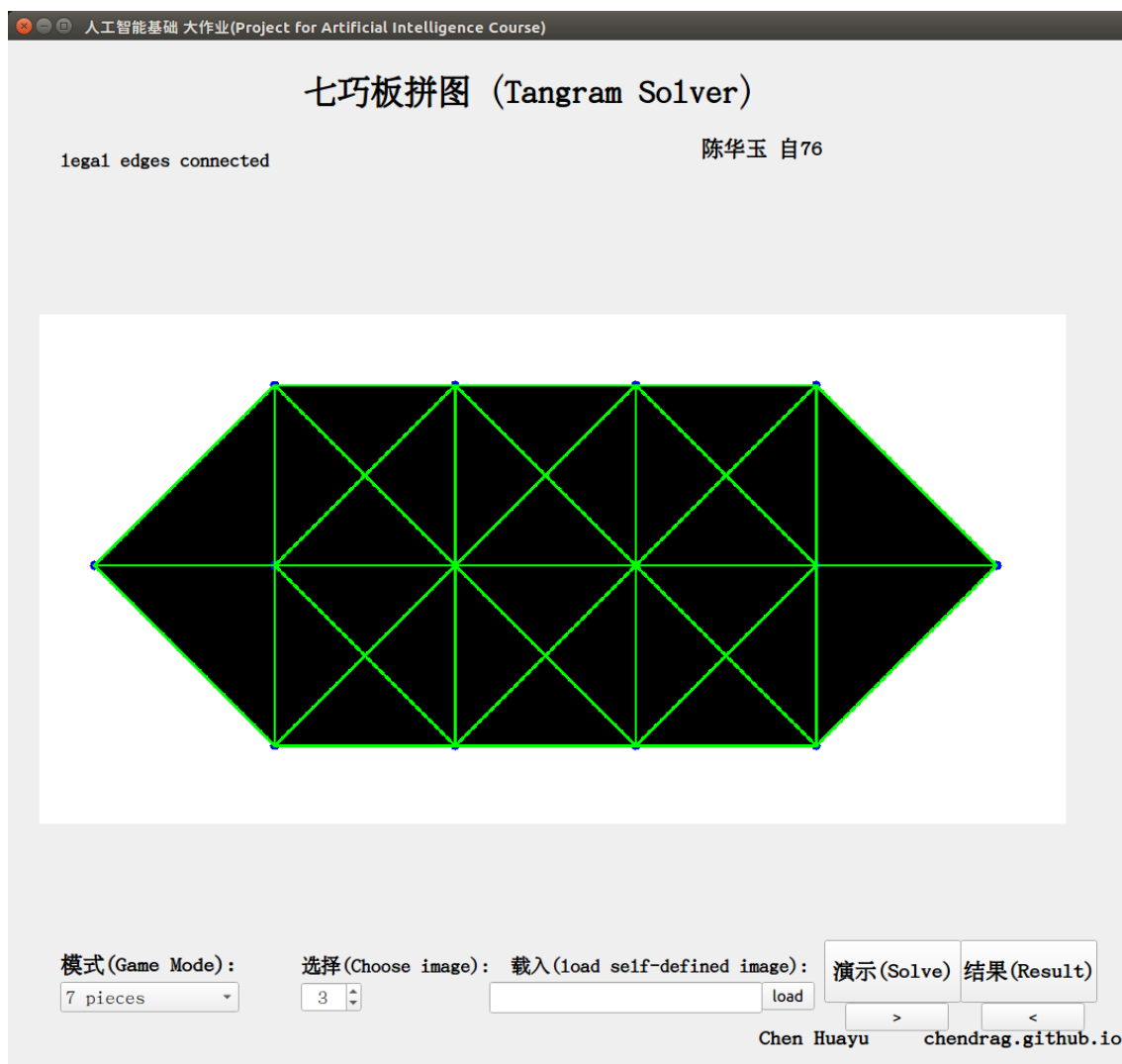


Figure 1一个大形状可以被分割成不同小元素

由于发现七巧板的图形元素之间存在包含关系。我们因而希望先对图形节点信息进行遍历搜索，找出基本元素（smallest triangle）与实际图像（block）的继承包含关系（如一个大三角形可以看做两个中等三角形，每个中等三角形又可以看作两个小三角形）。建立图像继承树来优化搜索时间。

继承树的实现与抽象类 BASESHAPE 与其子类实现。

总之，我们依靠先建图（找全搜索空间），后搜索思路大幅提高了计算速度（1s内）。了解到部分同学搜索13巧板所需时间超过10s（了解到其中一位同学甚至达到1min以上），可以认为我们的搜索效率得到了极为有效的优化。

## 2.2 启发式搜索函数

深入思考后，我认为课堂所讲的启发式搜索方法A\*方法并不能直接运用于本项目。这里主要原因是因为，A\*算法面向的路径搜索问题的主要目的是在解空间中搜索到代价最小的解，在此基础上追求速度更快。而七巧板问题中不存在代价最小的解（都是7步），因而主要追求速度最优。但是泛化的启发式函数仍然可

以被设计出来。在这里我们的启发式函数为：

$F = (g + w * h) / d$   
 $g = (-1) * \text{depth}$  \*其中depth即已经搜索的深度，保证为深度优先搜索  
 $h = \text{order}$  \*用于规范不同元素的搜索顺序  
 $w = \text{weight}$  \*w足够大保证h所规范的顺序优先于深度  
 $d(\text{eath}) = \text{alive}(\text{status})$

- alive函数

alive函数对搜索空间的一个状态进行判定，当程序判定一个状态一定不能产生最终解时（如有一个孤立的元素出现），则认为该状态死亡， $d = 0$ ，代价无穷大，不产生子代。

- 规范的顺序

不同形状搜索的顺序被预先定义好。但是本项目采用动态定义的方式，如一个形状在建图（graph）环节中，若被发现可能的解空间越少，则其将会越优先搜索（一般而言，我们发现面积更大，或形状更为奇异的图像倾向于被先搜索，也就是order越小）。可以证明在概率意义上着有利于减少搜索次数。

- Weight的设置

w的值是一个虚值并非实际调参确定，可以理解为w足够大，使得程序大体上按照分析出的order搜索，在order相同时（也就是说搜索深度相同），会进行深度优先搜索。

## 2.3 对比试验

平均搜索次数	浮动顺序（正常）	顺序	逆序	无生死
7巧板	10.5	10	282	12
13巧板	68	372	>10000	>10000

我们在tangram 13, 14, 15, 20与十三巧板数据上实验

- \*顺序表示面积大的固定优先搜索
- \*逆序表示面积小的固定有限搜索
- \*无生死表示永远为1不作额外判断

可以看出启发函数在搜索空间较大时优势更加明显，且alive函数设置可以避免非常多的不必要判断。

## 3 十三巧板

与一些基于像素级别边搜边判断的方法相比，本方法的劣势在于每次给定一个新的七巧板问题，都需要对基本形状（block）的相互集成关系重新定义。但是好处在于一旦定义完成，对于图像的预处理速度极快，搜索代价极小。可以认

为这是用工程复杂度换效率的一个trade off。十三巧版实际上某种程度比七巧板更加简单，因为其基本的element都是一个一个小方块。

由于十三巧版网上并没用优质数据，因而这里我只拿了同学人工生成的一个数据作演示。同样，搜索过程可以再程序中看到。



#### 4 统计

我们在所有输入数据中进行统计。

	7巧板	13巧板
平均搜索次数	11.4	68
理论最少次数	7	9

可见我们的策略设计时相当有效的。